Preparing and Using Service Blueprints in vRealize Automation

21 July 2021

vRealize Automation 7.6

You can find the most up-to-date technical documentation on the VMware website at:

https://docs.vmware.com/

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Contents

Preparing and Using Service Blueprints in vRealize Automation

1

You prepare vRealize Automation to support the service blueprints that you provide to your users in the service catalog. The service blueprints can range from a single, simple machine with no guest operating system to complex custom application stacks delivered on multiple machine under a load balancer.

Depending on the service blueprints that you provide, the preparation might include configuring your environment for integration with vRealize Automation, and ensuring that your tenants and resources can support your environment.

You then use vRealize Automation to design and publish the service blueprints that meet the needs of your service catalog users.

Foundations and Concepts 2

Before you begin working with vRealize Automation, you can familiarize yourself with basic vRealize Automation concepts.

This chapter includes the following topics:

vRealize Automation Environment User Interfaces

You use and manage your vRealize Automation environment with several interfaces.

User Interfaces

These tables describe the interfaces that you use to manage your vRealize Automation environment.

Table 2-1. vRealize AutomationAdministration Console

Purpose	Access	Required Credentials
You use the vRealize Automation console for these system administrator tasks. - Add tenants. - Customize the vRealize Automation user interface. - Configure email servers. - View event logs. - Configure vRealize Orchestrator.	1 Start a browser and open the vRealize Automation appliance splash page using the fully qualified domain name of the virtual appliance: https://*vrealize-automation-appliance-FQDN*. 2 Click **vRealize Automation console**. You can also use this URL to open the vRealize Automation console: https://*vrealize-automation-appliance-FQDN*/vcac 3 Log in.	You must be a user with the system administrator role.

Table 2-2. vRealize Automation Tenant Console. This interface is the primary user interface that you use to create and manage your services and resources.

Purpose	Access	Required Credentials
You use vRealize Automation for these tasks. - Request new IT service blueprints. - Create and manage cloud and IT resources. - Create and manage custom groups. - Create and manage business groups. - Assign roles to users.	1 Start a browser and enter the URL of your tenancy using the fully qualified domain name of the virtual appliance and the tenant URL name: https://*vrealize-automation-appliance-FQDN*/vcac/org/*tenant_URL_name* . 2 Log in.	You must be a user with one or more of these roles: - Application Architect - Approval Administrator - Catalog Administrator - Container Administrator - Container Architect - Health Consumer - Infrastructure Architect - Secure Export Consumer - Software Architect - Tenant Administrator - XaaS Architect

Table 2-3. vRealize Automation Appliance Management Interface.

Purpose	Access	Required Credentials
You use vRealize Automation Appliance Management for these tasks. ■ View the status of registered services. ■ View system information and reboot or shutdown the appliance. ■ Manage participation in the Customer Experience Improvement Program. ■ View network status. ■ View update status and install updates. ■ Manage administration settings. ■ Manage vRealize Automation host settings. ■ Manage SSO settings. ■ Manage product licenses. ■ Configure the vRealize Automation Postgres database. ■ Configure vRealize Automation messaging. ■ Configure vRealize Automation logging. ■ Install IaaS components. ■ Migrate from an existing vRealize Automation installation. ■ Manage IaaS component certificates. ■ Configure Xenon service.	1 Start a browser and open the vRealize Automation appliance splash page using the fully qualified domain name of the virtual appliance: https://*vrealize-automation-appliance-FQDN* 2 Click **vRealize Automation Appliance Management**. You can also use this URL to open the vRealize Automation appliance management interface: https://*vrealize-automation-appliance-FQDN*:5480 3 Log in.	■ User name: root ■ Password: Password you entered when you deployed the vRealize Automation appliance.

Table 2-4. vRealize Orchestrator Client

Purpose	Access	Required Credentials
You use the vRealize Orchestrator Client for these tasks. ■ Develop actions. ■ Develop workflows. ■ Manage policies. ■ Install packages. ■ Manage user and user group permissions. ■ Attach tags to URI objects. ■ View inventory.	1 Start a browser and open the vRealize Automation splash page using the fully qualified domain name of the virtual appliance: https://*vrealize-automation-appliance-FQDN* 2 To download the client.jnlp file to your local computer, click **vRealize Orchestrator Client**. 3 Right-click the `client.jnlp` file and select **Launch**. 4 On the Do you want to Continue? dialog box, click **Continue**. 5 Log in.	You must be a user with the system administrator role or part of the vcoadmins group configured in the vRealize Orchestrator Control Center Authentication Provider settings.

Table 2-5. vRealize Orchestrator Control Center

Purpose	Access	Required Credentials
You use the vRealize Orchestrator Control Center to edit the configuration of the default vRealize Orchestrator instance that is embedded in vRealize Automation.	1 Start a browser and open the vRealize Automation appliance splash page using the fully qualified domain name of the virtual appliance: https://*vrealize-automation-appliance-FQDN* 2 Click **vRealize Automation Appliance Management**. You can also use this URL to open the vRealize Automation appliance management interface: https://*vrealize-automation-appliance-FQDN*:5480 3 Log in. 4 Click **vRA > Orchestrator**. 5 Select **Orchestrator user interface**. 6 Click **Start**. 7 Click the Orchestrator user interface URL. 8 Log in.	User Name ■ Enter **`root`** if role-based authentication is not configured. ■ Enter your vRealize Automation user name if it is configured for role-based authentication. Password ■ Enter the password you entered when you deployed the vRealize Automation appliance if role-based authentication is not configured. ■ Enter the password for your user name if your user name is configured for role-based authentication.

Table 2-6. Linux Command Prompt

Purpose	Access	Required Credentials
You use the Linux command prompt on a host, such as the vRealize Automation appliance host, for these tasks. ■ Stop or start services ■ Edit configuration files ■ Run commands ■ Retrieve data	1 On the vRealize Automation appliance host, open a command prompt. One way to open the command prompt on your local computer is to start a session on the host using an application such as PuTTY. 2 Log in.	■ User name: root ■ Password: Password you created when you deployed the vRealize Automation appliance.

Table 2-7. Windows Command Prompt

Purpose	Access	Required Credentials
You can use a Windows command prompt on a host, such as the IaaS host, to run scripts.	1 On the IaaS host, log in to Windows. One way to log in from your local computer is to start a remote desktop session. 2 Open the Windows command prompt. One way to open the command prompt is to right-click the Start icon on the host and select **Command Prompt** or **Command Prompt (Admin)**.	■ User name: User with administrative privileges. ■ Password: User's password.

Introducing vRealize Automation

IT organizations can use VMware vRealize ™ Automation to deliver services to their lines of business.

vRealize Automation provides a secure portal where authorized administrators, developers, or business users can request new IT services and manage specific cloud and IT resources, while ensuring compliance with business policies. Requests for IT services, including infrastructure, applications, desktops, and many others, are processed through a common service catalog to provide a consistent user experience.

To improve cost control, you can integrate vRealize Business for Cloud with your vRealize Automation instance to expose the month-to-date expense of cloud and virtual machine resources, and help you better manage capacity, price, and efficiency.

Note Beginning with version 7.3, vRealize Automation supports only vRealize Business for Cloud version 7.3 and later.

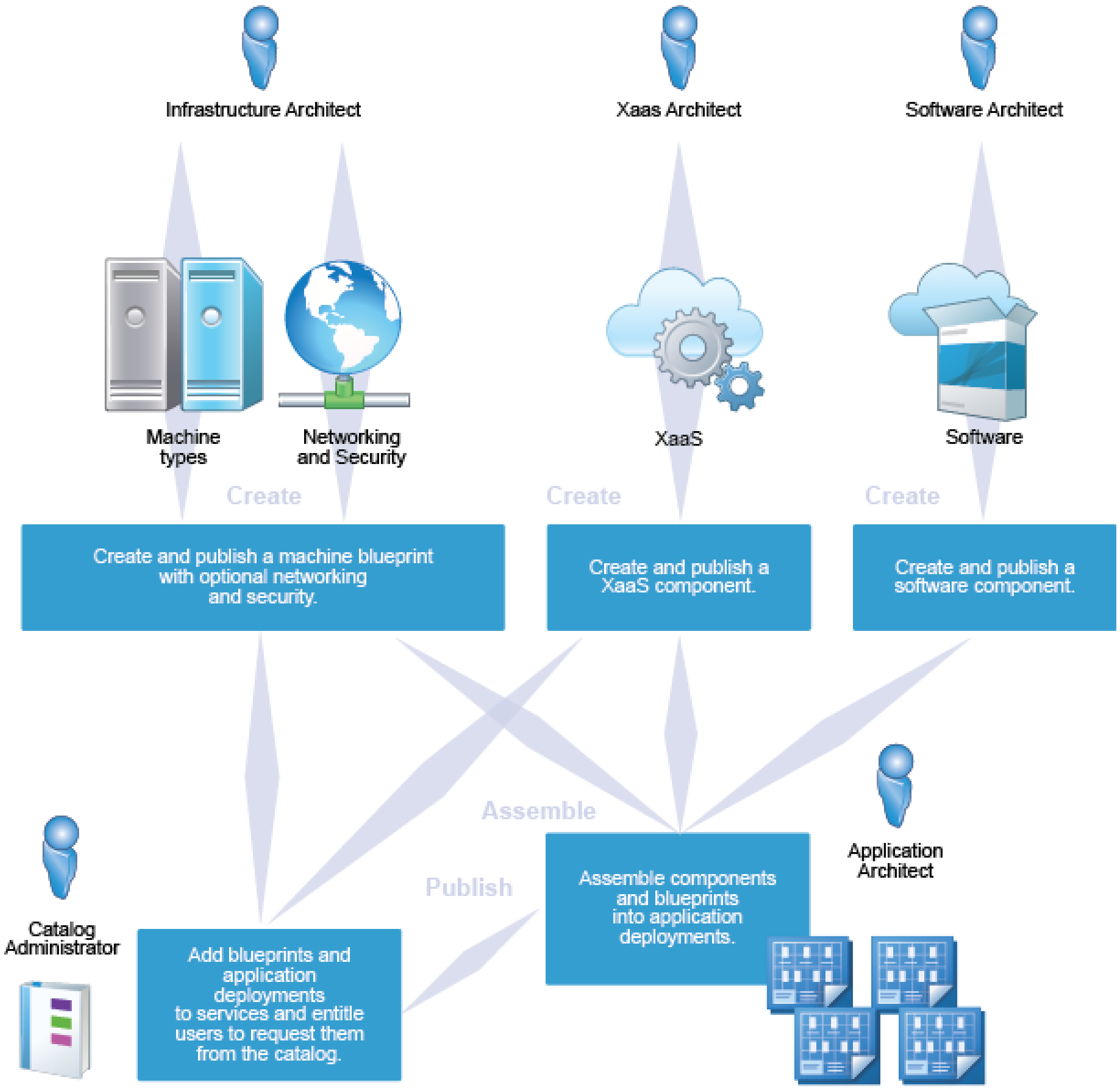
Providing On-Demand Services to Users Overview

You can use the IaaS, Software, and XaaS features of vRealize Automation to model custom on-demand IT services and deliver them to your users through the vRealize Automation common service catalog.

You use blueprints to define machine deployment settings. Published blueprints become catalog items, and are the means by which entitled users provision machine deployments. Catalog items can range in complexity from a single, simple machine with no guest operating system to complex custom application stacks delivered on multiple machines under an NSX load balancer with networking and security controls.

You can create and publish blueprints for a single machine deployment, or a single custom XaaS resource, but you can also combine machine blueprints and XaaS blueprints with other building blocks to design elaborate application blueprints that include multiple machines, networking and security, software with full life cycle support, and custom XaaS functionality. You can also control deployment settings by using a parameterized blueprint, which allows you to specify pre-configured size and image settings at request time. Because all published blueprints and blueprint components are reusable, you can create a library of these components and combine them in new nested blueprints to deliver increasingly complex on-demand services.

Published blueprints become catalog items that your service catalog administrators can deliver to your users. The service catalog provides a unified self-service portal for consuming IT services. Service catalog administrators can manage user access to catalog services, items, and actions by using entitlements and approvals, and users can browse the catalog to request items they need, track their requests, and manage their provisioned items.


Infrastructure Architect
Xaas Architect
Software Architect
Machine types
Networking and Security
XaaS
Software
Create
Create
Create
Create and publish a machine blueprint with optional networking and security.
Create and publish a XaaS component.
Create and publish a software component.
Assemble
Publish
Application Architect
Catalog Administrator
Assemble components and blueprints into application deployments.
Add blueprints and application deployments to services and entitle users to request them from the catalog.

- Infrastructure as a Service Overview

  With Infrastructure as a Service (IaaS), you can rapidly model and provision servers and desktops across virtual and physical, private and public, or hybrid cloud infrastructures.

- Software Components Overview

  Software components automate the installation, configuration, and life cycle management of middleware and application deployments in dynamic cloud environments. Applications can range from simple Web applications to complex and even packaged applications.

- XaaS Overview

  With the XaaS, XaaS architects can create XaaS blueprints and resource action, and publish them as catalog items.

- Service Catalog Overview

  The service catalog provides a unified self-service portal for consuming IT services. Users can browse the catalog to request items they need, track their requests, and manage their provisioned items.

- Containers Overview

  You can use containers to gain access to additional instrumentation for developing and deploying applications in vRealize Automation.

Infrastructure as a Service Overview

With Infrastructure as a Service (IaaS), you can rapidly model and provision servers and desktops across virtual and physical, private and public, or hybrid cloud infrastructures.

Modeling is accomplished by creating a machine blueprint, which is a specification for a machine. Blueprints are published as catalog items in the common service catalog, and are available for reuse as components inside of application blueprints. When an entitled user requests a machine based on one of these blueprints, IaaS provisions the machine.

With IaaS, you can manage the machine life cycle from a user request and administrative approval through decommissioning and resource reclamation. Built-in configuration and extensibility features also make IaaS a highly flexible means of customizing machine configurations and integrating machine provisioning and management with other enterprise-critical systems such as load balancers, configuration management databases (CMDBs), ticketing systems, IP address management systems, or Domain Name System (DNS) servers.

Software Components Overview

Software components automate the installation, configuration, and life cycle management of middleware and application deployments in dynamic cloud environments. Applications can range from simple Web applications to complex and even packaged applications.

By using a configurable scriptable engine, software architects fully control how middleware and application deployment components are installed, configured, updated, and uninstalled on machines. Through the use of Software properties, software architects can require or allow blueprint architects and end-users to specify configuration elements such as environment variables. For repeated deployments, these blueprints standardize the structure of the application, including machine blueprints, software components, dependencies, and configurations, but can allow environment variables and property binding to be reconfigured if necessary.

To successfully add software components to the design canvas, you must also have business group member, business group administrator, or tenant administrator role access to the target catalog.

Deploying Any Application and Middleware Service

You can deploy Software components on Windows or Linux operating systems on vSphere, vCloud Director, vCloud Air, and Amazon Web Services machines.

- IaaS architects create reusable machine blueprints based on templates, snapshots, or Amazon machine images that contain the guest agent and Software bootstrap agent to support Software components.
- Software architects create reusable software components that specify exactly how the software is installed, configured, updated during deployment scale operations, and uninstalled on machines.
- Software architects, IaaS architects, and application architects use a graphical interface to model application deployment topologies. Architects reconfigure Software properties and bindings as required by the software architect, and publish application blueprints that combine Software components and machine blueprints.
- Catalog administrators add the published blueprints to a catalog service, and entitle users to request the catalog item.
- Entitled users request the catalog item and provide any configuration values designed to be editable. vRealize Automation deploys the requested application, provisioning any machine(s), networking and security components, and Software component(s) defined in the application blueprint.
- Entitled users request the scale in or scale out actions to adjust their deployments to changing workload demands. vRealize Automation installs or uninstalls Software components on machines for scale, and runs update scripts for dependent Software components.

Standardization in Software

With Software, you can create reusable services using standardized configuration properties to meet strict requirements for IT compliance. Software includes the following standardized configuration properties:

- Model-driven architecture that enables adding IT certified machine blueprints and middleware services within the application blueprint.
- A delegation model for overriding configuration name value pairs between software architect, application architect, and end user to standardize configuration values for application and middleware service.

Software Extensibility and Open Architecture

You can download predefined Software components for a variety of middleware services and applications from the VMware Solution Exchange. Using either the vRealize CloudClient or vRealize Automation REST API , you can programmatically import predefined Software components into your vRealize Automation instance.

- To visit the VMware Solution Exchange, see https://solutionexchange.vmware.com/store/category_groups/cloud-management.

- For information about vRealize Automation REST API, see *Programming Guide* and *vRealize Automation API Reference*.
- For information about vRealize CloudClient, see https://developercenter.vmware.com/tool/cloudclient.

XaaS Overview

With the XaaS, XaaS architects can create XaaS blueprints and resource action, and publish them as catalog items.

With XaaS, you can provide anything as a service using the capabilities of VMware vRealize ™ Orchestrator ™. For example, you can create a blueprint that allows a user to request a backup of a database. After completing and submitting a backup request, the user receives a backup file of the database they specified.

An XaaS architect can create custom resource types mapped to vRealize Orchestrator object types and define them as items to be provisioned. A XaaS architect can then create blueprints from vRealize Orchestrator workflows and publish the blueprints as catalog items. The vRealize Orchestrator workflows can be either predefined or independently developed by workflow developers.

You can also use the XaaS to design additional actions that the consumer can perform on the provisioned items. These additional actions are connected to vRealize Orchestrator workflows and take the provisioned item as input to the workflow. To use this function for items provisioned by sources other than the XaaS, you must create resource mappings to define their resource types in vRealize Orchestrator.

For more information about vRealize Orchestrator and its capabilities, see the vRealize Orchestrator documentation.

Service Catalog Overview

The service catalog provides a unified self-service portal for consuming IT services. Users can browse the catalog to request items they need, track their requests, and manage their provisioned items.

Service architects and administrators can define new services and publish them to the common catalog. When defining a service, the architect can specify the kind of item that can be requested, and what options are available to the consumer as part of submitting the request.

Group managers or line-of-business administrators can specify business policies such as who is entitled to request specific catalog items or perform specific actions on provisioned items. They can also apply configurable approval policies to catalog requests.

Users responsible for managing the catalog, such as tenant administrators and service architects, can manage the presentation of catalog items to the consumers of IT services, for example by grouping items into service categories for easier navigation and highlighting new services to consumers in a broadcast message.

Containers Overview

You can use containers to gain access to additional instrumentation for developing and deploying applications in vRealize Automation.

Containers for vRealize Automation allows vRealize Automation to support containers. You can provision an application that is built from containers or from a combination of containers and VMs.

Container administrators can use Containers to perform the following tasks:

- Model containerized applications in vRealize Automation blueprints.
- Provision container hosts from the vRealize Automation service catalog.
- Manage container hosts from within vRealize Automation.
- Create and configure hosts.
- Set resource quotas for containers.
- Work with templates, images, and registries.
- Create and edit blueprints in the vRealize Automation service catalog.
- Develop multi-container templates.

Container architects can add container components to a vRealize Automation blueprint.

The integrated Containers application uses the Docker Remote API to provision and manage containers, including retrieving information about container instances. From a deployment perspective, developers can use Docker Compose to create their application and deploy it through Containers in vRealize Automation. Because that application is ready to be promoted from development to production, developers can enhance the application to include dynamic networks or micro-segmentation.

Cloud administrators can manage the container host infrastructure, for example to govern capacity quotas and approval workflows.

vRealize Business for Cloud Overview

With vRealize Business for Cloud, directors of cloud operations can monitor their expenditures and design more price-efficient cloud services.

vRealize Business for Cloud provides the following benefits:

- Drives accountability by providing visibility into the price of virtual infrastructure and public cloud providers and providing daily price and month-to-date expense updates in vRealize Automation.
- Promotes efficiencies in the virtual infrastructure by making it possible to compare the prices, efficiency, and availability of their private cloud with public cloud providers and industry benchmark data.

- Optimizes decisions about placement for virtual workloads and tradeoffs between buying new hardware and using public cloud providers.

For more information about vRealize Business for Cloud, see the vRealize Business for Cloud documentation.

Tenancy and User Roles

vRealize Automation supports multiple tenants in the same installation. Users always log in and perform their tasks in a specific tenant. Some administrator roles can manage configuration that affects multiple tenants.

Tenancy Overview

A tenant is an organizational unit in a vRealize Automation deployment. A tenant can represent a business unit in an enterprise or a company that subscribes to cloud services from a service provider.

Each tenant has its own dedicated configuration. Some system-level configuration is shared across tenants.

Table 2-8. Tenant Configuration

Configuration Area	**Description**
Login URL	Each tenant has a unique URL to the vRealize Automation console. ■ The default tenant URL is in the following format: https://*hostname*/vcac ■ The URL for additional tenants is in the following format: https://*hostname*/vcac/org/*tenantURL*
Identity stores	Each tenant requires access to one or more directory services, such as OpenLDAP or Microsoft Active Directory servers, that are configured to authenticate users. You can use the same directory service for more than one tenant, but you must configure it separately for each tenant.
Branding	A tenant administrator can configure the branding of the vRealize Automation console including the logo, background color, and information in the header and footer. System administrators control the default branding for all tenants.
Notification providers	System administrators can configure global email servers that process email notifications. Tenant administrators can override the system default servers, or add their own servers if no global servers are specified.
Business policies	Administrators in each tenant can configure business policies such as approval workflows and entitlements. Business policies are always specific to a tenant.
Service catalog offerings	Service architects can create and publish catalog items to the service catalog and assign them to service categories. Services and catalog items are always specific to a tenant.
Infrastructure resources	The underlying infrastructure fabric resources, for example, vCenter servers, Amazon AWS accounts, or Cisco UCS pools, are shared among all tenants. For each infrastructure source that vRealize Automation manages, a portion of its compute resources can be reserved for users in a specific tenant to use.

About the Default Tenant

When the system administrator configures an Active Directory link using Directories management during the installation of vRealize Automation, a default tenant is created with the built-in system administrator account to log in to the vRealize Automation console. The system administrator can then configure the default tenant and create additional tenants.

The default tenant supports all of the functions described in Tenant Configuration. In the default tenant, the system administrator can also manage system-wide configuration, including global system defaults for branding and notifications, and monitor system logs.

User and Group Management

All user authentication is handled by Active Directory links that are configured through Directories Management. Each tenant has one or more Active Directory links that provide authentication on a user or group level.

The root system administrator performs the initial configuration of single sign-on and basic tenant creation and setup, including designating at least one tenant administrator for each tenant. Thereafter, a tenant administrator can configure Active Directory links and assign roles to users or groups as needed from within their designated tenant.

Tenant administrators can also create custom groups within their own tenants and add users and groups to those groups. Custom groups can be assigned roles or designated as the approvers in an approval policy.

Tenant administrators can also create business groups within their tenants. A business group is a set of users, often corresponding to a line of business, department or other organizational unit, that can be associated with a set of catalog services and infrastructure resources. Users and custom groups can be added to business groups.

Comparison of Single-Tenant and Multi-tenant Deployments

vRealize Automation supports deployments with either a single tenant or multiple tenants. The configuration can vary depending on how many tenants are in your deployment. Many NSX and vSphere-related blueprint selections are tenant-specific.

System-wide configuration is always performed in the default tenant and can apply to one or more tenants. For example, system-wide configuration might specify defaults for branding and notification providers.

Infrastructure configuration, including the infrastructure sources that are available for provisioning, can be configured in any tenant and is shared among all tenants. You divide your infrastructure resources, such as cloud or virtual compute resources, into fabric groups and assign an administrator to manage those resources as the fabric administrator. Fabric administrators can allocate resources in their fabric group to business groups by creating reservations.

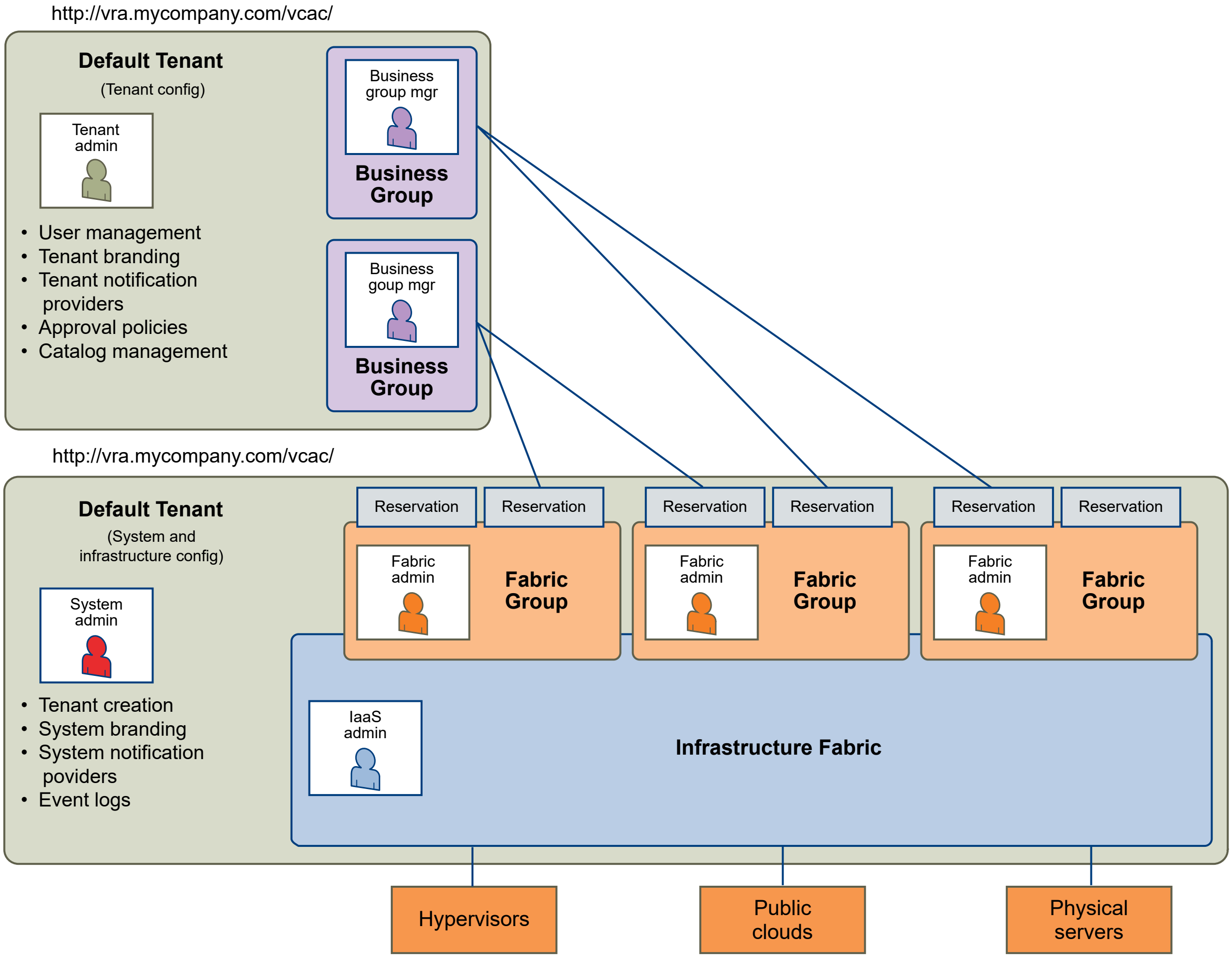
To support tenant allocation of vSphere and NSX endpoint resources, only the network profiles, reservation policies, storage policies, security groups and tags, and transport zones that are applicable to the current tenant are visible when authoring blueprints.

Single-Tenant Deployment

In a single-tenant deployment, all configuration can occur in the default tenant. Tenant administrators can manage users and groups, configure tenant-specific branding, notifications, business policies, and catalog offerings.

All users log in to the vRealize Automation console at the same URL, but the features available to them are determined by their roles.

Figure 2-1. Single-Tenant Example



Note In a single-tenant scenario, it is common for the system administrator and tenant administrator roles to be assigned to the same person, but two distinct accounts exist. The system administrator account is always administrator@vsphere.local, and the system administrator account creates a local user account to assign the tenant administrator role.

Multi-tenant Deployment

In a multi-tenant environment, the system administrator creates tenants for each organization that uses the same vRealize Automation instance. Tenant users log in to the vRealize Automation console at a URL specific to their tenant. Tenant-level configuration is segregated from other tenants and from the default tenant. Users with system-wide roles can view and manage configuration across multiple tenants.

There are two main scenarios for configuring a multi-tenant deployment.

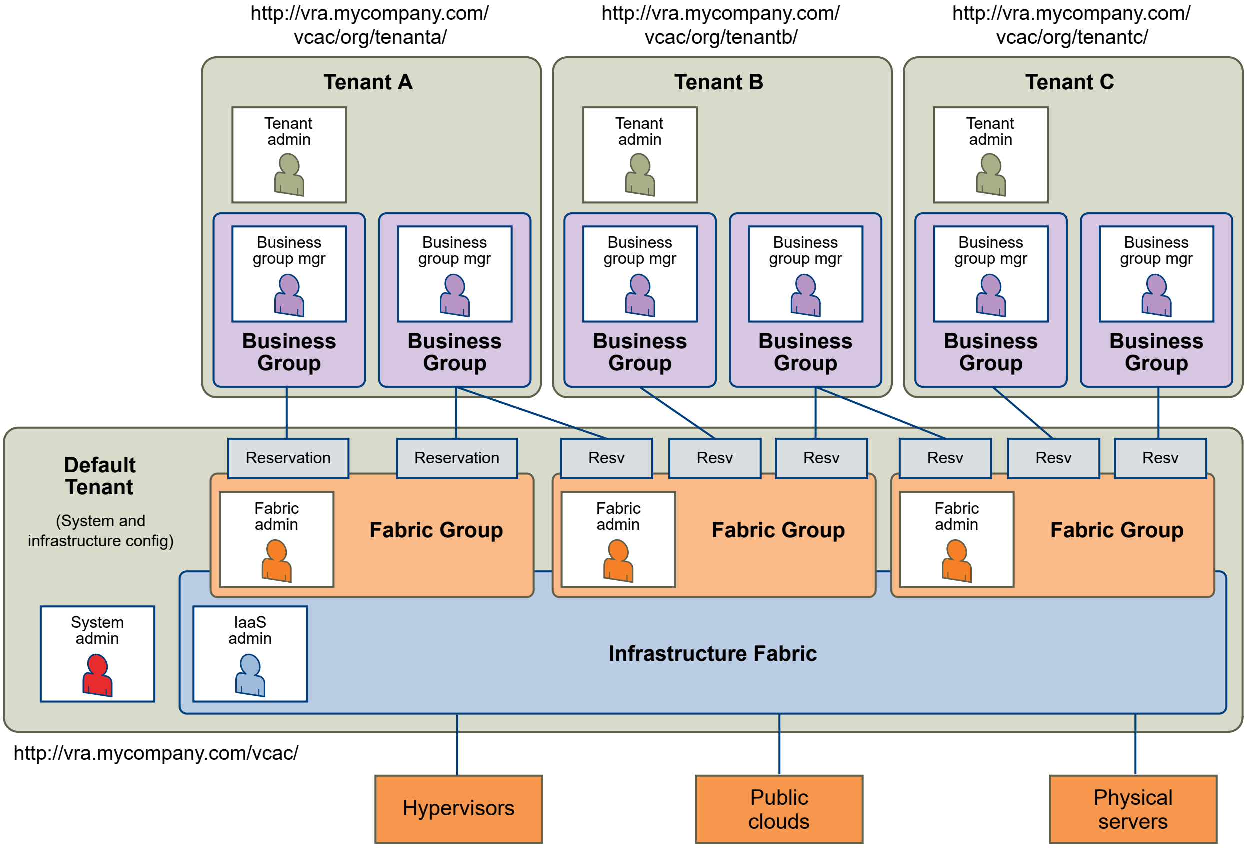
Table 2-9. Multi-tenant Deployment Examples

Example	Description
Manage infrastructure configuration only in the default tenant	In this example, all infrastructure is centrally managed by IaaS administrators and fabric administrators in the default tenant. The shared infrastructure resources are assigned to the users in each tenant by using reservations.
Manage infrastructure configuration in each tenant	In this scenario, each tenant manages its own infrastructure and has its own IaaS administrators and fabric administrators. Each tenant can provide its own infrastructure sources or can share a common infrastructure. Fabric administrators manage reservations only for the users in their own tenant.

The following diagram shows a multi-tenant deployment with centrally managed infrastructure. The IaaS administrator in the default tenant configures all infrastructure sources that are available for all tenants. The IaaS administrator can organize the infrastructure into fabric groups according to type and intended purpose. For example, a fabric group might contain all virtual resources, or all Tier One resources. The fabric administrator for each group can allocate resources from their fabric groups. Although the fabric administrators exist only in the default tenant, they can assign resources to business groups in any tenant.

Note Some infrastructure tasks, such as importing virtual machines, can only be performed by a user with both the fabric administrator and business group manager roles. These tasks might not be available in a multi-tenant deployment with centrally managed infrastructure.

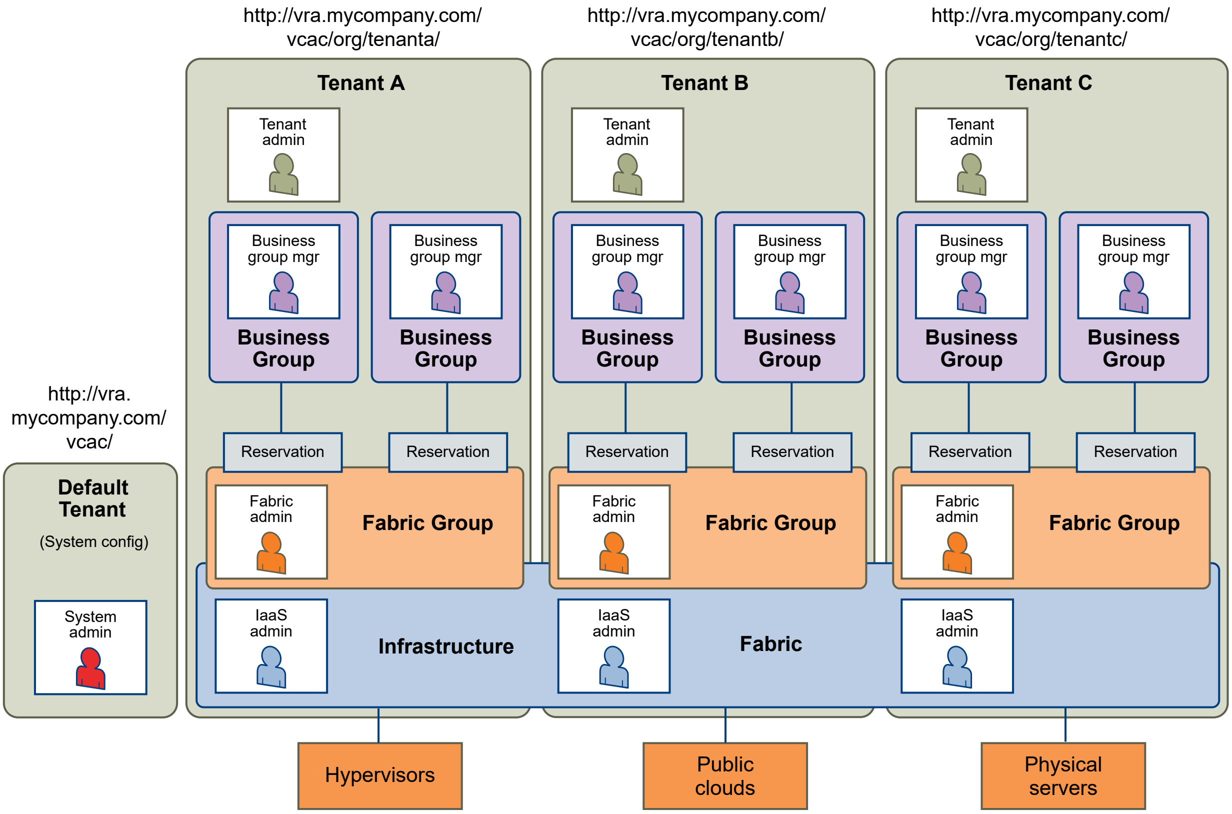
Figure 2-2. Multi-tenant Example with Infrastructure Configuration Only in Default Tenant



The following diagram shows a multi-tenant deployment where each tenant manages their own infrastructure. The system administrator is the only user who logs in to the default tenant to manage system-wide configuration and create tenants.

Each tenant has an IaaS administrator, who can create fabric groups and appoint fabric administrators with their respective tenants. Although fabric administrators can create reservations for business groups in any tenant, in this example they typically create and manage reservations in their own tenants. If the same identity store is configured in multiple tenants, the same users can be designated as IaaS administrators or fabric administrators in each tenant.

Figure 2-3. Multi-tenant Example with Infrastructure Configuration in Each Tenant



User Roles Overview

Roles consist of a set of privileges that can be associated with users to determine what tasks they can perform. Based on their responsibilities, individuals might have one or more roles associated with their user account.

All user roles are assigned within the context of a specific tenant. However, some roles in the default tenant can manage system-wide configuration that applies to multiple tenants.

System-Wide Role Overview

System-wide roles are typically assigned to an IT system administrator. In some organizations, the IaaS administrator role might be the responsibility of a cloud administrator.

System Administrator

The system administrator is typically the person who installs vRealize Automation and is responsible for ensuring its availability for other users. The system administrator creates tenants and manages system-wide configuration such as system defaults for branding and notification providers. This role is also responsible for monitoring system logs.

In a single-tenant deployment, the same person might also act as the tenant administrator.

IaaS Administrator

IaaS administrators manage cloud, virtual, networking, and storage infrastructure at the system level, creating and managing endpoints and credentials, and monitoring IaaS logs. IaaS administrators organize infrastructure into tenant-level fabric groups, appointing the fabric administrators who are responsible for allocating resources within each tenant through reservations and reservation, storage, and networking policies.

System-Wide Roles and Responsibilities

Users with system-wide roles manage configurations that can apply to multiple tenants. The system administrator is only present in the default tenant, but you can assign IaaS administrators to any tenant.

Table 2-10. System-Wide Roles and Responsibilities

Role	Responsibilities	How Assigned
System Administrator	■ Create tenants. ■ Configure tenant identity stores. ■ Assign IaaS administrator role. ■ Assign tenant administrator role. ■ Configure system default branding. ■ Configure system default notification providers. ■ Monitor system event logs, not including IaaS logs. ■ Configure the vRealize Orchestrator server for use with XaaS. ■ Create and manage (view, edit, and delete) reservations across tenants if also a fabric administrator.	Built-in administrator credentials are specified when configuring single sign-on.
IaaS Administrator	■ Configure IaaS features, system and custom properties. ■ Create and manage fabric groups. ■ Create and manage endpoints. ■ Manage endpoint credentials. ■ Configure proxy agents. ■ Manage Amazon AWS instance types. ■ Monitor IaaS-specific logs. ■ Create and manage (view, edit, and delete) reservations across tenants if also a fabric administrator.	The system administrator designates the IaaS administrator when configuring a tenant.

Tenant Role Overview

Tenant roles typically have responsibilities that are limited to a specific tenant and cannot affect other tenants in the system.

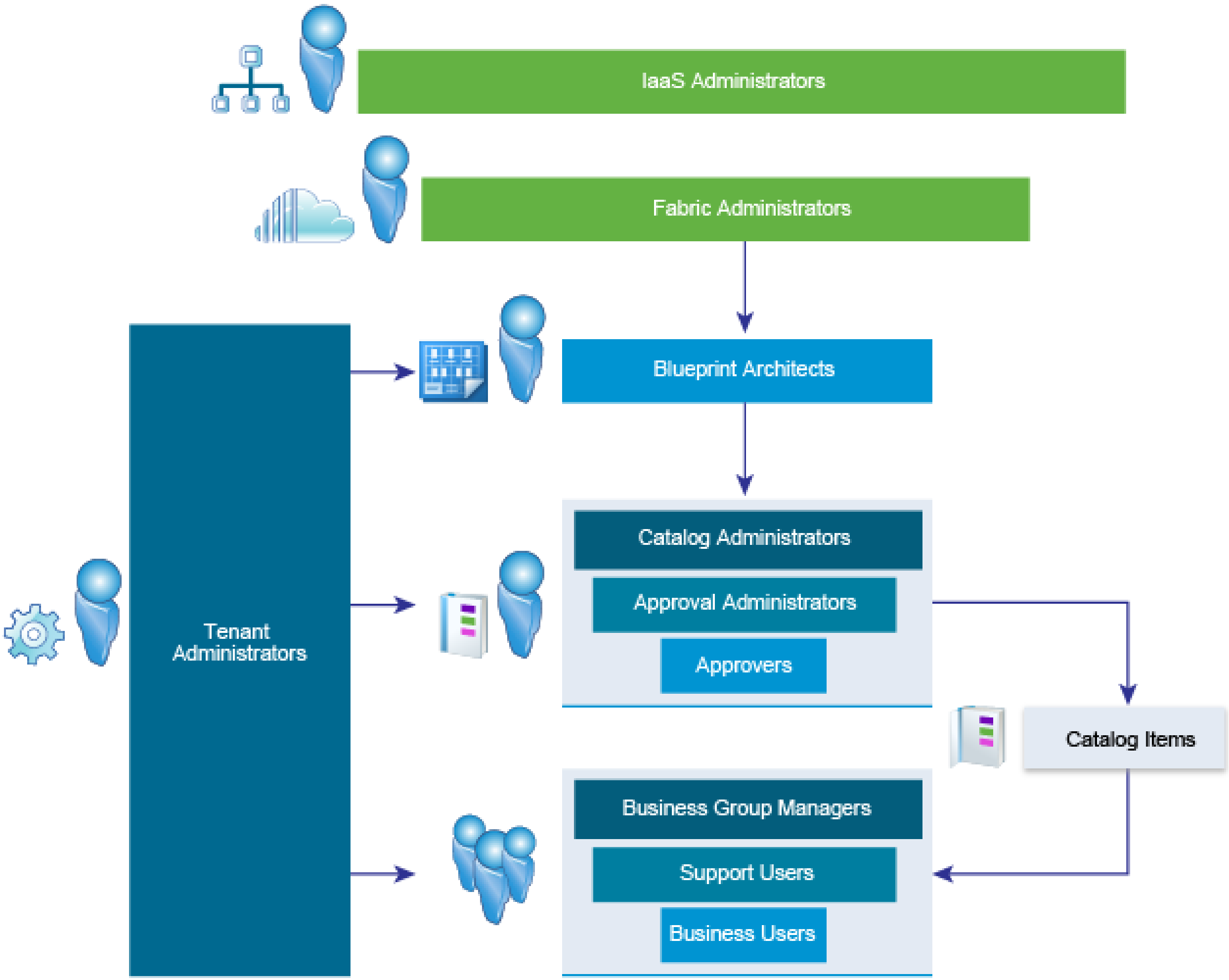

IaaS Administrators
Fabric Administrators
Blueprint Architects
Tenant Administrators
Catalog Administrators
Approval Administrators
Approvers
Catalog Items
Business Group Managers
Support Users
Business Users

Table 2-11. Tenant Role Overview

Role	Description
Tenant Administrator	Typically a line-of-business administrator, business manager, or IT administrator who is responsible for a tenant. Tenant administrators configure vRealize Automation for the needs of their organizations. They are responsible for user and group management, tenant branding and notifications, and business policies such as approvals and entitlements. They also track resource usage by all users within the tenant and initiate reclamation requests for virtual machines.
Fabric Administrator	Manages physical machines and compute resources assigned to their fabric groups and creates and manages the reservations and policies associated with those resources within the scope of their tenant. They also manage property groups, machine prefixes, and the property dictionary that are used across all tenants and business groups. **Note** If you add the fabric administrator role to a system-wide role such as IaaS administrator or system administrator, the fabric administrator can create reservations for any tenant, not just their own.
Blueprint Architects	Umbrella term for the individuals who are responsible for creating blueprint components and assembling the blueprints that define catalog items for consumers to request from the service catalog. These roles are typically assigned to individuals in the IT department, such as architects or analysts.
Catalog Administrator	Creates and manages catalog services and manages the placement of catalog items into services.
Approval Administrator	Defines approval policies. These policies can be applied to catalog requests through entitlements that a tenant administrator or business group manager manage.
Approver	Any user of vRealize Automation, for example, a line manager, finance manager, or project manager, can be designated as an approver as part of an approval policy.
Business Group Manager	Manages one or more business groups. Typically a line manager or project manager. Business group managers entitlements for their groups in the service catalog. They can request and manage items on behalf of users in their groups.
Support User	A role in a business group. Support users can request and manage catalog items on behalf of other members of their groups.

Table 2-11. Tenant Role Overview (continued)

Role	Description
Business User	Any user in the system can be a consumer of IT services. Users can request catalog items from the service catalog and manage their provisioned resources.
Health Consumer	Any user of vRealize Automation, for example, a line manager, finance manager, or project manager, can be designated as a Health Consumer with read-only privileges for Health Service reports.

Tenant Roles and Responsibilities in vRealize Automation

You can assign tenant roles to users in any tenant. The roles have responsibilities that are specific to that tenant.

Table 2-12. Tenant Roles and Responsibilities

Role	Responsibilities	How Assigned
Tenant administrator	■ Customize tenant branding. ■ Manage tenant identity stores. ■ Manage user and group roles. ■ Create custom groups. ■ Manage notification providers. ■ Enable notification scenarios for tenant users. ■ Configure vRealize Orchestrator servers, plug-ins and workflows for XaaS. ■ Create and manage catalog services. ■ Manage catalog items. ■ Manage actions. ■ Create and manage entitlements. ■ Create and manage approval policies. ■ Monitor tenant machines and send reclamation requests.	The system administrator designates a tenant administrator when creating a tenant. Tenant administrators can assign the role to other users in their tenant at any time from the **Administration** tab.
Fabric administrator	■ Manage property groups. ■ Manage compute resources. ■ Manage network profiles. ■ Manage Amazon EBS volumes and key pairs. ■ Manage machine prefixes. ■ Manage property dictionary. ■ Create and manage reservations and reservation policies in their own tenant. ■ If this role is added to a user with IaaS administrator or system administrator privileges, the user can create and manage reservations and reservation policies in any tenant.	The IaaS administrator designates the fabric administrator when creating or editing fabric groups.
Application architect To successfully add software components to the design canvas, you must also have business group member, business group administrator, or tenant administrator role access to the target catalog.	■ Assemble and manage composite blueprints.	Tenant administrators can assign this role to users in their tenant at any time from the **Administration** tab.

Table 2-12. Tenant Roles and Responsibilities (continued)

Role	Responsibilities	How Assigned
Infrastructure architect To successfully add software components to the design canvas, you must also have business group member, business group administrator, or tenant administrator role access to the target catalog.	■ Create and manage infrastructure blueprint components. ■ Assemble and manage composite blueprints.	Tenant administrators can assign this role to users in their tenant at any time from the **Administration** tab.
XaaS architect	■ Define custom resource types. ■ Create and publish XaaS blueprints. ■ Create and manage resource mappings. ■ Create and publish resource actions.	Tenant administrators can assign this role to users in their tenant at any time from the **Administration** tab.
Software architect To successfully add software components to the design canvas, you must also have business group member, business group administrator, or tenant administrator role access to the target catalog.	■ Create and manage software blueprint components. ■ Assemble and manage composite blueprints.	Tenant administrators can assign this role to users in their tenant at any time from the **Administration** tab.
Container architect	■ Add, edit, and remove container components in a blueprint by using options on the **Design** tab. ■ Add, edit, and remove container network components in a blueprint by using options on the **Design** tab.	Tenant administrators can assign this role to users and groups in their tenant at any time from the **Administration** tab.
Container administrator	Use all available options in the **Containers** tab, including the following tasks: ■ Configure container hosts, placements, and registries ■ Configure container network settings ■ Create container templates	Tenant administrators can assign this role to users and groups in their tenant at any time from the **Administration** tab.
Catalog administrator	■ Create and manage catalog services. ■ Manage catalog items. ■ Assign icons to actions.	Tenant administrators can assign this role to users in their tenant at any time from the **Administration** tab.

Table 2-12. Tenant Roles and Responsibilities (continued)

Role	Responsibilities	How Assigned
Business group manager	■ Add and delete users within the business group. ■ Assign support user roles to users in the business group. ■ Create and manage entitlements for the business group. ■ Request and manage items on behalf of a user in the business group. ■ Assign approval policies for the business group. ■ Monitor resource usage in a business group. ■ Change machine owner.	The tenant administrator designates the business group manager when creating or editing business groups.
Shared access user	■ Use and run actions on the resources that other business group members deploy. ■ Can request a deployment for themself but cannot request a deployment on behalf of another user.	The tenant administrator designates the shared access users when creating or editing business groups.
Approval administrator	■ Create and manage approval policies.	Tenant administrators can assign this role to users in their tenant at any time from the **Administration** tab.
Approver	■ Approve service catalog requests, including provisioning requests or any resource actions.	The tenant administrator or approval administrator creates approval policies and designates the approvers for each policy.
Support user	■ Request and manage service catalog items on behalf of the other members of the business group ■ Change machine owner.	The tenant administrator designates the support user when creating or editing business groups.
Business user	■ Request service catalog items to which they are entitled. ■ Manage their provisioned resources.	The tenant administrator designates the business users who can consume IT services when creating or editing business groups.
Health Consumer	■ Can view test results. ■ Cannot configure, edit, or delete a test.	The IaaS administrator designates privilege to any role..
Security administrator	■ Create a message board allowlist.	Tenant administrators can assign this role to users in their tenant at any time from the **Administration** tab.

Containers User Roles and Access Privileges

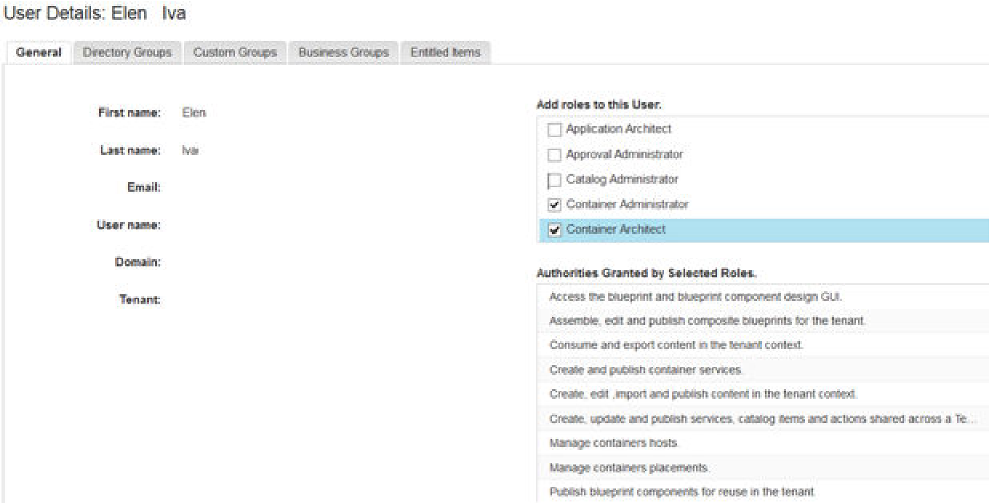
You can use container-specific roles to control who can create and configure containers by using options in the vRealize Automation Containers tab and who can add and configure container components in blueprints by using options in the **Design** tab.

When you enable Containers, two container-specific roles appear in the list of roles that a vRealize Automation tenant administrator can assign to users and groups.

User Role	Description
Container Administrator	Users and groups with this role can see the **Containers** tab in vRealize Automation. They can use all theContainers options, such as configuring hosts, placements, and registries. They can also create templates and provision containers and applications for configuration and validation purposes.
Container Architect	Users and groups with this role can use containers as components when creating and editing blueprints in vRealize Automation. They have permission to see the **Design** tab in vRealize Automation and to work with blueprints.

For related information about vRealize Automation administrator and user roles, see Tenant Roles and Responsibilities in vRealize Automation.

Tenant administrators can assign one or both of these roles to users or groups in their tenant at any time by opening the **Administrator** tab, clicking **Users and Groups > Directories Users and Groups** and clicking on a user name to open the following screen:


User Details: Elen Iva

First name: Elen
Last name: Iva
Email:
User name:
Domain:
Tenant:

Add roles to this User.

- □ Application Architect
- □ Approval Administrator
- □ Catalog Administrator
- ☑ Container Administrator
- ☑ Container Architect

Authorities Granted by Selected Roles.

Access the blueprint and blueprint component design GUI.
Assemble, edit and publish composite blueprints for the tenant.
Consume and export content in the tenant context.
Create and publish container services.
Create, edit ,import and publish content in the tenant context.
Create, update and publish services, catalog items and actions shared across a Te...
Manage containers hosts.
Manage containers placements.
Publish blueprint components for reuse in the tenant.

IaaS administrators automatically inherit the container administrator permissions to perform Containers administrative tasks.

Consumers of catalog items that involve containers inherit the necessary privileges to access the resources provided by the Containers. They can open and see the details of their container-related items and perform day-two operations on them.

vRealize Automation users authenticated through VMware Identity Manager have access to Containers.

vRealize Automation multi-tenancy and business group membership is implemented in Containers.

Service Catalog

The service catalog provides a common interface for consumers of IT services to use to request and manage the services and resources they need.

Requesting and Managing Items in the Catalog

The catalog provides a self-service portal for requesting service deployments and also enables business users to manage their own provisioned resources.

The following example is of a typical life cycle for requesting items from the service catalog and managing their deployments.

Connie, the consumer of IT services, logs in to the vRealize Automation console. On the **Catalog** tab, she finds the catalog item she wants and clicks **Request**.

When Connie clicks **Submit** on the request form, the **Deployments** page opens and she can track the progress of the deployment request, including whether it is pending approval, in progress, or completed. When the request is complete, she can click **Actions > View Details** on the **Deployments** page and perform various actions on the deployment by using the **Actions** drop-down menu. The actions she can perform are based on entitlements and can also be made subject to approval based on flexible approval policies.

Creating and Publishing Catalog Items

Catalog administrators and tenant administrators can define new catalog items and publish them to the service catalog. Tenant administrators and business group managers can entitle the new item to consumers.

Typically, a catalog item provides a complete specification of the resource to be provisioned and the process to initiate when the item is requested. It also defines the options that are available to a requester of the item, such as virtual machine configuration or lease duration, or any additional information that the requester is prompted to provide when submitting the request.

For example, Sean has privileges to create and publish blueprints, including software components and XaaS. After the blueprint is published, Sean, or a catalog administrator or a tenant administrator responsible for managing the catalog, can then configure the catalog item, including specifying an icon and adding the item to a service.

To make the catalog item available to users, a tenant administrator or business group manager must entitle the item to the users and groups who should have access to it in the service catalog.

Services for the Service Catalog

Services are used to organize catalog items into related offerings to make it easier for service catalog users to browse for the catalog items they need.

For example, catalog offerings can be organized into Infrastructure Services, Application Services, and Desktop Services.

A tenant administrator or catalog administrator can specify information about the service such as the service hours, support team, and change window. Although the catalog does not enforce service-level agreements on services, this information is available to business users browsing the service catalog.

Catalog Items

Users can browse the service catalog for catalog items that they are entitled to request.

Some catalog items result in an item being provisioned that the user can manage through its life cycle. For example, an application developer can request storage as a service, then later add capacity, request backups, and restore previous backups.

Other catalog items do not result in provisioned items. For example, a cell phone user can submit a request for additional minutes on a mobile plan. The request initiates a workflow that adds minutes to the plan. The user can track the request as it progresses, but cannot manage the minutes after they are added.

Some catalog items are available only in a specific business group, other catalog items are shared between business groups in the same tenant.

Actions

Actions are operations that you can perform on provisioned items.

Users can manage their provisioned items on the **Deployments** tab. The **Actions > View Details** option is always present for each deployment. Deployment actions are then available by selecting **Actions** on the details page. The available actions are dependent on the type of deployment and the user's entitlements.

Entitlements

Entitlements determine which users and groups can request specific catalog items or perform specific actions. Entitlements are specific to a business group.

Business group managers can create entitlements for the groups that they manage. Tenant administrators can create entitlements for any business group in their tenant. When you create an entitlement, you must select a business group and specify individual users and groups in the business group for the entitlement.

You can entitle an entire service category, which entitles all of the catalog items in that service, including items that are added to the service after you create the entitlement.

You can also add individual catalog items in a service to an entitlement. Services do not contain actions. You must add actions to an entitlement individually by using the **Administration > Catalog Management > Entitlements** menu sequence.

For each service, catalog item, or action that you entitle, you can optionally specify an approval policy to apply to requests for that item. If you entitle an entire service and a specific catalog item in that service in the same entitlement, the approval policy on the catalog item overrides the policy on the service. For example, you can entitle the Cloud Infrastructure service to members of a business group and allow them to request any of its items with no approval policy. For a select number of catalog items that require more governance for their provisioning, you can entitle those in the same entitlement and apply an approval policy on just those items.

The actions that you entitle to users apply to any items that support the entitled action and they are not limited to the services and actions in the same entitlement. For example, if Connie, a consumer of infrastructure services, is entitled to Machine Blueprint 1 and the action Reconfigure in one entitlement, and she is also entitled to Machine Blueprint 2 in a different entitlement, then she is entitled to reconfigure machines provisioned from Machine Blueprint 1 and Machine Blueprint 2, as long as both blueprints allow that action to be performed.

If multiple entitlements exist for the same business group, you can prioritize the entitlements. When a user makes a catalog request, the entitlement and associated approval policy that applies is the highest priority entitlement that grants the user access to that item or action.

Approval Policies

An approval policy is used to govern whether a service catalog user needs approval from someone in your organization to provision items in your environment.

A tenant administrator or approval administrator can create approval policies. The policies can be for pre-provisioning or post-provisioning. If a pre-approval is configured, then the request must be approved before the request is provisioned. If it is a post-approval, the request must be approved before the provisioned item is released to the requesting user.

The policies are applied to items in an entitlement. You can apply them to services, catalog items, catalog item components, or actions that require an approver to approve or reject a provisioning request by using the **Administion > Approval Policies** menu sequence.

When a service catalog user requests an item that includes one or more approval policies, the approval request is sent to the approvers. If approved, the request moves forward. If rejected, the request is canceled and the service catalog user is notified regarding the rejection.

Infrastructure as a Service

With Infrastructure as a Service (IaaS), you can rapidly model and provision servers and desktops across virtual and physical, private and public, or hybrid cloud infrastructures.

- Configuring Infrastructure Fabric

  The IaaS administrator and fabric administrator roles are responsible for configuring the fabric to enable provisioning of infrastructure services. Fabric configuration is system-wide and is shared across all tenants.

- Infrastructure Source Endpoints

  Infrastructure sources can include a group of virtualization compute resources or a cloud service account.

- Compute Resources

  A compute resource is an object that represents a host, host cluster, or pool in a virtualization platform, a virtual datacenter, or an Amazon region on which machines can be provisioned.

- Data Collection

  vRealize Automation collects data from infrastructure source endpoints and their compute resources.

- Fabric Groups

  An IaaS administrator can organize virtualization compute resources and cloud endpoints into fabric groups by type and intent. One or more fabric administrators manage the resources in each fabric group.

- Business Groups

  A business group associates a set of services and resources to a set of users, often corresponding to a line of business, department, or other organizational unit.

- Machine Prefixes

  You use machine prefixes to generate the names of provisioned machines.

- Resource Reservations

  You can create a reservation to allocate provisioning resources to a specific business group.

- Machine Blueprints

  A blueprint that contains a machine component specifies the workflow used to provision a machine and includes information such as CPU, memory, and storage. Machine blueprints specify the workflow used to provision a machine and include additional provisioning information such as the locations of required disk images or virtualization platform objects. Blueprints also specify policies such as the lease period and can include networking and security components such as security groups, policies, or tags.

- Machine Leases and Reclamation

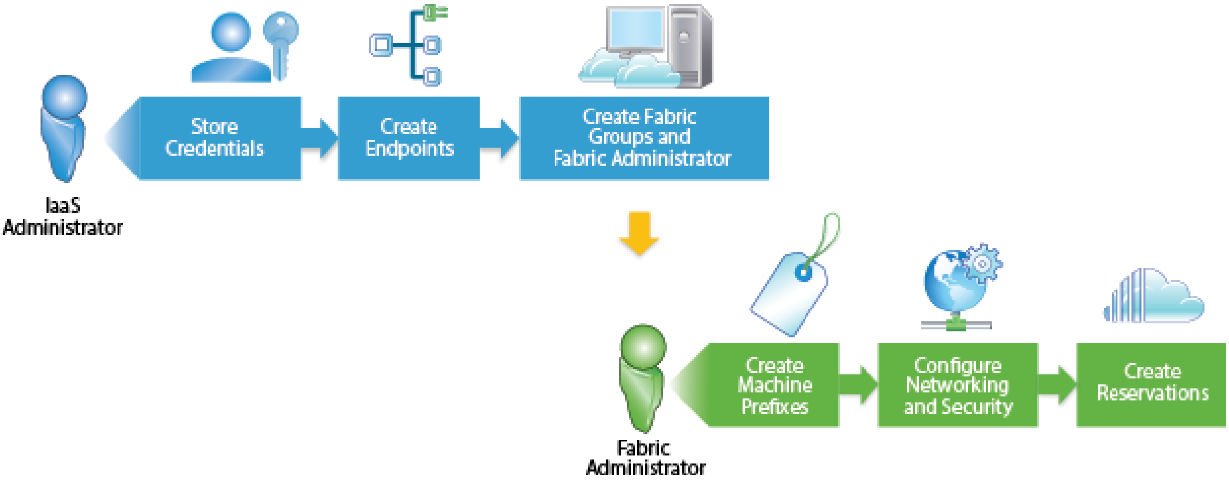
  Machine lease and reclamation options provides mechanisms for controlling resource use and controlling prices.

- Scaling and Reconfiguring Deployments

  You can scale provisioned deployments to adjust to changing workload demands. You use the scale in or scale out actions for horizontal scale, and the machine reconfigure action for vertical scale. You govern scale and reconfigure actions by using entitlements, approval policies, or by designing constraints directly into blueprints.

Configuring Infrastructure Fabric

The IaaS administrator and fabric administrator roles are responsible for configuring the fabric to enable provisioning of infrastructure services. Fabric configuration is system-wide and is shared across all tenants.



An IaaS administrator creates an endpoint to configure access to an infrastructure source. When the connection to an infrastructure source is established, vRealize Automation collects information about the compute resources available through that source. The IaaS administrator can then organize those resources into fabric groups and assign a fabric administrator to manage each group as well as cross-tenant configuration such as machine prefixes.

A fabric administrator can create reservations to allocate provisioning resources in the fabric group to specific business groups that the tenant administrator created during tenant configuration. Optionally, the fabric administrator can configure reservation, network, or storage reservation policies. For example, they can create a reservation policy to control placement of provisioned machines.

When the fabric administrator has created reservations, the IaaS architects can create and publish machine blueprints for reuse in application blueprints and for catalog administrators to make available in the service catalog.

Infrastructure Source Endpoints

Infrastructure sources can include a group of virtualization compute resources or a cloud service account.

An IaaS administrator configures an infrastructure source by specifying the endpoint details and credentials that vRealize Automation can use to communicate with the source.

vRealize Automation collects information about all configured infrastructure sources at regular intervals.

Table 2-13. vRealize Automation Infrastructure Endpoints

Infrastructure Source	Endpoints
vSphere	vCenter Server
vCloud Air	vCloud Air OnDemand or subscription service
vCloud Director	vCloud Director server
Amazon AWS or OpenStack	Cloud service account
Microsoft Azure	Cloud service account
NSX for vSphere or NSX-T	NSX network and security associated with a vSphere infrastructure source
Hyper-V (SCVMM)	Microsoft System Center Virtual Machine Manager server
KVM (RHEV)	Red Hat Enterprise Virtualization server

For information about creating endpoints, see Configuring Endpoints.

Compute Resources

A compute resource is an object that represents a host, host cluster, or pool in a virtualization platform, a virtual datacenter, or an Amazon region on which machines can be provisioned.

An IaaS administrator can add compute resources to or remove compute resources from a fabric group. A compute resource can belong to more than one fabric group, including groups that different fabric administrators manage. After a compute resource is added to a fabric group, a fabric administrator can create reservations on it for specific business groups. Users in those business groups can then be entitled to provision machines on that compute resource.

Information about the compute resources on each infrastructure source endpoint and machines provisioned on each compute resource is collected at regular intervals.

Table 2-14. Examples of Compute Resources for Infrastructure Sources

Infrastructure Source	Compute Resource
vSphere (vCenter)	ESX or ESXi host or cluster
Hyper-V (SCVMM)	Hyper-V host
KVM (RHEV)	KVM host
vCloud Director	virtual datacenter
Amazon AWS	Amazon region

Data Collection

vRealize Automation collects data from infrastructure source endpoints and their compute resources.

Data collection occurs at regular intervals. Each type of data collection has a default interval that you can override or modify. Each type of data collection also has a default timeout interval that you can override or modify.

IaaS administrators can manually initiate data collection for infrastructure source endpoints and fabric administrators can manually initiate data collection for compute resources.

Table 2-15. Data Collection Types

Data Collection Type	Description
Infrastructure Source Endpoint Data Collection	Updates information about virtualization hosts, templates, and ISO images for virtualization environments. Updates virtual datacenters and templates for vCloud Director. Updates Amazon regions and machines provisioned on Amazon regions. Endpoint data collection runs every 4 hours.
Inventory Data Collection	Updates the record of the virtual machines whose resource use is tied to a specific compute resource, including detailed information about the networks, storage, and virtual machines. This record also includes information about unmanaged virtual machines, which are machines provisioned outside of vRealize Automation. Inventory data collection runs every 24 hours. The default timeout interval for inventory data collection is 2 hours.
State Data Collection	Updates the record of the power state of each machine discovered through inventory data collection. State data collection also records missing machines that vRealize Automation manages but cannot be detected on the virtualization compute resource or cloud endpoint. State data collection runs every 15 minutes. The default timeout interval for state data collection is 1 hour.
Performance Data Collection (vSphere compute resources only)	Updates the record of the average CPU, storage, memory, and network usage for each virtual machine discovered through inventory data collection. Performance data collection runs every 24 hours. The default timeout interval for performance data collection is 2 hours.
Network and security inventory data collection (vSphere compute resources only)	Updates the record of network and security data related to vCloud Networking and Security and NSX, particularly information about security groups and load balancing, for each machine following inventory data collection.
WMI data collection (Windows compute resources only)	Updates the record of the management data for each Windows machine. A WMI agent must be installed, typically on the Manager Service host, and enabled to collect data from Windows machines.

Fabric Groups

An IaaS administrator can organize virtualization compute resources and cloud endpoints into fabric groups by type and intent. One or more fabric administrators manage the resources in each fabric group.

Fabric administrators are responsible for creating reservations on the compute resources in their groups to allocate fabric to specific business groups.

Fabric groups are created in a specific tenant, but their resources can be made available to users who belong to business groups in all tenants.

Business Groups

A business group associates a set of services and resources to a set of users, often corresponding to a line of business, department, or other organizational unit.

Business groups are managed by using the **Administration > Users & Groups > Business Groups** menu sequence. They are used when creating reservations and entitling users to items in the service catalog.

To request catalog items, a user must belong to the business group that is entitled to request the item. A business group can have access to catalog items specific to that group and to catalog items that are shared between business groups in the same tenant. Each business group has one or more reservations that determine on which compute resources the machines that this group requested can be provisioned.

A business group must have at least one business group manager, who monitors the resource use for the group and often is an approver for catalog requests. Business groups can include support users. Support users can request and manage machines on behalf of other group members. Business group managers can also submit requests on behalf of their users. A user can be a member of more than one business group, and can have different roles in different groups.

For information about creating a business group, see Create a Business Group.

Machine Prefixes

You use machine prefixes to generate the names of provisioned machines.

You should assign a default machine prefix to every business group that you expect to need infrastructure resources. Every blueprint must have a machine prefix or use the group default prefix.

Only the machine prefixes that are applicable to the current tenant are exposed with authoring a blueprint or editing a business group.

You establish the default machine prefix for the business group by using the **Infrastructure** tab made available by the **Administration > Users & Groups > Business Groups** menu sequence.

Fabric administrators are responsible for managing machine prefixes. A prefix is a base name to be followed by a counter of a specified number of digits. For example, a prefix of g1dw for group1 and developer workstation, with a counter of three digits produces machines named g1dw001, g1dw002, and so on. A prefix can also specify a number other than 1 to start the counter.

If a business group is not intended to provision infrastructure resources, tenant administrators do not need to assign a default machine prefix when they create the business group. If the business group is intended to provision infrastructure resources, tenant administrators should assign one of the existing machine prefixes as the default for the business group. This assignment does not restrict blueprint architects from choosing a different prefix when they create blueprints. A tenant administrator can change the default prefix of a business group at any time. The new default prefix is used in the future, but does not affect previously provisioned machines.

For information about creating machine prefixes, see Configure Machine Prefixes.

Resource Reservations

You can create a reservation to allocate provisioning resources to a specific business group.

A reservation allocates a share of the memory, CPU and storage resources on a particular compute resource or the provisioning services of a cloud service account data center.

A business group can have multiple reservations on the same compute resource or different compute resources, or any number of reservations containing any number of machines.

A compute resource can also have multiple reservations for multiple business groups.

When a user requests a machine, it can be provisioned on any reservation of the appropriate type that has sufficient capacity for the machine. You can apply a reservation policy to a blueprint to restrict the machines provisioned from that blueprint to a subset of available reservations.

For information about creating reservations and reservation policies, see Configuring Reservations and Reservation Policies.

Machine Blueprints

A blueprint that contains a machine component specifies the workflow used to provision a machine and includes information such as CPU, memory, and storage. Machine blueprints specify the workflow used to provision a machine and include additional provisioning information such as the locations of required disk images or virtualization platform objects. Blueprints also specify policies such as the lease period and can include networking and security components such as security groups, policies, or tags.

A machine blueprint typically refers to a blueprint that contains only one machine component and the associated security and networking elements. It can be published as a standalone blueprint and made available to users in the service catalog. However, published machine blueprints also become available for reuse in your design library, and you can assemble multiple machine blueprints, along with Software components and XaaS blueprints, to design elaborate application blueprints for delivering catalog items that include multiple machines, networking and security, software with full life cycle support, and custom XaaS functionality to your users.

An example of a standalone virtual machine blueprint might be one that specifies a Windows developer workstation with one CPU, 2 GB of memory, and a 30 GB hard disk. A standalone cloud machine blueprint might specify a Red Hat Linux web server image in a small instance type with one CPU, 2 GB of memory, and 160 GB of storage.

Blueprints can be specific to a business group or shared among groups in a tenant, depending on the entitlements that are configured for the published blueprint .

You can add custom properties to a machine component in a blueprint to specify attributes of a machine or to override default specifications. You can also add property groups as a convenience for specifying multiple custom properties.

For information about creating blueprints, see Chapter 5 Providing Service Blueprints to Users.

Machine Leases and Reclamation

Machine lease and reclamation options provides mechanisms for controlling resource use and controlling prices.

Machine leases provide access to a machine for a limited period.

Deployment reclamation allows you to identify underused resources and reclaim them from their owners.

Machine Leases

A blueprint can define a lease duration for its provisioned machines.

If a blueprint does not specify a lease period, machines are provisioned from that blueprint with no expiration date. If a blueprint specifies a single value for lease duration, machines are provisioned from that blueprint with an expiration date based on the blueprint lease duration. The expiration date is calculated from the time of the request, not from when the machine is provisioned. Specified lease durations can be up to one year.

If a blueprint specifies a range of possible lease durations, a user can select the desired lease duration within that range when submitting the machine request. Machine requests can be subject to approval based on the requested lease duration only if you use the Always condition.

Enter the lease information in the vRealize Automation blueprint. Lease information specified in an external application is not recognized.

When a machine lease expires, the machine is powered off. When the archive period expires, the machine is destroyed. You can reactivate an archived machine by setting the expiration date to a date in the future to extend its lease, and powering it back on.

You can send notification emails to alert machine owners and business group managers that a machine's lease is about to expire and again when the lease expires. See Customize the Date for Email Notification for Machine Expiration and Configuring Templates for Automatic IaaS Emails.

Users can be entitled to request a lease extension at any time before it expires. A business group manager or support user can also change the expiration date for a machine after it is provisioned.

Reclamation Overview

You can use metrics to identify underused machines that might be candidates for deployment reclamation.

You can use the basic metrics provided by vRealize Automation to sort and filter metrics information for all of your machines, or you can configure a vRealize Operations Manager endpoint to provide metrics and health badges for your vSphere virtual machines.

Select the candidate deployment and send a reclamation request to the owners of the machines. The machine owner has a fixed period of time to respond to the request. If machines in the deployment are still in use, the machine owner can stop the reclamation process and continue using the machine. If the machine is no longer needed, the owner can release the machine for reclamation, in which case the machine lease is ended. If the owner does not respond in a timely manner, a lease determined by the administrator is imposed. If the owner continues to take no action, the machine is powered off on the new expiration date, the machine is reclaimed, and the resources are freed.

Scaling and Reconfiguring Deployments

You can scale provisioned deployments to adjust to changing workload demands. You use the scale in or scale out actions for horizontal scale, and the machine reconfigure action for vertical scale. You govern scale and reconfigure actions by using entitlements, approval policies, or by designing constraints directly into blueprints.

Scale In or Scale Out

After you provision a deployment, you can adjust to changing workload demands by increasing or decreasing the number of instances of virtual or cloud machines in your deployment. For example, you deployed a three-tiered banking application with a clustered application server node, a database node, and a load balancer node. Demand increases, and you find that the two instances of your application server node cannot handle all the traffic. Because your blueprint supports up to ten instances of the application server, and you are entitled to scale actions, you can scale out your application. You navigate to your provisioned application item in vRealize

Automation and select the scale out action to add another instance of your application server node to the deployment. vRealize Automation provisions a new machine, installs the application software component, and updates your load balancer so your application can handle the increased demands.

If demand decreases, you can scale the deployment in. The newest machines and software components are destroyed first, and your networking and security components are updated so that your deployed application isn't using any unnecessary resources.

Table 2-16. Support for Scalable Components

Component Type	Supported	Notes
Machine components	Yes	Scale out provisions additional instances of your machines, and scale in destroys machines in last in, first out order.
Software components	Yes	Software components are provisioned or destroyed along with machines that are scaled, and the update life cycle scripts are run for any software components that depend on the scaled machine components.
Networking and security components	Yes	Networking and security components, including NSX load balancers, security groups and security tags, are updated for the new deployment configuration. Scaling impacts the network and security, including load balancer, settings for the deployment. When you scale in or scale out a deployment that contains one or more nodes, the associated NSX networking components are updated. For example, if there is an on-demand NAT networking component associated with the deployment, the NAT rules are updated in accordance with the scaling request. When you scale in or scale out a deployment that contains an associated load balancer, the load balancer is automatically configured to include newly added machines or to stop load balancing machines that are targeted for tear down. When you scale out a deployment that contains a load balancer, secondary IP addresses are added to the load balancer. Depending on whether you scale in or scale out, virtual machines are added or removed from the load balancer and saved or removed in the IaaS database.
XaaS components	Yes	XaaS components that are marked as scalable and that have lifecycle workflows assigned can be scaled in and out. You can specify the number of instances. For more information, see Add an XaaS Blueprint.
Nested blueprints	Yes	Supported components in nested blueprints might only update if you create explicit dependencies to scaled machine components. You create explicit dependencies by drawing dependency lines on the design canvas.

When you scale out a deployment, vRealize Automation allocates the requested resources on the current reservation before proceeding. If the scale is partially successful, and fails to provision one or more items against those allocated resources, the resources are not deallocated and do not become available for new requests. Resources that are allocated but unused because of a scale failure are known as dangling resources. You can try to repair partially successful scale operations by attempting to scale the deployment again. However, you cannot scale a deployment to its current size, and fixing a partially successful scale this way does not deallocate the dangling resources. You can view the request execution details screen and find out which tasks failed on which nodes to help you decide whether to fix the partially successful scale with another scale operation. Failed and partially successful scale operations do not impact the functionality of your original deployment, and you can continue to use your catalog items while you troubleshoot any failures.

For a clustered deployment, in which the deployment created from a blueprint contains more than one VM, scaling fails if the blueprint uses a hostname custom property but does not contain a machine prefix value. To avoid this issue, you can use the machine prefix option in the blueprint definition. Otherwise, the scaling function attempts to use the same hostname setting for each VM in the cluster. For more information, see VMware Knowledge Base article 2148213 at http://kb.vmware.com/kb/2148213.

Scale Up or Scale Down By Using Reconfigure

After you provision a vSphere, vCloud Air, or vCloud Director virtual or cloud machine you can adjust to changing workload demands by requesting a machine reconfigure to increase (scale up) or decrease (scale down) machine resource specifications for CPU, memory, storage, or networks. You can also add, edit, or remove custom properties and change descriptions. You can request to reconfigure machines for scale up or scale down that are in the On or Off state.

When you reconfigure a virtual or cloud machine for scale up, vRealize Automation allocates the requested resources on the current reservation before proceeding. If the resources are not available, the machine reconfigure fails. If a machine reconfigure request fails, any resources allocated for scale up are deallocated and available for new requests. When you reconfigure a virtual or cloud machine for scale down, resources are not made available to new requests unless the reconfigure finishes successfully.

Table 2-17. Required Entitlements for Machine Reconfigure for Scaling Scenarios (vSphere, vCloud Air, and vCloud Director only

Virtual or Cloud Machine Owner wants to...	Required Entitlements
Run the reconfigure for scaling immediately after any required approvals are given.	Reconfigure
Specify a date and time to run the reconfiguration for scaling.	Reconfigure
Reschedule a reconfigure for scaling because the request was not approved until after the scheduled time.	Reconfigure
Retry a failed reconfigure request.	Execute reconfigure
Cancel a failed reconfigure request.	Cancel reconfigure
Cancel a scheduled reconfigure request.	Cancel reconfigure

XaaS Blueprints and Resource Actions

XaaS architects can use the XaaS options to create blueprints and publish them to the service catalog. They can also create and publish post-provisioning operations that the consumers can perform on provisioned items.

Creating XaaS Blueprints and Actions

By using the XaaS blueprints and resource actions, you define new provisioning, request, or action offerings and publish them to the common catalog as catalog items.

You can create XaaS blueprints and actions for either requesting or provisioning. The XaaS blueprints for requesting do not provision items and provide no options for post-provisioning operations. Examples of XaaS blueprints for requesting include blueprints for sending emails, generating reports, performing complex calculations, and so on. For an XaaS blueprint, the result is a provisioned item. You can create a custom resource so that you can access and manage the items on the **Deployments** tab.

To define the XaaS specification, you create a blueprint and publish it as a catalog item. After you publish a catalog item, you must include it in a service category. You can use an existing service or create one. A tenant administrator or business group manager can entitle the whole service or only the catalog item to specific users.

If you created a custom resource for a provisioned item, you can create resource actions to define the post-provisioning operations that the consumers can perform. You can also create resource actions for an item that is provisioned by a source different from the XaaS blueprints, for example by IaaS. For this purpose, first you must create a resource mapping to define the type of the catalog item.

To learn more, see Creating XaaS Blueprints and Resource Actions.

Custom Resources

You must create a custom resource so that you can create an XaaS blueprint for provisioning with the option to access and manage the provisioned items. Custom resources define the items for provisioning, and you can use them to define post-provisioning operations that the consumers can perform.

You create a custom resource to define a new type of provisioned item and map it to an existing vRealize Orchestrator object type. vRealize Orchestrator object types are the objects exposed through the APIs of the vRealize Orchestrator plug-ins. The custom resource is the output type of a blueprint workflow for provisioning and can be the input type for a resource action workflow.

For example, if you have a running vCenter Server instance, and you also have the vCenter Server plug-in that is configured to work with vRealize Orchestrator, all of the object types from the vCenter Server API are exposed in vRealize Orchestrator. The vCenter Server plug-in exposes the vSphere inventory objects in the vRealize Orchestrator inventory. The vSphere inventory objects include data centers, folders, ESXi hosts, virtual machines and appliances, resource pools, and so on. You can perform operations on these objects. For example, you can create, clone, or destroy virtual machines.

For more information about the vRealize Orchestrator object types exposed through the vCenter Server API, see the *vCenter Server Plug-In API Reference for vCenter Orchestrator*.

Resource Mappings

You create resource mappings between the vRealize Automation catalog resource type and the vRealize Orchestrator inventory type to manage resources provisioned outside of XaaS.

For example, you might want to create an action so that users can take a snapshot of their Amazon machines. For this action to work on an Amazon machine provisioned, the three components involved, XaaS, vRealize Orchestrator, and IaaS, need a common language You create that common language by adding a resource mapping in XaaS that runs a vRealize Orchestrator scripting action or workflow to map the IaaS Cloud Machine resource type to the vRealize Orchestrator AWS:EC2Instance inventory type.

vRealize Automation provides resource mappings, and the underlying vRealize Orchestrator script actions and workflows, for vSphere, vCloud Director, and vCloud Air machines.

XaaS Blueprints

An XaaS blueprint is a complete specification of a resource.

With XaaS blueprints, you publish predefined and custom vRealize Orchestrator workflows as catalog items for either requesting or provisioning. Blueprints for requesting run workflows with no provisioning and provide no options for managing a provisioned item. Before you create a blueprint for provisioning, you must map the workflow output parameter as a custom resource. Then you can assign resource actions that define post-provisioning operations.

Resource Actions

You can create custom resource actions to configure the post-provisioning operations that the consumers can perform.

To create post-provisioning operations, you must publish vRealize Orchestrator workflows as resource actions. To create a resource action for an item provisioned by using XaaS, you use a custom resource as an input parameter for the workflow. To create a resource action for an item that is provisioned by a source different from XaaS, you use a resource mapping as an input parameter for the workflow. When you entitle the resource actions, they appear in the **Actions** drop-down menus of the provisioned items on the **Deployments** tab.

Common Components

vRealize Automation includes several common components in addition to the service catalog and catalog item sources such as Infrastructure as a Service and XaaS.

Notifications

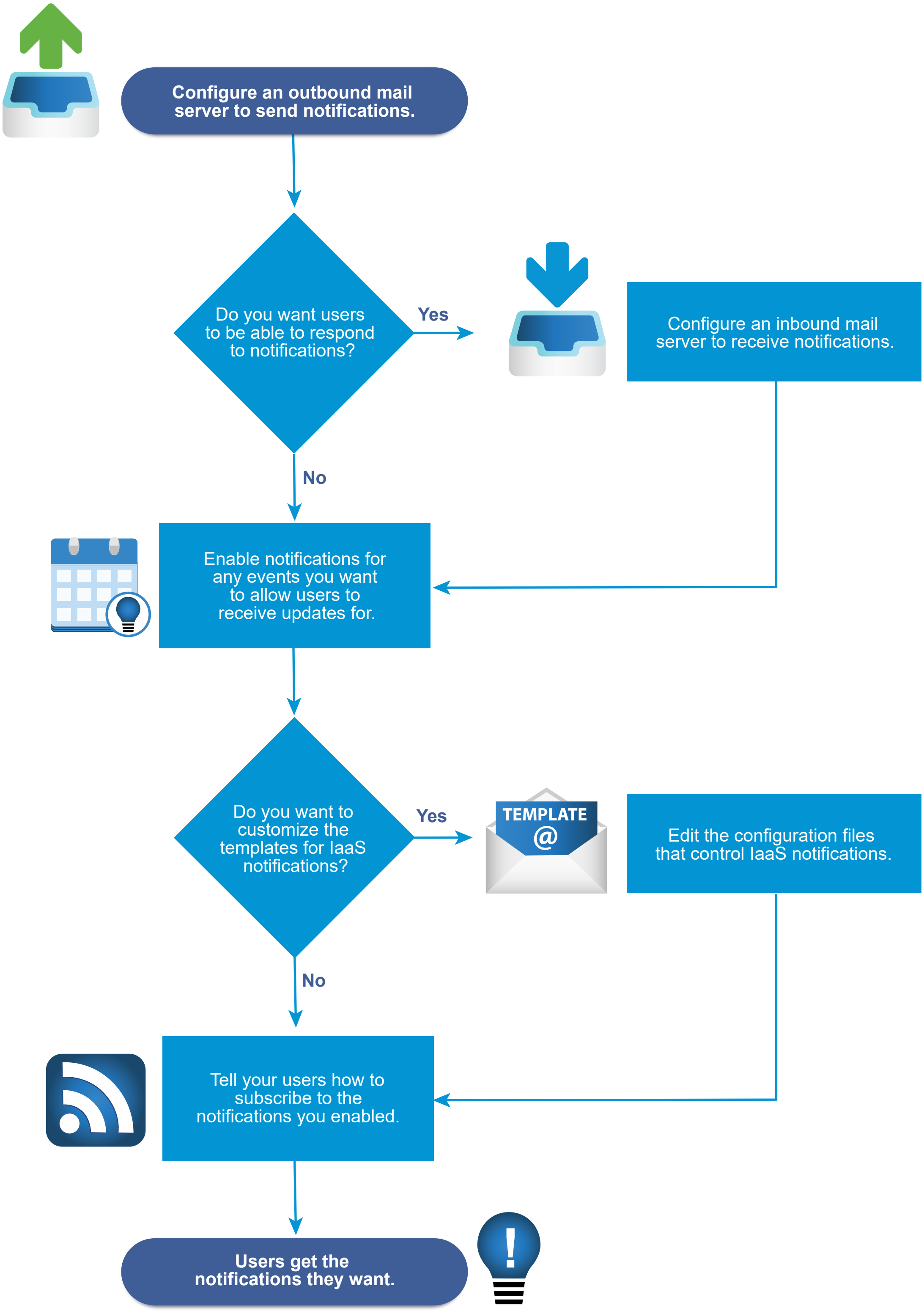
You can send automatic notifications for several types of events, such as the successful completion of a catalog request or a required approval.

Tenant administrators select which events cause notifications to be sent to users in their tenants by using the **Administration > Notifications** menu sequence.

Each user can choose whether to receive notifications. Users either receive all notifications configured by the tenant administrator or no notifications, they do not have fine-grained control over which notifications to receive.

System administrators can configure global email servers that process email notifications. Tenant administrators can override the system default servers, or add their own servers if no global servers are specified.

Some emails have links that users can use to respond to the notification. For example, a notification about a request that requires approval can have one link for approving the request and one for rejecting it.


Configure an outbound mail server to send notifications.
Do you want users to be able to respond to notifications?
Yes
Configure an inbound mail server to receive notifications.
No
Enable notifications for any events you want to allow users to receive updates for.
Do you want to customize the templates for IaaS notifications?
Yes
TEMPLATE
@
Edit the configuration files that control IaaS notifications.
No
Tell your users how to subscribe to the notifications you enabled.
Users get the notifications they want.
!

For more information about notifications, see Checklist for Configuring Notifications.

Branding

Each tenant can change the appearance of the vRealize Automation console and login pages.

System administrators control the default branding for all tenants by using the **Administration > Branding** menu sequence.

A tenant administrator can change the branding of the portal including the login pages, logo, the background color, and the information in the header and footer using that same menu sequence.

If the branding for a tenant is changed, a tenant administrator can always revert back to the system defaults.

For more information about branding, Configuring Custom Branding.

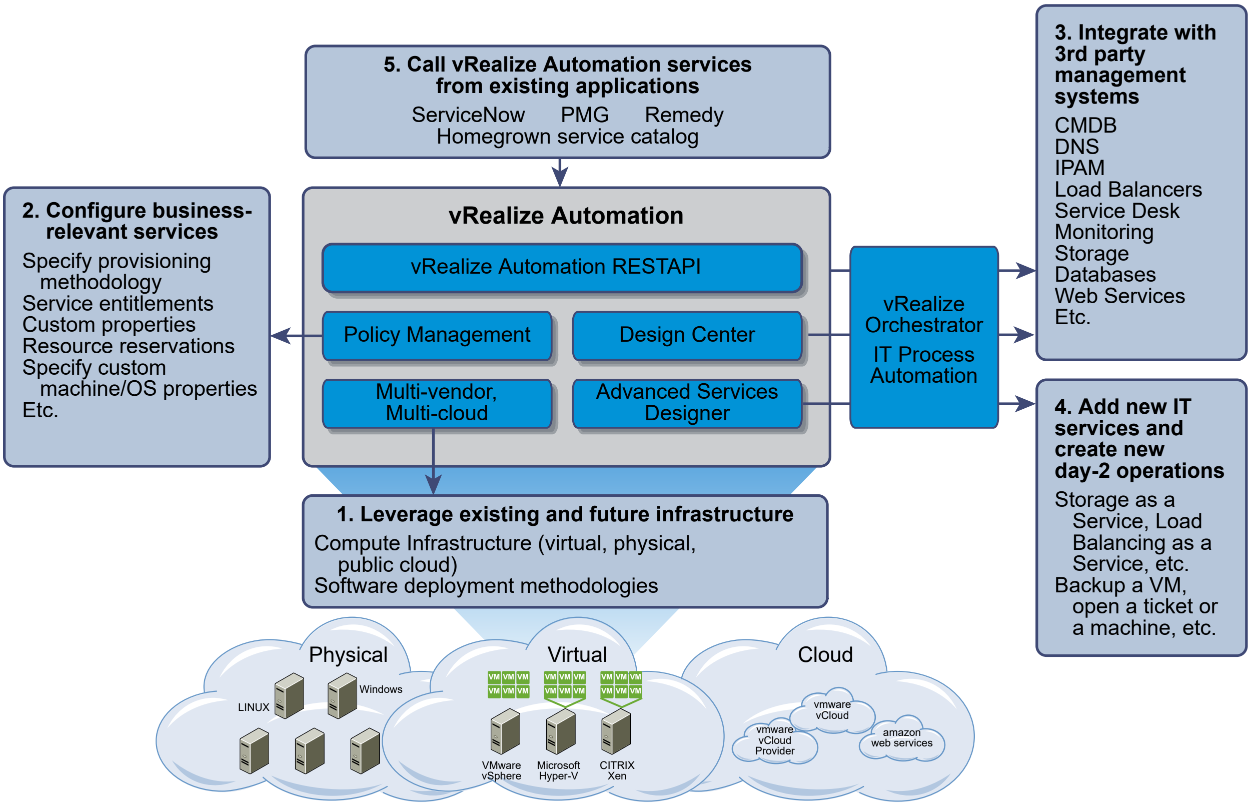
Life Cycle Extensibility

The architecture of vRealize Automation is designed with extensibility in mind. To satisfy different extensibility use cases, vRealize Automation offers a variety of configuration options and tools.

In addition to these extensibility topics, more information is available in the configuration section of the product documentation at Chapter 7 Life Cycle Extensibility.

vRealize Automation Extensibility Options

vRealize Automation is a flexible cloud management platform that enables customization and extensibility at multiple levels.


5. Call vRealize Automation services from existing applications
ServiceNow PMG Remedy
Homegrown service catalog
2. Configure business-relevant services
Specify provisioning methodology
Service entitlements
Custom properties
Resource reservations
Specify custom machine/OS properties
Etc.
vRealize Automation
vRealize Automation RESTAPI
Policy Management
Design Center
Multi-vendor, Multi-cloud
Advanced Services Designer
vRealize Orchestrator
IT Process Automation
3. Integrate with 3rd party management systems
CMDB
DNS
IPAM
Load Balancers
Service Desk
Monitoring
Storage
Databases
Web Services
Etc.
4. Add new IT services and create new day-2 operations
Storage as a Service, Load Balancing as a Service, etc.
Backup a VM, open a ticket or a machine, etc.
1. Leverage existing and future infrastructure
Compute Infrastructure (virtual, physical, public cloud)
Software deployment methodologies
Physical
LINUX
Windows
Virtual
VM
VMware vSphere
Microsoft Hyper-V
CITRIX Xen
Cloud
vmware vCloud
vmware vCloud Provider
amazon web services

Leveraging Existing and Future Infrastructure

vRealize Automation provides support for many types of infrastructure and provisioning methods.

IaaS administrators can integrate with several infrastructure sources including virtual hypervisors, such as vSphere, Hyper-V, KVM (RHEV), and so on, public clouds including VMware vCloud ® Air ™ and Amazon AWS, and physical infrastructure.

Blueprint authors can control many machine options, including provisioning methods, by configuring blueprints for various types of infrastructure.

For a full list of supported infrastructure types and provisioning methods, see *vRealize Automation Support Matrix*. For information about configuring infrastructure blueprints, see *Configuring vRealize Automation*.

Configuring Business-Relevant Services

The vRealize Automation console enables administrators to configure business- and user-specific policies through a web-based user interface without writing any code.

These business policies include entitlements and approvals for the service catalog, resource reservation policies for infrastructure, and many others.

For information about customization tasks that you can perform through the vRealize Automation console, see Designing Blueprints.

Using custom properties, machine blueprint authors can define additional machine properties or override their standard attributes for a variety of purposes.

For details about the use and configuration of custom properties, see Managing the Service Catalog.

Extending vRealize Automation with Event-Based Workflows

You can use workflow subscriptions to run vRealize Orchestrator workflows based on events.

vRealize Automation provides event topics to which you can subscribe, triggering your custom vRealize Orchestrator workflows when an IaaS resource is provisioned or modified.

Integrating with Third-Party Management Systems

Provisioning or decommissioning a new machine, especially for mission-critical systems, typically requires interacting with a number of different management systems, including DNS servers, load balancers, CMDBs, IP Address Management and other systems.

Administrators can inject custom logic (known as workflows) at various predetermined IaaS life cycle stages. These IaaS workflows can call out to vRealize Orchestrator for bi-directional integration with external management systems.

Adding New IT Services and Creating New Actions

The XaaS enables XaaS architects to define new services and new management operations on provisioned resources.

vRealize Automation provides a range of management operations that you can perform on machines. Your organization may find it valuable to extend the default IaaS machine menus with new options, such as creating a machine backup or running a security check.

It can also be beneficial to expose entirely new services in the service catalog, so that users can automate other initiatives directly via the portal. Service architects can create XaaS blueprints for storage-as-a-service, networking services or virtually any kind of IT service by using XaaS.

For details about how to create new catalog items, see Designing XaaS Blueprints and Resource Actions.

Calling vRealize Automation Services from External Applications

In some cases, organizations may want to interact with vRealize Automation programmatically rather than via the vRealize Automation console.

For such scenarios, the vRealize Automation API provides a standardized, secured RESTful interface for cloud access and interaction, controlled through business-aware policy for consumers such as users, infrastructure, devices, and applications.

All blueprints, including the ones created via the XaaS, are automatically exposed through the vRealize Automation API.

Distributed Execution

All core vRealize Automation workflows are executed in a distributed execution environment.

The vRealize Automation runtime environment consists of one or more DEM Worker instances that can execute any workflow installed in the core engine. Additional Worker instances can be added as needed for scalability, availability and distribution.

Skills can be used to associate DEMs and workflows, restricting execution of a given workflow to a particular DEM or set of DEMs with matching skills. Any number and combination of skills can be associated with a given workflow or DEM. For example, workflow execution can be restricted to a specific datacenter, or to environments that support a specific API the workflow requires. The vRealize Automation Designer and the CloudUtil command-line tool provide facilities for mapping skills to DEMs and workflows.

For more information about distributed execution and working with skills, see *Life Cycle Extensibility*.

External Preparations for Blueprint Provisioning

3

You may need to create or prepare some elements outside of vRealize Automation to support catalog item provisioning. For example, if you want to provide a catalog item for provisioning a clone machine, you need to create a template on your hypervisor to clone from.

This chapter includes the following topics:

Preparing Your Environment for vRealize Automation Management

Depending on your work environment, you might need to make some configuration changes before you can bring your environment under vRealize Automation management, or before you can leverage certain features.

Table 3-1. Preparing Your Environment for vRealize Automation Integration

Environment	Preparations
NSX for vSphere and NSX-T	If you want to leverage NSX for vSphere or NSX-T to manage network, security, and load balancer features of VMs provisioned with vRealize Automation, prepare your NSX instance for integration. See Checklist for Preparing NSX Network and Security Configuration.
vCloud Director	Install and configure your vCloud Director instance, set up your vSphere and cloud resources, and identify or create appropriate credentials to provide vRealize Automation with access to your vCloud Director environment. See Preparing Your vCloud Director Environment for vRealize Automation .
vCloud Air	Register for your vCloud Air account, set up your vCloud Air environment, and identify or create appropriate credentials to provide vRealize Automation with access to your environment. See Preparing for vCloud Air and vCloud Director Provisioning.
Amazon Web Services	Prepare elements and user roles in your Amazon Web Services environment for use in vRealize Automation, and understand how Amazon Web Services features map to vRealize Automation features. See Preparing Your Amazon Web Services Environment.
Microsoft Azure	Configure networking to use VPN tunneling to support Software components on Azure blueprints. See Configure Network-to-Azure VPC Connectivity.
Red Hat OpenStack	If you want to leverage Red Hat OpenStack to manage networking and security features of machines provisioned with vRealize Automation, prepare your Red Hat OpenStack instance for integration. See Preparing Red Hat OpenStack Network and Security Features.
SCVMM	Configure storage, networking, and understand template and hardware profile naming restrictions. See Preparing Your SCVMM Environment.

Table 3-1. Preparing Your Environment for vRealize Automation Integration (continued)

Environment	Preparations
External IPAM Providers	Register an external IPAM provider package or plug-in, run the configuration workflows, and register the IPAM solution as a new vRealize Automation endpoint. See Checklist For Providing Third-Party IPAM Provider Support.
All other environments	You do not need to make changes to your environment. You can begin preparing for machine provisioning by creating templates, boot environments, or machine images. See Preparing for Machine Provisioning.

Checklist for Preparing NSX Network and Security Configuration

Before you can use NSX network and security options in vRealize Automation, you must configure the external NSX for vSphere or NSX-T network and security environment that you intend to use.

To use XaaS to extend your vRealize Automation and NSX for vSphere integration, install the NSX plug-in in vRealize Orchestrator. The plug-in does not support NSX-T.

In preparation for using NSX network, security, and load balancing capabilities in vRealize Automation, when using NSX Manager credentials you must use the NSX Manager administrator account.

vRealize Automation supports NSX for vSphere and NSX-T. For related information about your NSX application, see NSX for vSphere product documentation or NSX-T product documentation.

Much of the NSX network and security settings that you use in vRealize Automation is configured externally and made available after data collection is run on the compute resources.

For information about NSX settings that you can configure for vRealize Automation blueprints, see Configuring Network and Security Component Settings in vRealize Automation.

Table 3-2. Preparing NSX Networking and Security Checklist

Task	Location	Details
❑ Configure NSX network settings, including gateway and transport zone settings.	Configure network settings in your NSX application.	Depending on your NSX product, see administration topics in the following NSX documentation: ■ NSX for vSphere product documentation ■ NSX-T product documentation
❑ Create NSX security policies, tags, and groups.	Configure security settings in your NSX application.	Depending on your NSX product, see administration topics in the following NSX documentation: ■ NSX for vSphere product documentation ■ NSX-T product documentation

Table 3-2. Preparing NSX Networking and Security Checklist (continued)

Task	Location	Details
❑ Configure NSX load balancer settings.	Configure an NSX load balancer settings in your NSX application.	Depending on your NSX product, see administration topics in the following NSX documentation: ■ NSX for vSphere product documentation ■ NSX-T product documentation Also see Custom Properties for Networking and Security.
❑ For cross-virtual center deployments in NSX for vSphere, verify that the compute NSX manager has the primary NSX manager role.	vRealize Automation provisioning requires that the compute NSX manager for the region in which the machines reside has the primary NSX manager role.	See Administrator Requirements for Provisioning NSX for vSphere Universal Objects. See information about cross-virtual center deployment, universal objects, and the primary NSX manager role in NSX for vSphere product documentation .

Install the NSX Plug-In on vRealize Orchestrator

Installing the NSX plug-in requires that you download the vRealize Orchestrator installer file, use the vRealize Orchestrator Configuration interface to upload the plug-in file, and install the plug-in on a vRealize Orchestrator server.

For general plug-in update and troubleshooting information, see vRealize Orchestrator product documentation.

Prerequisites

To use XaaS to extend your vRealize Automation and NSX for vSphere integration, install the NSX plug-in in vRealize Orchestrator. The plug-in does not support NSX-T.

If you are using an embedded vRealize Orchestrator that already contains an installed NSX plug-in, you can skip this procedure.

- Verify that you are running a supported vRealize Orchestrator instance.

  For information about setting up vRealize Orchestrator, see *Installing and Configuring VMware vRealize Orchestrator* in vRealize Orchestrator product documentation.

- Verify that you have credentials for an account with permission to install vRealize Orchestrator plug-ins and to authenticate through vCenter Single Sign-On.
- Verify that you installed the vRealize Orchestrator client and that you can log in with administrator credentials.
- Confirm the correct version of the NSX plug-in in the vRealize Automation support matrix.

Procedure

1. Download the plug-in file to a location accessible from the vRealize Orchestrator server.

   The plug-in installer file name format, with appropriate version values, is `o11nplugin-nsx-1.n.n.vmoapp`. Plug-in installation files for NSX for vSphere are available from the VMware product download site.

2. Open a browser and start the vRealize Orchestrator configuration interface.

   An example of the URL format is https://*orchestrator_server*.com:8283.

3. Click **Plug-Ins** in the left pane and scroll down to the Install new plug-in section.
4. In the **Plug-In file** text box, browse to the plug-in installer file and click **Upload and install**.

   The file must be in `.vmoapp` format.

5. At the prompt, accept the license agreement in the Install a plug-in pane.
6. In the Enabled plug-ins installation status section, confirm that the correct NSX plug-in name is specified.

   For version information, see the vRealize Automation support matrix.

   The status `Plug-in will be installed at next server startup`, appears.

7. Restart the vRealize Orchestrator server service.
8. Restart the vRealize Orchestrator configuration interface.
9. Click **Plug-Ins** and verify that the status changed to `Installation OK`.
10. Start the vRealize Orchestrator client application, log in, and use the **Workflow** tab to navigate through the library to the `NSX` folder.

    You can browse through the workflows that the NSX plug-in provides.

What to do next

Create a vRealize Orchestrator endpoint in vRealize Automation to use for running workflows. See Create a vRealize Orchestrator Endpoint.

Administrator Requirements for Provisioning NSX for vSphere Universal Objects

To provision machines in a cross vCenter NSX environment when using NSX universal objects, you must provision to a vCenter Server in which the NSX compute manager has the primary role.

In a cross vCenter NSX for vSphere environment, you can have multiple vCenter servers, each of which must be paired with its own NSX manager. One NSX manager is assigned the role of primary NSX manager, and the others are assigned the role of secondary NSX manager.

The primary NSX manager can create universal objects, such as universal logical switches. These objects are synchronized to the secondary NSX managers. You can view these objects from the secondary NSX managers, but you cannot edit them there. You must use the primary NSX manager to manage universal objects. The primary NSX manager can be used to configure any of the secondary NSX managers in the environment.

For more information about the NSX cross-vCenter environment, see *Overview of Cross-vCenter Networking and Security* in the *NSX Administration Guide* in the NSX for vSphere product documentation.

For a vSphere (vCenter) endpoint that is associated to the NSX endpoint of a primary NSX manager, vRealize Automation supports NSX local objects, such as local logical switches, local edge gateways, and local load balancers, security groups, and security tags. It also supports NAT one-to-one and one-to-many networks with universal transport zone, routed networks with universal transport zone and universal distributed logical routers (DLRs), and a load balancer with any type of network.

vRealize Automation does not support NSX existing and on-demand universal security groups or tags.

To provision local on-demand networks as the primary NSX manager, use a vCenter-specific local transport zone. You can configure vRealize Automation reservations to use the local transport zone and virtual wires for deployments in that local vCenter Server.

If you connect a vSphere (vCenter) endpoint to a corresponding secondary NSX manager endpoint, you can only provision and use local objects.

vRealize Automation can consume an NSX universal logical switch as an external network. If a universal switch exists, it is data-collected and then attached to or consumed by each machine in the deployment.

- Provisioning an on-demand network to a universal transport zone can create a new universal logical switch.
- Provisioning an on-demand network to a universal transport zone on the primary NSX manager creates a universal logical switch.

- Provisioning an on-demand network to a universal transport zone on a secondary NSX manager fails, as NSX cannot create a universal logical switch on a secondary NSX manager.

See the VMware Knowledge Base article *Deployment of vRealize Automation blueprints with NSX objects fail (2147240)* at http://kb.vmware.com/kb/2147240 for more information about NSX universal objects.

Checklist For Providing Third-Party IPAM Provider Support

You can obtain IP addresses and ranges for use in network profile definition from a supported third-party IPAM provider, such as Infoblox.

Before you can create and use an external IPAM provider endpoint in a vRealize Automation network profile, you must download or otherwise obtain a vRealize Orchestrator IPAM provider plug-in or package, import the plug-in or package and run required workflows in vRealize Orchestrator, and register the IPAM solution as a vRealize Automation endpoint.

For an overview of the provisioning process for using an external IPAM provider to supply a range of possible IP addresses, see Provisioning a vRealize Automation Deployment Using a Third-Party IPAM Provider.

Table 3-3. Preparing for External IPAM Provider Support Checklist

Task	Description	Details
❑ Obtain and import the supported external IPAM Provider vRealize Orchestrator plug-in.	Download the IPAM provider plug-in or package, for example The Infoblox IPAM Plug-in for vRealize Orchestrator plug-in and supporting documentation, from the VMware Solution Exchange (https://solutionexchange.vmware.com/store/category_groups/cloud-management) and import the plug-in or package to vRealize Orchestrator. If the VMware Solution Exchange does not contain the IPAM provider package that you need, you can create your own by using a third-party IPAM Solution Provider SDK and supporting documentation. A vRealize Automation version-specific third-party IPAM Solution Provider SDK, supporting documentation, and associated starter package for vRealize Orchestrator and vRealize Automation is available at https://code.vmware.com/sdks or https://code.vmware.com/samples.	See Obtain and Import a Third-Party IPAM Provider Package in vRealize Orchestrator.
❑ Run the required configuration workflows and register the external IPAM solution as a vRealize Automation endpoint.	Run the vRealize Orchestrator configuration workflows and register the IPAM provider endpoint type in vRealize Orchestrator.	See Run Workflow to Register Third-Party IPAM Endpoint Type in vRealize Orchestrator.

Obtain and Import a Third-Party IPAM Provider Package in vRealize Orchestrator

To prepare to define and use an third-party IPAM provider endpoint, you must first obtain the third-party IPAM provider package and import the package in vRealize Orchestrator.

You can download and use an existing third-party IP Address Management provider plug-in, such as Infoblox IPAM. You can also create your own third-party IPAM plug-in or package by using a VMware-supplied starter package and accompanying SDK documentation for use with another third-party IPAM solution provider, such as BlueCat.

- Obtain the existing Infoblox IPAM Plug-in for vRealize Orchestrator plug-in and supporting documentation from marketplace.vmware.com. The download also contains documentation for installing and using the plug-in.
- Create your own third-party IPAM solution by obtaining and using a third-party IPAM Solution Provider SDK, supporting documentation, and an associated starter package for vRealize Orchestrator and vRealize Automation. See the vRealize Automation Example Third-Party IPAM Package page at code.vmware.com/web/sdk.

After you import the third-party IPAM provider plug-in or package in vRealize Orchestrator, you must run the required workflows, and register the IPAM endpoint type in vRealize Orchestrator.

For more information about importing plug-ins and packages and running vRealize Orchestrator workflows, see *Using the VMware vRealize Orchestrator Client*. For more information about extending vRealize Automation with vRealize Orchestrator plug-ins, packages, and workflows, see *Life Cycle Extensibility*.

This step sequence uses the Infoblox IPAM plug-in as an example. Your step sequence may differ depending on your vRealize Automation or plug-in version.

Prerequisites

- Download the package or plug-in from marketplace.vmware.com.
- Log in to vRealize Orchestrator with administrator privileges for importing, configuring, and registering a vRealize Orchestrator plug-in or package.

Procedure

1 Open the marketplace.vmware.com site.

2 Locate and download the plug-in or package.

   For example, import the Infoblox plug-in that supports the Infoblox third-party IPAM endpoint in vRealize Orchestrator and vRealize Automation 7.1 and later.

   a In the **Publisher** category, select **Infoblox** and click **Apply**.

   b Select The Infoblox Plug-in for vRealize Orchestrator.

   c Click **Tech Specs** and review the prerequisites.

   d Click **Try** for additional information and to receive an email that contains a link to the download.

   e Download the zip file as specified in the emailed instructions.

     Version 4.0 and greater of the plug-in supports vRealize Automation 7.1 and greater. The zip file also contains documentation about the plug-in.

3 In vRealize Orchestrator, click the **Administrator** tab and click **Import package**.

4 Select the package to import.

5 Select all workflows and artifacts and click **Import selected elements**.

What to do next

Run Workflow to Register Third-Party IPAM Endpoint Type in vRealize Orchestrator.

Run Workflow to Register Third-Party IPAM Endpoint Type in vRealize Orchestrator

Run the registration workflow in vRealize Orchestrator to support vRealize Automation use of the third-party IPAM provider and register the IPAM endpoint type for use in vRealize Automation.

For more information about importing packages and running workflows, see *Using the VMware vRealize Orchestrator Client* in your vRealize Automation release documentation. For more information about extending vRealize Automation with vRealize Orchestrator packages and workflows, see Machine Extensibility Overview.

Prerequisites

- Obtain and Import a Third-Party IPAM Provider Package in vRealize Orchestrator .
- Verify that you are logged in to vRealize Orchestrator with the authority to run registration workflows.
- Be prepared to enter the vRealize Automation administrator credentials when prompted by the registration workflow. When you register IPAM endpoint types in vRealize Orchestrator, you are prompted to enter vRealize Automation administrator credentials.

Procedure

1 In vRealize Orchestrator, click the **Design** tab, select **Administrator > Library**, and select **IPAM Service Package SDK**.

  Each IPAM provider package is uniquely named and contains unique workflows. Each provider supplies their own registration workflow. While the workflow names might be similar between provider packages, the location of the workflows in vRealize Orchestrator can be different and is provider-specific.

2 For this example, run the `Register IPAM Endpoint` registration workflow and specify the IPAM Infloblox endpoint type.

3 At the prompt for vRealize Automation credentials, enter your vRealize Automation administrator credentials, for example fabric administrator credentials.

   You must supply the registration workflow with vRealize Automation system administrator credentials. Even if a non-system administrator user is logged in to the vRealize Orchestrator client, if the vRealize Automation system administrator credentials are provided to the workflow the registration will succeed.

Results

In this example, the package registers Infoblox as a new IPAM endpoint type in the vRealize Automation endpoint service and makes the endpoint type available when you create or edit endpoints in vRealize Automation.

Note If the Infoblox IPAM connection disappears from the vRealize Orchestrator **Inventory** tab after you restart the vRealize Orchestrator server in the vRealize Orchestrator Control Center. To resolve this issue, run the `Create IPAM Connection` workflow from the **vRO *admin* > Library > Infoblox > vRA > Helpers** menu sequence. You can then the vRealize Orchestrator **Inventory** tab, select **Infoblox IPAM**, and refresh the page to display the Infoblox IPAM connection.

What to do next

You can now create an IPAM Infloblox type endpoint, or and endpoint for whatever third-party package or plug-in you have just registered, in vRealize Automation. See Create a Third-Party IPAM Provider Endpoint.

Checklist for Configuring Containers for vRealize Automation

To get started with Containers, you must configure the feature to support vRealize Automation user roles.

After you configure container definitions in Containers you can add and configure container components in a blueprint.

Table 3-4. Checklist for Configuring Containers for vRealize Automation

Task	Details
Assign the container administrator and container architect roles.	See Container roles information in *Foundations and Concepts*.
Define container definitions in the **Containers** tab in vRealize Automation.	See *Configuring vRealize Automation*.
Add container components and container networking components to blueprints in the **Design** tab in vRealize Automation.	See *Configuring vRealize Automation*.

Configuring Containers Using the vRealize Automation Appliance

Xenon service information is accessible in the vRealize Automation vRealize Automation appliance (**vRA Settings > Xenon**.

It contains information about the Xenon host VM, listening port, and service status. It also displays information about clustered Xenon nodes.

You can manage the Xenon Linux service with the following CLI commands in the vRealize Automation appliance.

Command	Description
`service xenon-service status`	Shows the status of the service as either running or stopped.
`service xenon-service start`	Starts the service.
`service xenon-service stop`	Stops the service.
`service xenon-service restart`	Restarts the service.
`service xenon-service get_host`	Shows the hostname on which the service is running.
`service xenon-service get_port`	Shows the service port.
`service xenon-service status_cluster`	Shows information about all clustered nodes in JSON format.
`service xenon-service reset`	Deletes the directory where Xenon keeps all configuration files and restarts the service.

Clustering Containers

You can use the Xenon service in conjunction with Containers for vRealize Automation to join nodes to a cluster. If the nodes are clustered, the Xenon service connects other nodes automatically when it starts.

You can monitor the cluster status on the **Xenon** tab in the vRealize Automation appliance or by running the following command in a CLI:

```
service xenon-service status_cluster
```

Xenon works on quorum-based clustering. The quorum is calculated by using the `(number of nodes / 2) + 1` formula.

Preparing Your vCloud Director Environment for vRealize Automation

Before you can integrate vCloud Director with vRealize Automation, you must install and configure your vCloud Director instance, set up your vSphere and cloud resources, and identify or create appropriate credentials to provide vRealize Automation with access to your vCloud Director environment.

Configure Your Environment

Configure your vSphere resources and cloud resources, including virtual datacenters and networks. For more information, see the vCloud Director documentation.

Required Credentials for Integration

Create or identify either organization administrator or system administrator credentials that your vRealize Automation IaaS administrators can use to bring your vCloud Director environment under vRealize Automation management as an endpoint.

User Role Considerations

vCloud Director user roles in an organization do not need to correspond with roles in vRealize Automation business groups. If the user account does not exist in vCloud Director, vCloud Director performs a lookup in the associated LDAP or Active Directory and creates the user account if the user exists in the identity store. If it cannot create the user account, it logs a warning but does not fail the provisioning process. The provisioned machine is then assigned to the account that was used to configure the vCloud Director endpoint.

For related information about vCloud Director user management, see the vCloud Director documentation.

Preparing Your vCloud Air Environment for vRealize Automation

Before you integrate vCloud Air with vRealize Automation, you must register for your vCloud Air account, set up your vCloud Air environment, and identify or create appropriate credentials to provide vRealize Automation with access to your environment.

Configure Your Environment

Configure your environment as instructed in the vCloud Air documentation.

Required Credentials for Integration

Create or identify either virtual infrastructure administrator or account administrator credentials that your vRealize Automation IaaS administrators can use to bring your vCloud Air environment under vRealize Automation management as an endpoint.

User Role Considerations

vCloud Air user roles in an organization do not need to correspond with roles in vRealize Automation business groups. For related information about vCloud Air user management, see the vCloud Air documentation.

Preparing Your Amazon Web Services Environment

Prepare elements and user roles in your Amazon Web Services environment, prepare Amazon Web Services to communicate with the guest agent and Software bootstrap agent, and understand how Amazon Web Services features map to vRealize Automation features.

Amazon Web Services User Roles and Credentials Required for vRealize Automation

You must configure credentials in Amazon AWS with the permissions required for vRealize Automation to manage your environment.

vRealize Automation requires access keys for endpoint credentials and does not support user names and passwords.

- Role and Permission Authorization in Amazon Web Services

  While the Power User role in AWS provides an AWS Directory Service user or group with full access to AWS services and resources, it is not required. Lower privileged user roles are also supported. The AWS security policy that meets the needs of vRealize Automation functionality is:

  ```
  {
     "Version": "2012-10-17",
     "Statement": [{
        "Effect": "Allow",
        "Action": [
                  "ec2:DescribeInstances",
                  "ec2:DescribeImages",
                  "ec2:DescribeKeyPairs",
                  "ec2:DescribeVpcs",
                  "ec2:DescribeSubnets",
                  "ec2:DescribeSecurityGroups",
                  "ec2:DescribeVolumes",

                  "ec2:DescribeVpcAttribute",
                  "ec2:DescribeAddresses",
                  "ec2:DescribeAvailabilityZones",
                  "ec2:DescribeImageAttribute",
                  "ec2:DescribeInstanceAttribute",
                  "ec2:DescribeVolumeStatus",
                  "ec2:DescribeVpnConnections",
                  "ec2:DescribeRegions",
                  "ec2:DescribeTags",
                  "ec2:DescribeVolumeAttribute",
                  "ec2:DescribeNetworkInterfaces",
                  "ec2:DescribeNetworkInterfaceAttribute",

                  "ec2:DisassociateAddress",
                  "ec2:GetPasswordData",

                  "ec2:ImportKeyPair",
                  "ec2:ImportVolume",

                  "ec2:CreateVolume",
                  "ec2:DeleteVolume",
                  "ec2:AttachVolume",
                  "ec2:ModifyVolumeAttribute",
                  "ec2:DetachVolume",

                  "ec2:AssignPrivateIpAddresses",
                  "ec2:UnassignPrivateIpAddresses",

                  "ec2:CreateKeyPair",
                  "ec2:DeleteKeyPair",
  ```

```
                "ec2:CreateTags",
                "ec2:AssociateAddress",
                "ec2:ReportInstanceStatus",
                "ec2:StartInstances",
                "ec2:StopInstances",
                "ec2:ModifyInstanceAttribute",
                "ec2:MonitorInstances",
                "ec2:RebootInstances",
                "ec2:RunInstances",
                "ec2:TerminateInstances",

                "elasticloadbalancing:RegisterInstancesWithLoadBalancer",
                "elasticloadbalancing:DeregisterInstancesFromLoadBalancer",
                "elasticloadbalancing:DescribeLoadBalancerAttributes",
                "elasticloadbalancing:DescribeLoadBalancers",
                "elasticloadbalancing:DescribeInstanceHealth"
      ],
      "Resource": "*"
    }
]}
```

- Authentication Credentials in Amazon Web Services

  For management of Amazon Identity and Access Management (IAM) users and groups, you must be configured with AWS Full Access Administrator credentials.

When you create an AWS endpoint in vRA, you're prompted to enter a key and secret key. To obtain the access key needed to create the Amazon endpoint, the administrator must either request a key from a user who has AWS Full Access Administrator credentials or be additionally configured with the AWS Full Access Administrator policy. See Create an Amazon Endpoint.

For information about enabling policies and roles, see the *AWS Identity and Access Management (IAM)* section of Amazon Web Services product documentation.

Allow Amazon Web Services to Communicate with the Software Bootstrap Agent and Guest Agent

If you intend to provision application blueprints that contain Software, or if you want the ability to further customize provisioned machines by using the guest agent, you must enable connectivity between your Amazon Web Services environment, where your machines are provisioned, and your vRealize Automation environment, where the agents download packages and receive instructions.

When you use vRealize Automation to provision Amazon Web Services machines with the vRealize Automation guest agent and Software bootstrap agent, you must set up network-to-Amazon VPC connectivity so your provisioned machines can communicate back to vRealize Automation to customize your machines.

For more information about Amazon Web Services VPC connectivity options, see the Amazon Web Services documentation.

Using Optional Amazon Features

vRealize Automation supports several Amazon features, including Amazon Virtual Private Cloud, elastic load balancers, elastic IP addresses, and elastic block storage.

Using Amazon Security Groups

Specify at least one security group when creating an Amazon reservation. Each available region requires at least one specified security group.

A security group acts as a firewall to control access to a machine. Every region includes at least the default security group. Administrators can use the Amazon Web Services Management Console to create additional security groups, configure ports for Microsoft Remote Desktop Protocol or SSH, and set up a virtual private network for an Amazon VPN.

When you create an Amazon reservation or configure a machine component in the blueprint, you can choose from the list of security groups that are available to the specified Amazon account region. Security groups are imported during data collection.

For information about creating and using security groups in Amazon Web Services, see Amazon documentation.

Understanding Amazon Web Service Regions

Each Amazon Web Services account is represented by a cloud endpoint. When you create an Amazon Elastic Cloud Computing endpoint in vRealize Automation, regions are collected as compute resources. After the IaaS administrator selects compute resources for a business group, inventory and state data collections occur automatically.

Inventory data collection, which occurs automatically once a day, collects data about what is on a compute resource, such as the following data:

- Elastic IP addresses
- Elastic load balancers
- Elastic block storage volumes

State data collection occurs automatically every 15 minutes by default. It gathers information about the state of managed instances, which are instances that vRealize Automation creates. The following are examples of state data:

- Windows passwords
- State of machines in load balancers
- Elastic IP addresses

A fabric administrator can initiate inventory and state data collection and deactivate or change the frequency of inventory and state data collection.

Using Amazon Virtual Private Cloud

Amazon Virtual Private Cloud allows you to provision Amazon machine instances in a private section of the Amazon Web Services cloud.

Amazon Web Services users can use Amazon VPC to design a virtual network topology according to your specifications. You can assign an Amazon VPC in vRealize Automation. However, vRealize Automation does not track the cost of using the Amazon VPC.

When you provision using Amazon VPC, vRealize Automation expects there to be a VPC subnet from which Amazon obtains a primary IP address. This address is static until the instance is terminated. You can also use the elastic IP pool to also attach an elastic IP address to an instance in vRealize Automation. That would allow the user to keep the same IP if they are continually provisioning and tearing down an instance in Amazon Web Services.

Use the AWS Management Console to create the following elements:

- An Amazon VPC, which includes Internet gateways, routing table, security groups and subnets, and available IP addresses.
- An Amazon Virtual Private Network if users need to log in to Amazon machines instances outside of the AWS Management Console.

vRealize Automation users can perform the following tasks when working with an Amazon VPC:

- A fabric administrator can assign an Amazon VPC to a cloud reservation. See Create an Amazon EC2 Reservation.
- A machine owner can assign an Amazon machine instance to an Amazon VPC.

For more information about creating an Amazon VPC, see Amazon Web Services documentation.

Using Elastic Load Balancers for Amazon Web Services

Elastic load balancers distribute incoming application traffic across Amazon Web Services instances. Amazon load balancing enables improved fault tolerance and performance.

Amazon makes elastic load balancing available for machines provisioned using Amazon EC2 blueprints.

The elastic load balancer must be available in the Amazon Web Services, Amazon Virtual Private Network and at the provisioning location. For example, if a load balancer is available in us-east1c and a machine location is us-east1b, the machine cannot use the available load balancer.

vRealize Automation does not create, manage, or monitor the elastic load balancers.

For information about creating Amazon elastic load balancers by using the Amazon Web Services Management Console, see Amazon Web Services documentation.

Using Elastic IP Addresses for Amazon Web Services

Using an elastic IP address allows you to rapidly fail over to another machine in a dynamic Amazon Web Services cloud environment. In vRealize Automation, the elastic IP address is available to all business groups that have rights to the region.

An administrator can allocate elastic IP addresses to your Amazon Web Services account by using the AWS Management Console. There are two groups of elastic IP addresses in any given a region, one range is allocated for non-Amazon VPC instances and another range is for Amazon VPCs. If you allocate addresses in a non-Amazon VPC region only, the addresses are not available in an Amazon VPC. The reverse is also true. If you allocate addresses in an Amazon VPC only, the addresses are not available in a non-Amazon VPC region.

The elastic IP address is associated with your Amazon Web Services account, not a particular machine, but only one machine at a time can use the address. The address remains associated with your Amazon Web Services account until you choose to release it. You can release it to map it to a specific machine instance.

An IaaS architect can add a custom property to a blueprint to assign an elastic IP address to machines during provisioning. Machine owners and administrators can view the elastic IP addresses assigned to machines, and machine owners or administrators with rights to edit machines can assign an elastic IP addresses after provisioning. However, if the address is already associated to a machine instance, and the instance is part of the Amazon Virtual Private Cloud deployment, Amazon does not assign the address.

For more information about creating and using Amazon elastic IP addresses, see Amazon Web Services documentation.

Using Elastic Block Storage for Amazon Web Services

Amazon elastic block storage provides block level storage volumes to use with an Amazon machine instance and Amazon Virtual Private Cloud. The storage volume can persist past the life of its associated Amazon machine instance in the Amazon Web Services cloud environment.

When you use an Amazon elastic block storage volume in conjunction with vRealize Automation, the following caveats apply:

- You cannot attach an existing elastic block storage volume when you provision a machine instance. However, if you create a new volume and request more than one machine at a time, the volume is created and attached to each instance. For example, if you create one volume named volume_1 and request three machines, a volume is created for each machine. Three volumes named volume_1 are created and attached to each machine. Each volume has a unique volume ID. Each volume is the same size and in the same location.
- The volume must be of the same operating system and in the same location as the machine to which you attach it.
- vRealize Automation does not manage the primary volume of an elastic block storage-backed instance.

For more information about Amazon elastic block storage, and details on how to enable it by using Amazon Web Services Management Console, see Amazon Web Services documentation.

Configure Network-to-Amazon VPC Connectivity for a Proof of Concept Environment

As the IT professional setting up an environment to evaluate vRealize Automation, you want to temporarily configure network-to-Amazon VPC connectivity to support the vRealize Automation Software feature.

Network-to-Amazon VPC connectivity is only required if you want to use the guest agent to customize provisioned machines, or if you want to include Software components in your blueprints. For a production environment, you would configure this connectivity officially through Amazon Web Services, but because you are working in a proof of concept environment, you want to create temporary network-to-Amazon VPC connectivity. You establish the SSH tunnel and then configure an Amazon reservation in vRealize Automation to route through your tunnel.

Prerequisites

- Create an Amazon Web Services security group called TunnelGroup and configure it to allow access on port 22.
- Create or identify a CentOS machine in your Amazon Web Services TunnelGroup security group and note the following configurations:
  - Administrative user credentials, for example *root*.
  - Public IP address.
  - Private IP address.
- Create or identify a CentOS machine on the same local network as your vRealize Automation installation.
- Install OpenSSH SSHD Server on both tunnel machines.

Procedure

1. Log in to your Amazon Web Services tunnel machine as the root user or similar.
2. Deactivate iptables.

   ```
   # service iptables save
   # service iptables stop
   # chkconfig iptables off
   ```

3. Edit `/etc/ssh/sshd_config` to enable `AllowTCPForwarding` and `GatewayPorts`.
4. Restart the service.

   ```
   /etc/init.d/sshd restart
   ```

5. Log in to the CentOS machine on the same local network as your vRealize Automation installation as the root user.

6 Invoke the SSH Tunnel from the local network machine to the Amazon Web Services tunnel machine.

```
ssh -N -v -o "ServerAliveInterval 30" -o "ServerAliveCountMax 40" -o "TCPKeepAlive yes” \

    -R 1442:vRealize_automation_appliance_fqdn:5480 \
    -R 1443:vRealize_automation_appliance_fqdn:443 \
    -R 1444:manager_service_fqdn:443 \
    User of Amazon tunnel machine@Public IP Address of Amazon tunnel machine
```

You configured port forwarding to allow your Amazon Web Services tunnel machine to access vRealize Automation resources, but your SSH tunnel does not function until you configure an Amazon reservation to route through the tunnel.

What to do next

1 Install the software bootstrap agent and the guest agent on a Windows or Linux reference machine to create an Amazon Machine Image that your IaaS architects can use to create blueprints. See Preparing for Software Provisioning.

2 Configure your Amazon reservation in vRealize Automation to route through your SSH tunnel. See Scenario: Create an Amazon Reservation for a Proof of Concept Environment.

Preparing Red Hat OpenStack Network and Security Features

vRealize Automation supports several features in OpenStack including security groups and floating IP addresses. Understand how these features work with vRealize Automation and configure them in your environment.

Using OpenStack Security Groups

Security groups allow you to specify rules to control network traffic over specific ports.

You can specify security groups in a reservation when requesting a machine. You can also specify an existing or on-demand NSX security group in the design canvas.

Security groups are imported during data collection.

Each available region requires at least one specified security group. When you create a reservation, the available security groups that are available to you in that region are displayed. Every region includes at least the default security group.

Additional security groups must be managed in the source resource. For more information about managing security groups for the various machines, see the OpenStack documentation.

Using Floating IP Addresses with OpenStack

You can assign floating IP addresses to a running virtual instance in OpenStack.

To enable assignment of floating IP addresses, you must configure IP forwarding and create a floating IP pool in Red Hat OpenStack. For more information, see the Red Hat OpenStack documentation.

You must entitle the Associate Floating IP and Disassociate Floating IP actions to machine owners. The entitled users can then associate a floating IP address to a provisioned machine from the external networks attached to the machine by selecting an available address from the floating IP address pool. After a floating IP address has been associated with a machine, a vRealize Automation user can select a Disassociate Floating IP option to view the currently assigned floating IP addresses and disassociate an address from a machine.

Preparing Your SCVMM Environment

Before you begin creating SCVMM templates and hardware profiles for use in vRealize Automation machine provisioning, you must understand the naming restrictions on template and hardware profile names, and configure SCVMM network and storage settings.

For related information about preparing your environment, see SCVMM Requirements.

For related information about machine provisioning, see Create a Hyper-V (SCVMM) Endpoint.

vRealize Automation does not support a deployment environment that uses an SCVMM private cloud configuration. vRealize Automation cannot currently collect from, allocate to, or provision based on SCVMM private clouds.

Template and Hardware Profile Naming

Because of naming conventions that SCVMM and vRealize Automation use for templates and hardware profiles, do not start your template or hardware profile names with the words temporary or profile. For example, the following terms are ignored during data collection:

- TemporaryTemplate
- Temporary Template
- TemporaryProfile
- Temporary Profile
- Profile

Required Network Configuration for SCVMM Clusters

SCVMM clusters only expose virtual networks to vRealize Automation, so you must have a 1:1 relationship between your virtual and logical networks. Using the SCVMM console, map each logical network to a virtual network and configure your SCVMM cluster to access machines through the virtual network.

Required Storage Configuration for SCVMM Clusters

On SCVMM Hyper-V clusters, vRealize Automation collects data and provisions on shared volumes only. Using the SCVMM console, configure your clusters to use shared resource volumes for storage.

Required Storage Configuration for Standalone SCVMM Hosts

For standalone SCVMM hosts, vRealize Automation collects data and provisions on the default virtual machine path. Using the SCVMM console, configure default virtual machine paths for your standalone hosts.

Configure Network-to-Azure VPC Connectivity

You must configure network-to-Azure connectivity if you want to use Software components in Azure blueprints.

Prerequisites

- Create an Azure security group called TunnelGroup and configure it to allow access on port 22.
- Create or identify a machine, such as a CentOS machine, in your Azure TunnelGroup security group and note the following configurations:
  - Administrative user credentials, for example *root*.
  - Public IP address.
  - Private IP address.
- Create or identify a CentOS machine on the same local network as your vRealize Automation installation.
- Install OpenSSH SSHD Server on both tunnel machines.

Procedure

1 Log in to your Azure tunnel machine as the root user or similar.

2 Deactivate iptables.

```
# service iptables save
# service iptables stop
# chkconfig iptables off
```

3 Edit `/etc/ssh/sshd_config` to enable `AllowTCPForwarding` and `GatewayPorts`.

4 Restart the service.

```
/etc/init.d/sshd restart
```

5 Log in to the CentOS machine on the same local network as your vRealize Automation installation as the root user.

6 Invoke the SSH Tunnel from the local network machine to the Azure tunnel machine.

```
ssh –N –v –o "ServerAliveInterval 30" –o "ServerAliveCountMax 40" –o "TCPKeepAlive yes” \
    -R 1442:vRealize_automation_appliance_fqdn:5480 \
    -R 1443:vRealize_automation_appliance_fqdn:443 \
    -R 1444:manager_service_fqdn:443 \
    User of Azure tunnel machine@Public IP Address of Azure tunnel machine
```

You configured port forwarding to allow your Azure tunnel machine to access vRealize Automation resources, but your SSH tunnel does not function until you configure an Azure reservation to route through the tunnel.

What to do next

1. Install the software bootstrap agent and the guest agent on a Windows or Linux reference machine to create an Azure Machine Image that your IaaS architects can use to create blueprints. See Preparing for Software Provisioning.
2. Configure your Azure reservation in vRealize Automation to route through your SSH tunnel. See Create a Reservation for Microsoft Azure .

Preparing for Machine Provisioning

Depending on your environment and your method of machine provisioning, you might need to configure elements outside of vRealize Automation.

For example, you might need to configure machine templates or machine images.

You might also need to configure NSX settings or run vRealize Orchestrator workflows.

For related information about specifying ports when preparing to provision machines, see the Reference Architecture section of the product documentation.

Choosing a Machine Provisioning Method to Prepare

For most machine provisioning methods, you must prepare some elements outside of vRealize Automation.

Table 3-5. Choosing a Machine Provisioning Method to Prepare

Scenario	Supported Endpoint	Agent Support	Provisioning Method	Pre-provisioning Preparations
Configure vRealize Automation to run custom Visual Basic scripts as additional steps in the machine life cycle, either before or after machine provisioning. For example, you could use a pre-provisioning script to generate certificates or security tokens before provisioning, and then a post-provisioning script to use the certificates and tokens after machine provisioning.	You can run Visual Basic scripts with any supported endpoint except Amazon Web Services.	Depends on the provisioning method you choose.	Supported as an additional step in any provisioning method, but you cannot use Visual Basic scripts with Amazon Web Services machines.	Checklist for Running Visual Basic Scripts During Provisioning
Provision application blueprints that automate the installation, configuration, and life cycle management of middleware and application deployment components such as Oracle, MySQL, WAR, and database Schemas.	■ vSphere ■ vCloud Air ■ vCloud Director ■ Amazon Web Services	■ (Required) Guest agent ■ (Required) Software bootstrap agent and guest agent	■ Clone ■ Clone (for vCloud Air or vCloud Director) ■ Linked clone ■ Amazon Machine Image	If you want the ability to use Software components in your blueprints, prepare a provisioning method that supports the guest agent and Software bootstrap agent. For more information about preparing for Software, see Preparing for Software Provisioning.
Further customize machines after provisioning by using the guest agent.	All virtual endpoints and Amazon Web Services.	■ (Required) Guest agent ■ (Optional) Software bootstrap agent and guest agent	Supported for all provisioning methods except Virtual Machine Image.	If you want the ability to customize machines after provisioning, select a provisioning method that supports the guest agent. .
Provision machines with no guest operating system. You can install an operating system after provisioning.	All virtual machine endpoints.	Not supported	Basic	No required pre-provisioning preparations outside of vRealize Automation.

Table 3-5. Choosing a Machine Provisioning Method to Prepare (continued)

Scenario	Supported Endpoint	Agent Support	Provisioning Method	Pre-provisioning Preparations
Provision a space-efficient copy of a virtual machine called a linked clone. Linked clones are based on a snapshot of a VM and use a chain of delta disks to track differences from a parent machine.	vSphere	■ (Optional) Guest agent ■ (Optional) Software bootstrap agent and guest agent	Linked Clone	You must have an existing vSphere virtual machine. If you want to support Software, you must install the guest agent and software bootstrap agent on the machine you intend to clone. Before you provision linked clone VMs, power off the VM snapshot.
Provision a space-efficient copy of a virtual machine by using Net App FlexClone technology.	vSphere	(Optional) Guest agent	NetApp FlexClone	See Checklist for Preparing to Provision by Cloning.
Provision machines by cloning from a template object created from an existing Windows or Linux machine, called the reference machine, and a customization object.	■ vSphere ■ KVM (RHEV) ■ SCVMM	■ (Optional) Guest agent ■ (Optional for vSphere only) Software bootstrap agent and guest agent	Clone	See Checklist for Preparing to Provision by Cloning. If you want to support Software, you must install the guest agent and software bootstrap agent on the vSpheremachine you intend to clone.
Provision vCloud Air or vCloud Director machines by cloning from a template and customization object.	■ vCloud Air ■ vCloud Director	■ (Optional) Guest agent ■ (Optional) Software bootstrap agent and guest agent	vCloud Air or vCloud Director Cloning	See Preparing for vCloud Air and vCloud Director Provisioning. If you want to support Software, create a template that contains the guest agent and software bootstrap agent. For vCloud Air, configure network connectivity between your vRealize Automation environment and your vCloud Air environment.
Provision a machine by booting from an ISO image, using a kickstart or autoYaSt configuration file and a Linux distribution image to install the operating system on the machine.	■ All virtual endpoints ■ Red Hat OpenStack	Guest agent is installed as part of the preparation instructions.	Linux Kickstart	Preparing for Linux Kickstart Provisioning

Table 3-5. Choosing a Machine Provisioning Method to Prepare (continued)

Scenario	Supported Endpoint	Agent Support	Provisioning Method	Pre-provisioning Preparations
Provision a machine and pass control to an SCCM task sequence to boot from an ISO image, deploy a Windows operating system, and install the vRealize Automation guest agent.	All virtual machine endpoints.	Guest agent is installed as part of the preparation instructions.	SCCM	Preparing for SCCM Provisioning
Provision a machine by booting into a WinPE environment and installing an operating system using a Windows Imaging File Format (WIM) image of an existing Windows reference machine.	■ All virtual endpoints ■ Red Hat OpenStack	Guest agent is required. When you create the WinPE image, you must manually insert the guest agent.	WIM	Preparing for WIM Provisioning
Launch an instance from a virtual machine image.	Red Hat OpenStack	Not supported	Virtual Machine Image	See Preparing for Virtual Machine Image Provisioning.
Launch an instance from an Amazon Machine Image.	Amazon Web Services	■ (Optional) Guest agent ■ (Optional) Software bootstrap agent and guest agent	Amazon Machine Image	Associate Amazon machine images and instance types with your Amazon Web Services account. If you want to support Software, create an Amazon Machine Image that contains the guest agent and software bootstrap agent, and configure network-to-VPC connectivity between your Amazon Web Services and vRealize Automation environments.

Checklist for Running Visual Basic Scripts During Provisioning

You can configure vRealize Automation to run your custom Visual Basic scripts as additional steps in the machine life cycle, either before or after machine provisioning. For example, you could use a pre-provisioning script to generate certificates or security tokens before provisioning, and then a post-provisioning script to use the certificates and tokens after machine provisioning. You can run Visual Basic scripts with any provisioning method, but you cannot use Visual Basic scripts with Amazon AWS machines.

Table 3-6. Running Visual Basic Scripts During Provisioning Checklist

Task	Location	Details
❑ Install and configure the EPI agent for Visual Basic scripts.	Typically the Manager Service host	See Installing the EPI Agent for Visual Basic Scripting.
❑ Create your visual basic scripts.	Machine where EPI agent is installed	vRealize Automation includes a sample Visual Basic script `PrePostProvisioningExample.vbs` in the `Scripts` subdirectory of the EPI agent installation directory. This script contains a header to load all arguments into a dictionary, a body in which you can include your functions, and a footer to return updated custom properties to vRealize Automation. When executing a Visual Basic script, the EPI agent passes all machine custom properties as arguments to the script. To return updated property values to vRealize Automation, place these properties in a dictionary and call a function provided by vRealize Automation.
❑ Gather the information required to include your scripts in blueprints.	Capture information and transfer to your infrastructure architects **Note** A fabric administrator can create a property group by using the property sets ExternalPreProvisioningVbScript and ExternalPostProvisioningVbScript to provide this required information. Doing so makes it easier for blueprint architects to include this information correctly in their blueprints.	■ The complete path to the Visual Basic script, including the filename and extension. For example, *`%System Drive%`*`Program Files (x86)\VMware\vCAC Agents\EPI_Agents\Scripts\SendEmail.vbs`. ■ To run a script before provisioning, instruct infrastructure architects to enter the complete path to the script as the value of the custom property `ExternalPreProvisioningVbScript`. To run a script after provisioning, they need to use the custom property `ExternalPostProvisioningVbScript`.

Using vRealize Automation Guest Agent in Provisioning

You can install the guest agent on reference machines to further customize a machine after deployment. You can use the reserved guest agent custom properties to perform basic customizations such as adding and formatting disks, or you can create your own custom scripts for the guest agent to run within the guest operating system of a provisioned machine.

After the deployment is completed and the customization specification is run (if you provided one), the guest agent creates an XML file that contains all of the deployed machine's custom properties `c:\VRMGuestAgent\site\workitem.xml`, completes any tasks assigned to it with the guest agent custom properties, and then deletes itself from the provisioned machine.

You can write your own custom scripts for the guest agent to run on deployed machines, and use custom properties on the machine blueprint to specify the location of those scripts and the order in which to run them. You can also use custom properties on the machine blueprint to pass custom property values to your scripts as parameters.

For example, you could use the guest agent to make the following customizations on deployed machines:

- Change the IP address
- Add or format drives
- Run security scripts
- Initialize another agent, for example Puppet or Chef

You can also provide an encrypted string as a custom property in a command line argument. This allows you to store encrypted information that the guest agent can decrypt and understand as a valid command line argument.

Note The Linux guest agent assigns static IPs during the create and cloning actions for Linux Kickstart and PXE provisioning relative to vRealize Automation custom properties in work items. The guest agent is unable to accommodate the newer consistent network naming scheme, such as in Ubuntu 16.x, when it assigns static IPs.

Your custom scripts do not have to be locally installed on the machine. As long as the provisioned machine has network access to the script location, the guest agent can access and run the scripts. This lowers maintenance costs because you can update your scripts without having to rebuild all of your templates.

You can configure security settings by specifying information in a reservation, blueprint, or guest agent script. If machines require a guest agent, add a security rule to the reservation or the blueprint.

If you choose to install the guest agent to run custom scripts on provisioned machines, your blueprints must include the appropriate guest agent custom properties. For example, if you install the guest agent on a template for cloning, create a custom script that changes the provisioned machine's IP address, and place the script in a shared location, you need to include a number of custom properties in your blueprint.

Table 3-7. Custom Properties for Changing IP Address of a Provisioned Machine with a Guest Agent

Custom Property	Description
`VirtualMachine.Admin.UseGuestAgent`	Set to **`true`** to initialize the guest agent when the provisioned machine is started.
`VirtualMachine.Customize.WaitComplete`	Set to True to prevent the provisioning workflow from sending work items to the guest agent until all customizations arecomplete. Set to False to allow work items to be created before customization is complete.

Table 3-7. Custom Properties for Changing IP Address of a Provisioned Machine with a Guest Agent (continued)

Custom Property	Description
`VirtualMachine.SoftwareN.ScriptPath`	Specifies the full path to an application's install script. The path must be a valid absolute path as seen by the guest operating system and must include the name of the script filename. You can pass custom property values as parameters to the script by inserting {*CustomPropertyName*} in the path string. For example, if you have a custom property named `ActivationKey` whose value is 1234, the script path is `D:\InstallApp.bat -key {ActivationKey}`. The guest agent runs the command `D:\InstallApp.bat -key 1234`. Your script file can then be programmed to accept and use this value. Insert {Owner} to pass the machine owner name to the script. You can also pass custom property values as parameters to the script by inserting {*YourCustomProperty*} in the path string. For example, entering the value **`\\vra-scripts.mycompany.com\scripts\changeIP.bat`** runs the `changeIP.bat` script from a shared location, but entering the value **`\\vra-scripts.mycompany.com\scripts\changeIP.bat {VirtualMachine.Network0.Address}`** runs the changeIP script but also passes the value of the `VirtualMachine.Network0.Address` property to the script as a parameter.
`VirtualMachine.ScriptPath.Decrypt`	Allows vRealize Automation to obtain an encrypted string that is passed as a properly formatted `VirtualMachine.SoftwareN.ScriptPath` custom property statement to the gugent command line. You can provide an encrypted string, such as your password, as a custom property in a command-line argument. This allows you to store encrypted information that the guest agent can decrypt and understand as a valid command-line argument. For example, the `VirtualMachine.Software0.ScriptPath = c:\dosomething.bat` *`password`* custom property string is not secure as it contains an actual password. To encrypt the password, you can create a vRealize Automation custom property, for example `MyPassword = password`, and enable encryption by selecting the available check box. The guest agent decrypts the **`[MyPassword]`** entry to the value in the custom property `MyPassword` and runs the script as `c:\dosomething.bat password`. ■ Create custom property **`MyPassword =`** ***`password`*** where *password* is the value of your actual password. Enable encryption by selecting the available check box.

Table 3-7. Custom Properties for Changing IP Address of a Provisioned Machine with a Guest Agent (continued)

Custom Property	Description
	■ Set custom property `VirtualMachine.ScriptPath.Decrypt` as **`VirtualMachine.ScriptPath.Decrypt = true`**. ■ Set custom property `VirtualMachine.Software0.ScriptPath` as **`VirtualMachine.Software0.ScriptPath = c:\dosomething.bat [MyPassword]`**. If you set `VirtualMachine.ScriptPath.Decrypt` to false, or do not create the `VirtualMachine.ScriptPath.Decrypt` custom property, then the string inside the square brackets ([and]) is not decrypted.

For more information about custom properties you can use with the guest agent, see Custom Properties for vRealize Automation Guest Agent.

Configuring the Guest Agent to Trust a Server

Installing the public key PEM file for the vRealize Automation Manager Service Host in the correct guest agent folder is the most secure approach to configuring the guest agent to trust a server.

Locate the guest agent folder on each template for the `cert.pem` PEM file for the Manager Service Host to trust a server:

- Windows guest agent folder on each template that uses the gugent

  ```
  C:\VRMGuestAgent\cert.pem
  ```

- Linux guest agent folder on each template that uses the gugent

  ```
  /usr/share/gugent/cert.pem
  ```

  If you do not put the `cert.pem` file in this location, the template reference machine cannot use the guest agent. For example, if you try to collect the public key information after the VM is started for by altering scripts, you break the security condition.

Additional considerations apply, depending on your configured environment:

- For WIM installations, you must add the public key PEM file contents to the console executable and user interface. The console flag is **/cert filename**.
- For RedHat kickstart installations, you must cut and paste the public key into the sample file, otherwise the guest agent fails to run.
- For SCCM installation, the `cert.pem` file must reside in the `VRMGuestAgent` folder.

- For Linux vSphere installs, the `cert.pem` file must reside in the `/usr/share/gugent` folder.

Note You can optionally install software and guest agents together by downloading the following script from `https://`*`APPLIANCE`*`/software/index.html`. The script allows you to handle acceptance of SSL certificate fingerprints as you create the templates.

- Linux

  `prepare_vra_template.sh`

- Windows

  `prepare_vra_template.ps1`

If you install the software and guest agent together, you do not need to use the instructions in Install the Guest Agent on a Linux Reference Machine or Install the Guest Agent on a Windows Reference Machine .

How to obtain the cert.pem file from the Manager Service host

1. On the Manager Service host, go to Administrative Tools and open Internet Information Services (IIS) Manager.
2. In the tree on the left, highlight the Manager Service host.
3. On the right, open Server Certificates.
4. Look for the certificate where **Issued To** is VMware vRA and **Issued By** is VMware vRA.
5. Right-click the certificate, and export it.
6. The saved certificate will be in PFX format. To convert it to PEM, use OpenSSL from the command line.

   ```
   openssl pkcs12 -in filename.pfx -out cert.pem -nodes
   ```

Install the Guest Agent on a Linux Reference Machine

Install the Linux guest agent on your reference machines to further customize machines after deployment.

Prerequisites

- Identify or create the reference machine.
- The guest agent files you download contain both `tar.gz` and `RPM` package formats. If your operating system cannot install `tar.gz` or `RPM` files, use a conversion tool to convert the installation files to your preferred package format.
- Establish secure trust between the guest agent and your Manager Service machine. See Configuring the Guest Agent to Trust a Server.

Procedure

1 Navigate to the vRealize Automation appliance management console page.

For example: https://*va-hostname.domain.com*.

2 Click **Guest and software agents page** in the vRealize Automation component installation section of the page.

For example: https://*va-hostname.domain.com*/software/index.html.

The **Guest and Software Agent Installers** page opens, displaying links to available downloads.

3 Click **Linux guest agent packages** in the guest agent installers section of the page to download and save the LinuxGuestAgentPkgs.zip file.

4 Unpack the downloaded LinuxGuestAgentPkgs.zip file to create the VraLinuxGuestAgent folder.

5 Install the guest agent package that corresponds to the guest operating system you are deploying during provisioning.

a Navigate to the VraLinuxGuestAgent subdirectory that corresponds to the guest operating system to deploy during provisioning, for example rhel32.

b Locate your preferred package format or convert a package to your preferred package format.

c Install the guest agent package on your reference machine.

For example, to install the files from the RPM package, run rpm -i gugent-*gugent-7.1.0-4201531.i386*.rpm.

6 Configure the guest agent to communicate with the Manager Service by running installgugent.sh *Manager_Service_Hostname_fdqn*:*portnumber* ssl *platform*.

The default port number for the Manager Service is 443. Accepted platform values are ec2, vcd, vca, and vsphere.

Option	Description

If you are using a load balancer	Enter the fully qualified domain name and port number of your Manager Service load balancer. For example:
	`cd /usr/share/gugent`
	`./installgugent.sh`
	`load_balancer_manager_service.mycompany.com:443 ssl ec2`

With no load balancer	Enter the fully qualified domain name and port number of your Manager Service machine. For example:
	`cd /usr/share/gugent`
	`./installgugent.sh manager_service_machine.mycompany.com:443`
	`ssl vsphere`

7 If deployed machines are not already configured to trust the Manager Service SSL certificate, you must install the `cert.pem` file on your reference machine to establish trust.

- For the most secure approach, obtain the `cert.pem` certificate and manually install the file on the reference machine.
- For a more convenient approach, you can connect to the manager service load balancer or manager service machine and download the `cert.pem` certificate.

Option	Description
If you are using a load balancer	As the root user on the reference machine, run the following command: `echo	openssl s_client -connect manager_service_load_balancer.mycompany.com:443	sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' > cert.pem`
With no load balancer	As the root user on the reference machine, run the following command: `echo	openssl s_client -connect manager_service_machine.mycompany.com:443	sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' > cert.pem`

8 If you are installing the guest agent on a Ubuntu operating system, create symbolic links for shared objects by running one of the following command sets.

Option	Description

64-bit systems	`cd /lib/x86_64-linux-gnu`
	`sudo ln -s libssl.so.1.0.0 libssl.so.10`
	`sudo ln -s libcrypto.so.1.0.0 libcrypto.so.10`

32-bit systems	`cd /lib/i386-linux-gnu`
	`sudo ln -s libssl.so.1.0.0 libssl.so.10`
	`sudo ln -s libcrypto.so.1.0.0 libcrypto.so.10`

What to do next

Convert your reference machine into a template for cloning, an Amazon Machine Image, or a snapshot that your IaaS architects can use when creating blueprints.

Install the Guest Agent on a Windows Reference Machine

Install the vRealize Automation Windows guest agent on a Windows reference machine to run as a Windows service and enable further customization of machines.

Prerequisites

- Identify or create the reference machine.
- Establish secure trust between the guest agent and your Manager Service machine. See Configuring the Guest Agent to Trust a Server.

Procedure

1 Navigate to the vRealize Automation appliance **Guest and Software Agent Installers** page:

  https://*vrealize-automation-appliance-FQDN*/software

2 Under **Guest Agent Installers**, download and save the 32-bit or 64-bit executable to the root of the `C:` drive.

  Note There is a command-line alternative to this procedure for guest agent installation. Instead of downloading the executables, you may go to **Windows Software Installers** on the Guest and Software Agent Installers page. There, you can download and run the `prepare_vra_template.ps1` PowerShell script:

  ```
  PowerShell -NoProfile -ExecutionPolicy Bypass -Command prepare_vra_template.ps1
  ```

3 Extract the Windows guest agent files by running the executable.

  Extraction creates `C:\VRMGuestAgent` and adds the files.

  Do not rename `C:\VRMGuestAgent`.

4 Configure the guest agent to communicate with the Manager Service.

  a Open an elevated command prompt.

  b Navigate to `C:\VRMGuestAgent`.

  c Put the trusted Manager Service PEM file in the `C:\VRMGuestAgent\` directory to configure the guest agent to trust your Manager Service machine.

  d Run `winservice -i -h` *`Manager_Service_Hostname_fdqn`*`:`*`portnumber`* `-p ssl`.

  The default port number for the Manager Service is 443.

Option	Description
If you are using a load balancer	Enter the fully qualified domain name and port number of your Manager Service load balancer. For example, `winservice -i -h` *`load_balancer_manager_service.mycompany.com:443`* `-p ssl`.
With no load balancer	Enter the fully qualified domain name and port number of your Manager Service machine. For example, `winservice -i -h` *`manager_service_machine.mycompany.com:443`* `-p ssl`.
If you are preparing an Amazon machine image	You need to specify that you are using Amazon. For example, `winservice -i -h` *`manager_service_machine.mycompany.com:443`*`:`*`443`* `-p ssl -c` *`ec2`*

Results

The name of the Windows service is VCACGuestAgentService. You can find the installation log `VCAC-GuestAgentService.log` in `C:\VRMGuestAgent`.

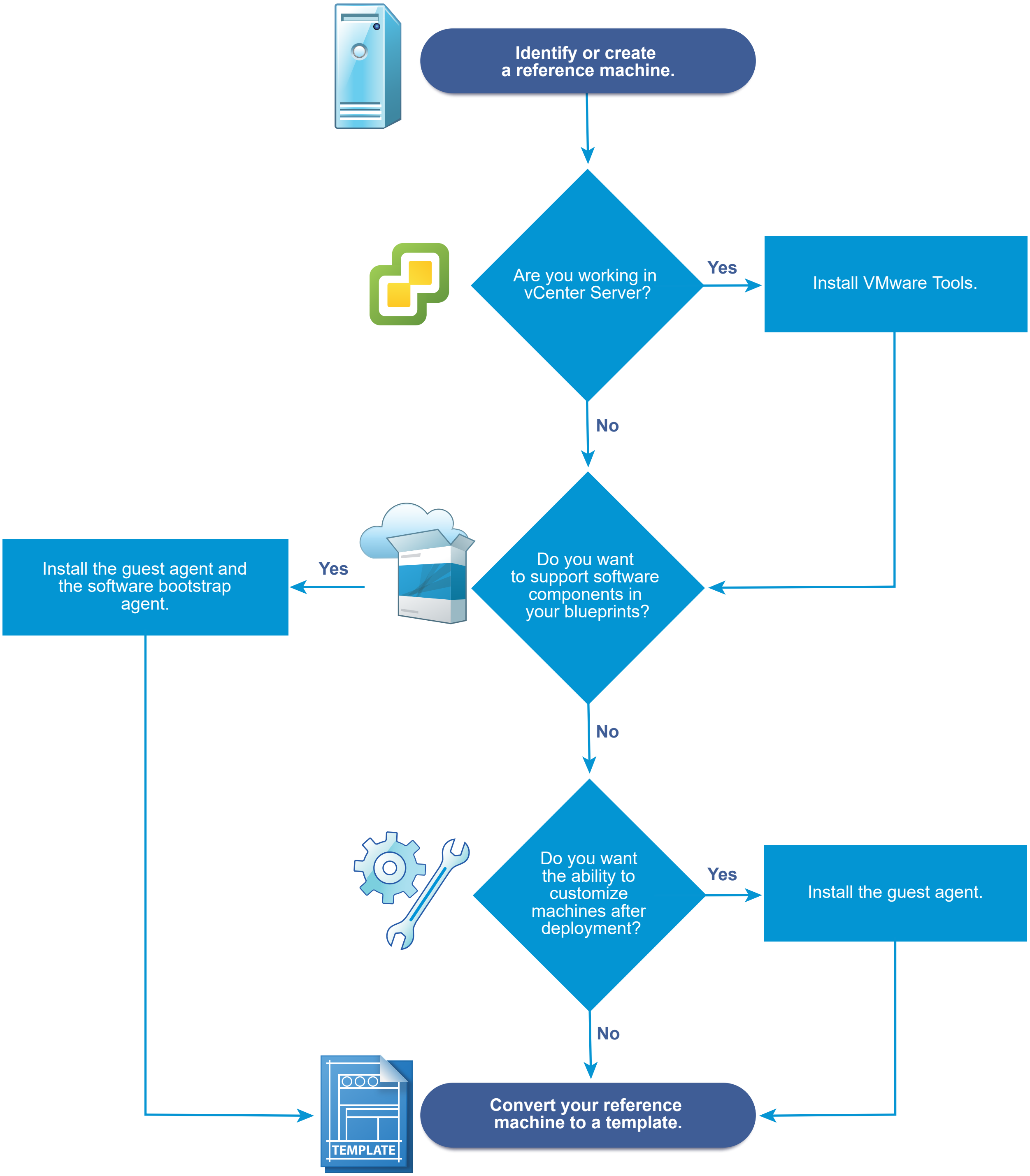
What to do next

Convert your reference machine into a template for cloning, an Amazon machine image, or a snapshot so your IaaS architects can use your template when creating blueprints.

Checklist for Preparing to Provision by Cloning

You must perform some preparation outside of vRealize Automation to create the template and the customization objects used to clone Linux and Windows virtual machines.

Cloning requires a template to clone from, created from a reference machine.


Identify or create a reference machine.
Are you working in vCenter Server?
Yes
Install VMware Tools.
No
Do you want to support software components in your blueprints?
Yes
Install the guest agent and the software bootstrap agent.
No
Do you want the ability to customize machines after deployment?
Yes
Install the guest agent.
No
TEMPLATE
Convert your reference machine to a template.

If you are provisioning a Windows machine by cloning, the only way to join the provisioned machine to an Active Directory domain is by using the customization specification from vCenter Server or by including a guest operating system profile with your SCVMM template. Machines provisioned by cloning cannot be placed in an Active Directory container during provisioning. You must do this manually after provisioning.

Table 3-8. Checklist for Preparing to Provision by Cloning

Task	Location	Details
❑ Identify or create the reference machine.	Hypervisor	See the documentation provided by your hypervisor.
❑ (Optional) If you want your clone template to support Software components, install the vRealize Automation guest agent and software bootstrap agent on your reference machine.	Reference machine	For Windows reference machines, see Prepare a Windows Reference Machine to Support Software. For Linux reference machines, see Prepare a Linux Reference Machine to Support Software.
❑ (Optional) If you do not need your clone template to support Software components, but you do want the ability to customize deployed machines, install the vRealize Automation guest agent on your reference machine.	Reference machine	See Using vRealize Automation Guest Agent in Provisioning.
❑ If you are working in a vCenter Server environment, install VMware Tools on the reference machine.	vCenter Server	See the VMware Tools documentation.
❑ Use the reference machine to create a template for cloning.	Hypervisor	The reference machine may be powered on or off. If you are cloning in vCenter Server, you can use a reference machine directly without creating a template. See the documentation provided by your hypervisor.
❑ Create the customization object to configure cloned machines by applying System Preparation Utility information or a Linux customization.	Hypervisor	If you are cloning for Linux you can install the Linux guest agent and provide external customization scripts instead of creating a customization object. If you are cloning with vCenter Server, you must provide the customization specification as the customization object. See the documentation provided by your hypervisor.
❑ Gather the information required to create blueprints that clone your template.	Capture information and transfer to your IaaS architects.	See Worksheet for Virtual Provisioning by Cloning.

Worksheet for Virtual Provisioning by Cloning

Complete the knowledge transfer worksheet to capture information about the template, customizations, and custom properties required to create clone blueprints for the templates you prepared in your environment. Not all of this information is required for every implementation. Use this worksheet as a guide, or copy and paste the worksheet tables into a word processing tool for editing.

Required Template and Reservation Information

Table 3-9. Template and Reservation Information Worksheet

Required Information	My Value	Details
Template name		
Reservations on which the template is available, or reservation policy to apply		To avoid errors during provisioning, ensure that the template is available on all reservations or create reservation policies that architects can use to restrict the blueprint to reservations where the template is available.
(vSphere only) Type of cloning requested for this template		■ Clone ■ Linked Clone ■ NetApp FlexClone
Customization specification name (Required for cloning with static IP addresses)		You cannot perform customizations of Windows machines without using a vSpherecustomization specification.
(SCVMM only) ISO name		
(SCVMM only) Virtual hard disk		
(SCVMM only) Hardware profile to attach to provisioned machines		

Required Property Groups

You can complete the custom property information sections of the worksheet, or you can create property groups and ask architects to add your property groups to their blueprints instead of numerous individual custom properties.

Required vCenter Server Operating System

You must supply the guest operating system custom property for vCenter Server provisioning.

Table 3-10. vCenter Server Operating System

Custom Property	My Value	Description
`VMware.VirtualCenter.OperatingSystem`		Specifies the vCenter Server guest operating system version (`VirtualMachineGuestOsIdentifier`) with which vCenter Server creates the machine. This operating system version must match the operating system version to be installed on the provisioned machine. Administrators can create property groups using one of several property sets, for example, `VMware[OS_Version]Properties`, that are predefined to include the correct `VMware.VirtualCenter.OperatingSystem` values. This property is for virtual provisioning.

Visual Basic Script Information

If you configured vRealize Automation to run your custom Visual Basic scripts as additional steps in the machine life cycle, you must include information about the scripts in the blueprint.

Note A fabric administrator can create a property group by using the property sets ExternalPreProvisioningVbScript and ExternalPostProvisioningVbScript to provide this required information. Doing so makes it easier for blueprint architects to include this information correctly in their blueprints.

Table 3-11. Visual Basic Script Information

Custom Property	My Value	Description
`ExternalPreProvisioningVbScript`		Run a script before provisioning. Enter the complete path to the script including the filename and extension. *%System Drive%*`Program Files (x86)\VMware\vCAC Agents \EPI_Agents\Scripts \SendEmail.vbs`.
`ExternalPostProvisioningVbScript`		Run a script after provisioning. Enter the complete path to the script including the filename and extension. *%System Drive%*`Program Files (x86)\VMware\vCAC Agents \EPI_Agents\Scripts\SendEmail.vbs`

Linux Guest Agent Customization Script Information

If you configured your Linux template to use the guest agent for running customization scripts, you must include information about the scripts in the blueprint.

Table 3-12. Linux Guest Agent Customization Script Information Worksheet

Custom Property	My Value	Description
`Linux.ExternalScript.Name`		Specifies the name of an optional customization script, for example `config.sh`, that the Linux guest agent runs after the operating system is installed. This property is available for Linux machines cloned from templates on which the Linux agent is installed. If you specify an external script, you must also define its location by using the `Linux.ExternalScript.LocationType` and `Linux.ExternalScript.Path` properties.
`Linux.ExternalScript.LocationType`		Specifies the location type of the customization script named in the `Linux.ExternalScript.Name` property. This can be either local or nfs. You must also specify the script location using the `Linux.ExternalScript.Path` property. If the location type is nfs, also use the `Linux.ExternalScript.Server` property.
`Linux.ExternalScript.Server`		Specifies the name of the NFS server, for example lab-ad.lab.local, on which the Linux external customization script named in `Linux.ExternalScript.Name` is located.
`Linux.ExternalScript.Path`		Specifies the local path to the Linux customization script or the export path to the Linux customization on the NFS server. The value must begin with a forward slash and not include the file name, for example `/scripts/linux/config.sh`.

Other Guest Agent Custom Properties

If you installed the guest agent on your reference machine, you can use custom properties to further customize machines after deployment.

Table 3-13. Custom Properties for Customizing Cloned Machines with a Guest Agent Worksheet

Custom Property	My Value	Description
`VirtualMachine.Admin.AddOwnerToAdmins`		Set to True (default) to add the machine's owner, as specified by the `VirtualMachine.Admin.Owner` property, to the local administrators group on the machine.
`VirtualMachine.Admin.AllowLogin`		Set to True (default) to add the machine owner to the local remote desktop users group, as specified by the `VirtualMachine.Admin.Owner` property.
`VirtualMachine.Admin.UseGuestAgent`		If the guest agent is installed as a service on a template for cloning, set to True on the machine blueprint to enable the guest agent service on machines cloned from that template. When the machine is started, the guest agent service is started. Set to False to deactivate the guest agent. If set to False, the enhanced clone workfow will not use the guest agent for guest operating system tasks, reducing its functionality to `VMwareCloneWorkflow`. If not specified or set to anything other than False, the enhanced clone workflow sends work items to the guest agent.
`VirtualMachine.DiskN.Active`		Set to True (default) to specify that the machine's disk *N* is active. Set to False to specify that the machine's disk *N* is not active.
`VirtualMachine.DiskN.Label`		Specifies the label for a machine's disk *N*. The disk label maximum is 32 characters. Disk numbering must be sequential. When used with a guest agent, specifies the label of a machine's disk *N* inside the guest operating system.
`VirtualMachine.DiskN.Letter`		Specifies the drive letter or mount point of a machine's disk *N*. The default is C. For example, to specify the letter D for Disk 1, define the custom property as `VirtualMachine.Disk1.Letter` and enter the value D. Disk numbering must be sequential. When used in conjunction with a guest agent, this value specifies the drive letter or mount point under which an additional disk *N* is mounted by the guest agent in the guest operating system.

Table 3-13. Custom Properties for Customizing Cloned Machines with a Guest Agent Worksheet (continued)

Custom Property	My Value	Description
`VirtualMachine.Admin.CustomizeGuestOSDelay`		Specifies the time to wait after customization is complete and before starting the guest operating system customization. The value must be in HH:MM:SS format. If the value is not set, the default value is one minute (00:01:00). If you choose not to include this custom property, provisioning can fail if the virtual machine reboots before guest agent work items are completed, causing provisioning to fail.
`VirtualMachine.Customize.WaitComplete`		Set to True to prevent the provisioning workflow from sending work items to the guest agent until all customizations arecomplete. Set to False to allow work items to be created before customization is complete.
`VirtualMachine.SoftwareN.Name`		Specifies the descriptive name of a software application *N* or script to install or run during provisioning. This is an optional and information-only property. It serves no real function for the enhanced clone workflow or the guest agent but it is useful for a custom software selection in a user interface or for software use reporting.
`VirtualMachine.SoftwareN.ScriptPath`		Specifies the full path to an application's install script. The path must be a valid absolute path as seen by the guest operating system and must include the name of the script filename. You can pass custom property values as parameters to the script by inserting {*CustomPropertyName*} in the path string. For example, if you have a custom property named `ActivationKey` whose value is 1234, the script path is `D:\InstallApp.bat –key {ActivationKey}`. The guest agent runs the command `D:\InstallApp.bat –key 1234`. Your script file can then be programmed to accept and use this value.

Table 3-13. Custom Properties for Customizing Cloned Machines with a Guest Agent Worksheet (continued)

Custom Property	My Value	Description
`VirtualMachine.SoftwareN.ISOName`		Specifies the path and filename of the ISO file relative to the datastore root. The format is *`/folder_name/subfolder_name/file_name`*`.iso`. If a value is not specified, the ISO is not mounted.
`VirtualMachine.SoftwareN.ISOLocation`		Specifies the storage path that contains the ISO image file to be used by the application or script. Format the path as it appears on the host reservation, for example `netapp-1:it_nfs_1`. If a value is not specified, the ISO is not mounted.

Networking Custom Properties

You can specify configuration for specific network devices on a machine by using custom properties.

Common networking-related custom properties are listed in the following table. For additional and related custom properties, see *Custom Properties for Clone Blueprints* and *Custom Properties for Networking* in *Custom Properties Reference*.

Table 3-14. Custom Properties for Networking Configuration

Custom Property	My Value	Description
`VirtualMachine.NetworkN.Address`		Specifies the IP address of network device *N* in a machine provisioned with a static IP address.
`VirtualMachine.NetworkN.MacAddressType`		Indicates whether the MAC address of network device *N* is generated or user-defined (static). This property is available for cloning. The default value is generated. If the value is static, you must also use `VirtualMachine.NetworkN.MacAddress` to specify the MAC address. `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks.

Table 3-14. Custom Properties for Networking Configuration (continued)

Custom Property	My Value	Description
`VirtualMachine.NetworkN.MacAddress`		Specifies the MAC address of a network device *N*. This property is available for cloning. If the value of `VirtualMachine.NetworkN.MacAddressType` is generated, this property contains the generated address. If the value of `VirtualMachine.NetworkN.MacAddressType` is static, this property specifies the MAC address. For virtual machines provisioned on ESX server hosts, the address must be in the range specified by VMware. For details, see vSphere documentation. `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks.

Table 3-14. Custom Properties for Networking Configuration (continued)

Custom Property	My Value	Description
`VirtualMachine.NetworkN.Name`		Specifies the name of the network to connect to, for example the network device *N* to which a machine is attached. This is equivalent to a network interface card (NIC). By default, a network is assigned from the network paths available on the reservation on which the machine is provisioned. Also see `VirtualMachine.NetworkN.AddressType`. You can ensure that a network device is connected to a specific network by setting the value of this property to the name of a network on an available reservation. For example, if you give properties for N= 0 and 1, you get 2 NICs and their assigned value, provided the network is selected in the associated reservation. `VirtualMachine.Network`*N* custom properties are specific to blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks. For an example of how to use this custom property to dynamically set `VirtualMachine.Network0.Name` based on a consumer's selection from a list of predefined available networks, see the Adding a Network Selection Drop-Down in vRA 7 blog post.

Table 3-14. Custom Properties for Networking Configuration (continued)

Custom Property	My Value	Description
`VirtualMachine.NetworkN.PortID`		Specifies the port ID to use for network device *N* when using a dvPort group with a vSphere distributed switch. `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks.
`VirtualMachine.NetworkN.NetworkProfileName`		Specifies the name of a network profile from which to assign a static IP address to network device *N* or from which to obtain the range of static IP addresses that can be assigned to network device *N* of a cloned machine, where *N*=0 for the first device, 1 for the second, and so on. The network profile that the property points to is used to allocate an IP address. The property determines the network that the machine attaches to, based on the reservation.

Table 3-14. Custom Properties for Networking Configuration (continued)

Custom Property	My Value	Description
- `VirtualMachine.NetworkN.SubnetMask` - `VirtualMachine.NetworkN.Gateway` - `VirtualMachine.NetworkN.PrimaryDns` - `VirtualMachine.NetworkN.SecondaryDns` - `VirtualMachine.NetworkN.PrimaryWins` - `VirtualMachine.NetworkN.SecondaryWins` - `VirtualMachine.NetworkN.DnsSuffix` - `VirtualMachine.NetworkN.DnsSearchSuffixes`		Appending a name allows you to create multiple versions of a custom property. For example, the following properties might list load balancing pools set up for general use and machines with high, moderate, and low performance requirements: - `VCNS.LoadBalancerEdgePool.Names` - `VCNS.LoadBalancerEdgePool.Names.moderate` - `VCNS.LoadBalancerEdgePool.Names.high` - `VCNS.LoadBalancerEdgePool.Names.low` Configures attributes of the network profile specified in `VirtualMachine.NetworkN.NetworkProfileName`.
`VCNS.LoadBalancerEdgePool.Names.`*name*		Specifies the NSX load balancing pools to which the virtual machine is assigned during provisioning. The virtual machine is assigned to all service ports of all specified pools. The value is an *edge/pool* name or a list of *edge/pool* names separated by commas. Names are case-sensitive. Appending a name allows you to create multiple versions of a custom property. For example, the following properties might list load balancing pools set up for general use and machines with high, moderate, and low performance requirements: - `VCNS.LoadBalancerEdgePool.Names` - `VCNS.LoadBalancerEdgePool.Names.moderate` - `VCNS.LoadBalancerEdgePool.Names.high` - `VCNS.LoadBalancerEdgePool.Names.low`

Table 3-14. Custom Properties for Networking Configuration (continued)

Custom Property	My Value	Description
`VCNS.SecurityGroup.Names.`*name*		Specifies the NSX security group or groups to which the virtual machine is assigned during provisioning. The value is a security group name or a list of names separated by commas. Names are case-sensitive. Appending a name allows you to create multiple versions of the property, which can be used separately or in combination. For example, the following properties can list security groups intended for general use, for the sales force, and for support: ■ `VCNS.SecurityGroup.Names` ■ `VCNS.SecurityGroup.Names.sales` ■ `VCNS.SecurityGroup.Names.support`
`VCNS.SecurityTag.Names.`*name*		Specifies the NSX security tag or tags to which the virtual machine is associated during provisioning. The value is a security tag name or a list of names separated by commas. Names are case-sensitive. Appending a name allows you to create multiple versions of the property, which can be used separately or in combination. For example, the following properties can list security tags intended for general use, for the sales force, and for support: ■ `VCNS.SecurityTag.Names` ■ `VCNS.SecurityTag.Names.sales` ■ `VCNS.SecurityTag.Names.support`

Joining a Linux Machine to a Windows Active Directory Domain

There are a several ways to join a Linux machine to a Windows Active Directory domain when you provision the machine.

- If you are provisioning by cloning, you must use either a customization specification (for provisioning a vSphere machine) or include a guest operating system profile with an SCVMM template. When you provision the machine, it is joined to the specified domain.

- If you are not provisioning by cloning, you can use the DNS suffix setting in the blueprint's associated network profile to identify the domain. However, for Windows clone provisioning with a static IP address assignment, you *must* use a vSphere customization specification.
- If you use a vSphere customization specification, when machines are provisioned they are joined to the domain identified in the customization specification and not the domain specified as the DNS suffix in the blueprint's associated network profile.

vSphere customization specifications are vSphere objects that contain a pre-defined set of conditions for Windows and Linux guest operating system settings. You can add a customization specification name to your vRealize Automation blueprint by using the **Customization spec** setting on the machine’s **Build Information** tab.

For information about creating customization specifications in vSphere, see customization specification topics in vSphere product documentation such as *Creating and Managing Customization Specifications*.

Preparing for vCloud Air and vCloud Director Provisioning

To prepare for provisioning vCloud Air and vCloud Director machines by using vRealize Automation, you must configure the organization virtual data center with templates and customization objects.

To provision vCloud Air and vCloud Director resources using vRealize Automation, the organization requires a template to clone from that consists of one or more machine resources.

Templates that are to be shared across organizations must be public. Only reserved templates are available to vRealize Automation as a cloning source.

Note When you create a blueprint by cloning from a template, that template's unique identifier becomes associated with the blueprint. When the blueprint is published to the vRealize Automation catalog and used in the provisioning and data collection processes, the associated template is recognized. If you delete the template in vCloud Air or vCloud Director, subsequent vRealize Automation provisioning and data collection fails because the associated template no longer exists. Instead of deleting and recreating a template, for example to upload an updated version, replace the template using the vCloud Air vCloud Director template replacement process. Using vCloud Air or vCloud Director to replace the template, rather than deleting and recreating the template, keeps the template's unique ID intact and allows provisioning and data collection to continue functioning.

The following overview illustrates the steps you need to perform before you use vRealize Automation to create endpoints and define reservations and blueprints. For more information about these administrative tasks, see vCloud Air and vCloud Director product documentation.

1 In vCloud Air or vCloud Director, create a template for cloning and add it to the organization catalog.

2 In vCloud Air or vCloud Director, use the template to specify custom settings such as passwords, domain, and scripts for the guest operating system on each machine.

You can use vRealize Automation to override some of these settings.

Customization can vary depending on the guest operating system of the resource.

3 In vCloud Air or vCloud Director, configure the catalog to be shared with everyone in the organization.

   In vCloud Air or vCloud Director, configure account administrator access to applicable organizations to allow all users and groups in the organization to have access to the catalog. Without this sharing designation, the catalog templates are not be visible to endpoint or blueprint architects in vRealize Automation.

4 Gather the following information so that you can include it in blueprints:

   - Name of the vCloud Air or vCloud Director template.
   - Amount of total storage specified for the template.

Preparing for Linux Kickstart Provisioning

Linux Kickstart provisioning uses a configuration file to automate a Linux installation on a newly provisioned machine. To prepare for provisioning you must create a bootable ISO image and a Kickstart or autoYaST configuration file.

The following is a high-level overview of the steps required to prepare for Linux Kickstart provisioning:

1 Verify that a DHCP server is available on the network. vRealize Automation cannot provision machines by using Linux Kickstart provisioning unless DHCP is available.

2 Prepare the configuration file. In the configuration file, you must specify the locations of the vRealize Automation server and the Linux agent installation package. See Prepare the Linux Kickstart Configuration Sample File.

3 Edit the `isolinux/isolinux.cfg` or `loader/isolinux.cfg` to specify the name and location of the configuration file and the appropriate Linux distribution source.

4 Create the boot ISO image and save it to the location required by your virtualization platform. See the documentation provided by your hypervisor for information about the required location.

5 (Optional) Add customization scripts.

   a To specify post-installation customization scripts in the configuration file, see Specify Custom Scripts in a kickstart/autoYaST Configuration File.

   b To call Visual Basic scripts in blueprint, see Checklist for Running Visual Basic Scripts During Provisioning.

6 Gather the following information so that blueprint architects can include it in their blueprints:

   a The name and location of the ISO image.

b For vCenter Server integrations, the vCenter Server guest operating system version with which vCenter Server is to create the machine.

Note You can create a property group with the property set BootIsoProperties to include the required ISO information. This makes it easier to include this information correctly on blueprints.

Prepare the Linux Kickstart Configuration Sample File

vRealize Automation provides sample configuration files that you can modify and edit to suit your needs. There are several changes required to make the files usable.

Procedure

1 Navigate to the vRealize Automation appliance management console page.

   For example: `https://`*`va-hostname.domain.com`*.

2 Click **Guest and software agents page** in the vRealize Automation component installation section of the page.

   For example: `https://`*`va-hostname.domain.com`*`/software/index.html`.

   The **Guest and Software Agent Installers** page opens, displaying links to available downloads.

3 Click **Linux guest agent packages** in the guest agent installers section of the page to download and save the `LinuxGuestAgentPkgs.zip` file.

4 Unpack the downloaded `LinuxGuestAgentPkgs.zip` file to create the `VraLinuxGuestAgent` folder.

5 Navigate to the `VraLinuxGuestAgent` subdirectory that corresponds to the guest operating system to deploy during provisioning.

   For example: `rhel32`.

6 Open a file in the samples subdirectory that corresponds to your target system.

   For example, `samples/sample-https-rhel6-x86.cfg`.

7 Replace all instances of the string `host=dcac.example.net` with the IP address or fully qualified domain name and port number for the Manager Service or the load balancer for the Manager Service.

Platform	Required Format
vSphere ESXi	IP Address, for example: `--host=172.20.9.59`
vSphere ESX	IP Address, for example: `--host=172.20.9.58`
SUSE 10	IP Address, for example: `--host=172.20.9.57`
All others	FQDN, for example: `--host=mycompany-host1.mycompany.local:443`

8. Locate each instance of `gugent.rpm` or `gugent.tar.gz` and replace the URL `rpm.example.net` with the location of the guest agent package.

   For example:

   ```
   rpm –i nfs:172.20.9.59/suseagent/gugent.rpm
   ```

9. Save the file to a location accessible to newly provisioned machines.

Specify Custom Scripts in a kickstart/autoYaST Configuration File

You can modify the configuration file to copy or install custom scripts onto newly provisioned machines. The Linux agent runs the scripts at the specified point in the workflow.

Your script can reference any of the `./properties.xml` files in the `/usr/share/gugent/site/`*`workitem`* directories.

Prerequisites

- Prepare a kickstart or autoYaST configuration file. See Prepare the Linux Kickstart Configuration Sample File.
- Your script must return a non-zero value on failure to prevent machine provisioning failure.

Procedure

1. Create or identify the script you want to use.
2. Save the script as *NN_scriptname*.

   NN is a two digit number. Scripts are run in order from lowest to highest. If two scripts have the same number, the order is alphabetical based on *scriptname*.
3. Make your script runnable.
4. Locate the post-installation section of your kickstart or autoYaST configuration file.

   In kickstart, this is indicated by `%post`. In autoYaST, this is indicated by `post-scripts`.
5. Modify the post-installation section of the configuration file to copy or install your script into the `/usr/share/gugent/site/`*`workitem`* directory of your choice.

   Custom scripts are most commonly run for virtual kickstart/autoYaST with the work items SetupOS (for create provisioning) and CustomizeOS (for clone provisioning), but you can run scripts at any point in the workflow.

   For example, you can modify the configuration file to copy the script `11_addusers.sh` to the `/usr/share/gugent/site/SetupOS` directory on a newly provisioned machine by using the following command:

   ```
   cp nfs:172.20.9.59/linuxscripts/11_addusers.sh /usr/share/gugent/site/SetupOS
   ```

Results

The Linux agent runs the script in the order specified by the work item directory and the script file name.

Preparing for SCCM Provisioning

vRealize Automation boots a newly provisioned machine from an ISO image, and then passes control to the specified SCCM task sequence.

SCCM provisioning is supported for the deployment of Windows operating systems. Linux is not supported. Software distribution and updates are not supported.

By default, an SCCM machine is configured to confirm membership in the applicable collection every 10 seconds following provisioning. In some cases, this interval can cause problems with the registration process. Two properties are available to customize the confirmation process. The first property is called `SCCM refresh collection setting`. By default this property is set to `true` to confirm that the machine performs a membership check. If appropriate, you can change it to `false` to configure the machine to skip membership checking. The second property is called `SCCM machine membership check interval`. As noted, the default is 10 seconds, but you can set it to a different value to increase the re-trigger window if you are experiencing registration problems. Both of these properties are located IaaS global settings under **Infrastructure > Administration > Global Settings**.

The following is a high-level overview of the steps required to prepare for SCCM provisioning:

1. Communication with SCCM requires the NetBIOS name of the SCCM server.

   Work with your network administrator to ensure that at least one Distributed Execution Manager (DEM) can resolve the FQDN of the SCCM server to its NetBIOS name.

   You do not need to place DEMs directly on the same network as the SCCM server, but DEMs need to be able to reach the SCCM server over IP.

2. Create a software package that includes the vRealize Automation guest agent. See Create a Software Package for SCCM Provisioning.

3. In SCCM, create the desired task sequence for provisioning the machine. The final step must be to install the software package you created that contains the vRealize Automation guest agent. For information about creating task sequences and installing software packages, see SCCM documentation.

4. Create a zero touch boot ISO image for the task sequence. By default, SCCM creates a light touch boot ISO image. For information about configuring SCCM for zero touch ISO images, see SCCM documentation.

5. Copy the ISO image to the location required by your virtualization platform. If you do not know the appropriate location, refer to the documentation provided by your hypervisor.

6. Gather the following information so that blueprint architects can include it on blueprints:

   a. The name of the collection containing the task sequence.

b. The fully qualified domain name of the SCCM server on which the collection containing the sequence resides.

c. The site code of the SCCM server.

d. Administrator-level credentials for the SCCM server.

e. (Optional) For SCVMM integrations, the ISO, virtual hard disk, or hardware profile to attach to provisioned machines.

Create a Software Package for SCCM Provisioning

The final step in your SCCM task sequence must be to install a software package that includes the vRealize Automation guest agent.

Procedure

1. Navigate to the vRealize Automation appliance management console page.

   For example: `https://`*`va-hostname.domain.com`*.

2. Click **Guest and software agents page** in the vRealize Automation component installation section of the page.

   For example: `https://`*`va-hostname.domain.com`*`/software/index.html`.

   The **Guest and Software Agent Installers** page opens, displaying links to available downloads.

3. Click Windows guest agent files (**32-bit**) or (**64-bit**) in the component installation section of the page to download and save the `GuestAgentInstaller.exe` or `GuestAgentInstaller_x64.exe` file.

4. Extract the Windows guest agent files to a location available to SCCM.

   This produces the directory `C:\VRMGuestAgent`. Do not rename this directory.

5. Create a software package from the definition file `SCCMPackageDefinitionFile.sms`.

6. Make the software package available to your distribution point.

7. Select the contents of the extracted Windows guest agent files as your source files.

Preparing for WIM Provisioning

Provision a machine by booting into a WinPE environment and then install an operating system using a Windows Imaging File Format (WIM) image of an existing Windows reference machine.

The following is a high-level overview of the steps required to prepare for WIM provisioning:

1. Identify or create the staging area. The staging area should be a network directory that can be specified as a UNC path or mounted as a network drive by

   - The reference machine.
   - The system where you build the WinPE image.

- The virtualization host where you provision the machines.

2 Ensure that the network has a DHCP server. vRealize Automation cannot provision machines with a WIM image unless DHCP is available.

3 Identify or create the reference machine in the virtualization platform you intend to use for provisioning. For vRealize Automation requirements, see Reference Machine Requirements for WIM Provisioning. For information about creating a reference machine, see the documentation provided by your hypervisor.

4 Using the System Preparation Utility for Windows, prepare the reference machine's operating system for deployment. See SysPrep Requirements for the Reference Machine.

5 Create the WIM image of the reference machine. Do not include any spaces in the WIM image file name or provisioning fails.

6 Create a WinPE image that contains the vRealize Automation guest agent.

   - (Optional) Create any custom scripts you want to use to customize provisioned machines and place them in the appropriate work item directory.
   - If you are using VirtIO for network or storage interfaces, you must ensure that the necessary drivers are included in your WinPE image and WIM image. See Preparing for WIM Provisioning with VirtIO Drivers.

   When you create the WinPE image, you must manually insert the vRealize Automation guest agent. See Manually Insert the Guest Agent into a WinPE Image.

7 Place the WinPE image in the location required by your virtualization platform. If you do not know the location, see your hypervisor documentation.

8 Gather the following information to include in the blueprint:

   a The name and location of the WinPE ISO image.

   b The name of the WIM file, the UNC path to the WIM file, and the index used to extract the desired image from the WIM file.

   c The user name and password under which to map the WIM image path to a network drive on the provisioned machine.

   d (Optional) If you do not want to accept the default, K, the drive letter to which the WIM image path is mapped on the provisioned machine.

   e For vCenter Server integrations, the vCenter Server guest operating system version with which vCenter Server is to create the machine.

f (Optional) For SCVMM integrations, the ISO, virtual hard disk, or hardware profile to attach to provisioned machines.

Note You can create a property group to include all of this required information. Using a property group makes it easier to include all the information correctly in blueprints.

Procedure

1 Reference Machine Requirements for WIM Provisioning

  WIM provisioning involves creating a WIM image from a reference machine. The reference machine must meet basic requirements for the WIM image to work for provisioning in vRealize Automation.

2 SysPrep Requirements for the Reference Machine

  A SysPrep answer file contains several required settings that are used for WIM provisioning.

3 Preparing for WIM Provisioning with VirtIO Drivers

  If you are using VirtIO for network or storage interfaces, you must ensure that the necessary drivers are included in your WinPE image and WIM image. VirtIO generally offers better performance when provisioning with KVM (RHEV).

4 Manually Insert the Guest Agent into a WinPE Image

  You must manually insert the vRealize Automation guest agent into your WinPE image.

Reference Machine Requirements for WIM Provisioning

WIM provisioning involves creating a WIM image from a reference machine. The reference machine must meet basic requirements for the WIM image to work for provisioning in vRealize Automation.

The following is a high-level overview of the steps to prepare a reference machine:

1 If the operating system on your reference machine is Windows Server 2008 R2, Windows Server 2012, Windows 7, or Windows 8, the default installation creates a small partition on the system's hard disk in addition to the main partition. vRealize Automation does not support the use of WIM images created on such multi-partitioned reference machines. You must delete this partition during the installation process.

2 Install NET 4.5 and Windows Automated Installation Kit (AIK) for Windows 7 (including WinPE 3.0) on the reference machine.

3 If the reference machine operating system is Windows Server 2003 or Windows XP, reset the administrator password to be blank. (There is no password.)

4 (Optional) If you want to enable XenDesktop integration, install and configure a Citrix Virtual Desktop Agent.

5 (Optional) A Windows Management Instrumentation (WMI) agent is required to collect certain data from a Windows machine managed by vRealize Automation, for example the Active Directory status of a machine's owner. To ensure successful management of Windows machines, you must install a WMI agent (typically on the Manager Service host) and enable the agent to collect data from Windows machines. See *Installing vRealize Automation*.

SysPrep Requirements for the Reference Machine

A SysPrep answer file contains several required settings that are used for WIM provisioning.

Table 3-15. Windows Server or Windows XP reference machine SysPrep required settings

`GuiUnattended` **Settings**	**Value**
`AutoLogon`	`Yes`
`AutoLogonCount`	`1`
`AutoLogonUsername`	*username* (*username* and *password* are the credentials used for auto logon when the newly provisioned machine boots into the guest operating system. Administrator is typically used.)
`AutoLogonPassword`	*password* corresponding to the `AutoLogonUsername`.

Table 3-16. Required SysPrep Settings for reference machine that are not using Windows Server 2003 or Windows XP:

`AutoLogon` **Settings**	**Value**
`Enabled`	`Yes`
`LogonCount`	`1`
`Username`	*username* (*username* and *password* are the credentials used for auto logon when the newly provisioned machine boots into the guest operating system. Administrator is typically used.)
`Password`	*password* (*username* and*password* are the credentials used for auto logon when the newly provisioned machine boots into the guest operating system. Administrator is typically used.) **Note** For reference machines that use a Windows platform newer than Windows Server 2003/Windows XP, you must set the autologon password by using the custom property `Sysprep.GuiUnattended.AdminPassword`. A convenient way to ensure this is done is to create a property group that includes this custom property so that tenant administrators and business group managers can include this information correctly in their blueprints.

Preparing for WIM Provisioning with VirtIO Drivers

If you are using VirtIO for network or storage interfaces, you must ensure that the necessary drivers are included in your WinPE image and WIM image. VirtIO generally offers better performance when provisioning with KVM (RHEV).

Windows drivers for VirtIO are included as part of the Red Hat Enterprise Virtualization and are located in the `/usr/share/virtio-win` directory on the file system of the Red Hat Enterprise Virtualization Manager. The drivers are also included in the Red Hat Enterprise Virtualization Guest Tools located `/usr/share/rhev-guest-tools-iso/rhev-tools-setup.iso`.

The high-level process for enabling WIM-based provisioning with VirtIO drivers is as follows:

1. Create a WIM image from a Windows reference machine with the VirtIO drivers installed or insert the drivers into an existing WIM image.
2. Copy the VirtIO driver files and insert the drivers into a WinPE image.
3. Upload the WinPE image ISO to the Red Hat Enterprise Virtualization ISO storage domains using the `rhevm-iso-uploader` command. For more information about managing ISO images in RHEV refer to the Red Hat documentation.
4. Create a KVM (RHEV) blueprint for WIM provisioning and select the WinPE ISO option. The custom property `VirtualMachine.Admin.DiskInterfaceType` must be included with the value **`VirtIO`**. A fabric administrator can include this information in a property group for inclusion on blueprints.

The custom properties `Image.ISO.Location` and `Image.ISO.Name` are not used for KVM (RHEV) blueprints.

Manually Insert the Guest Agent into a WinPE Image

You must manually insert the vRealize Automation guest agent into your WinPE image.

Prerequisites

- Select a Windows system from which the staging area you prepared is accessible and on which .NET 4.5 and Windows Automated Installation Kit (AIK) for Windows 7 (including WinPE 3.0) are installed.
- Create a WinPE.

Procedure

1. Install the Guest Agent in a WinPE

   You must manually copy the guest agent files to your WinPE image.

2. Configure the doagent.bat File

   You must manually configure the `doagent.bat` file.

3. Configure the doagentc.bat File

   You must manually configure the `doagentc.bat` file.

4 Configure the Guest Agent Properties Files

You must manually configure the guest agent properties files.

Procedure

1 Install the Guest Agent in a WinPE.

2 Configure the doagent.bat File.

3 Configure the doagentc.bat File.

4 Configure the Guest Agent Properties Files.

Install the Guest Agent in a WinPE

You must manually copy the guest agent files to your WinPE image.

Prerequisites

- Select a Windows system from which the staging area you prepared is accessible and on which .NET 4.5 and Windows Automated Installation Kit (AIK) for Windows 7 (including WinPE 3.0) are installed.
- Create a WinPE.

Procedure

◆ Download and install the vRealize Automation guest agent from `https://`*`vRealize_VA_Hostname_fqdn`*`/software/index.html`.

a Download `GugentZip_`*`version`* to the C drive on the reference machine.

Select either `GuestAgentInstaller.exe` (32-bit) or `GuestAgentInstaller_x64.exe` (64-bit) depending on which is appropriate for your operating system.

b Right-click the file and select **Properties**.

c Click **General**.

d Click **Unblock**.

e Extract the files to `C:\`.

This produces the directory `C:\VRMGuestAgent`. Do not rename this directory.

What to do next

Configure the doagent.bat File.

Configure the doagent.bat File

You must manually configure the `doagent.bat` file.

Prerequisites

Install the Guest Agent in a WinPE.

Procedure

1 Navigate to the VRMGuestAgent directory within your WinPE Image.

For example: C:\Program Files (x86)\VMware\Plugins\VRM Agent\VRMGuestAgent.

2 Make a copy of the file doagent-template.bat and name it doagent.bat.

3 Open doagent.bat in a text editor.

4 Replace all instances of the string #Dcac Hostname# with the fully qualified domain name and port number of the IaaS Manager Service host.

Option	Description
If you are using a load balancer	Enter the fully qualified domain name and port of the load balancer for the IaaS Manager Service. For example, `manager_service_LB.mycompany.com:443`
With no load balancer	Enter the fully qualified domain name and port of the machine on which the IaaS Manager Service is installed. For example, `manager_service.mycompany.com:443`

5 Replace all instances of the string #Protocol# with the string /ssl.

6 Replace all instances of the string #Comment# with REM (REM must be followed by a trailing space).

7 (Optional) If you are using self-signed certificates, uncomment the openSSL command.

```
echo QUIT | c:\VRMGuestAgent\bin\openssl s_client -connect
```

8 Save and close the file.

9 Edit the Startnet.cmd script for your WinPE to include the doagent.bat as a custom script.

What to do next

Configure the doagentc.bat File.

Configure the doagentc.bat File

You must manually configure the doagentc.bat file.

Prerequisites

Configure the doagent.bat File.

Procedure

1 Navigate to the VRMGuestAgent directory within your WinPE Image.

For example: C:\Program Files (x86)\VMware\Plugins\VRM Agent\VRMGuestAgent.

2 Make a copy of the file doagentsvc-template.bat and name it doagentc.bat.

3 Open `doagentc.bat` in a text editor.

4 Remove all instance of the string `#Comment#`.

5 Replace all instances of the string `#Dcac Hostname#` with the fully qualified domain name and port number of the Manager Service host.

   The default port for the Manager Service is 443.

Option	Description
If you are using a load balancer	Enter the fully qualified domain name and port of the load balancer for the Manager Service. For example, `load_balancer_manager_service.mycompany.com:443`
With no load balancer	Enter the fully qualified domain name and port of the Manager Service. For example, `manager_service.mycompany.com:443`

6 Replace all instances of the string `#errorlevel#` with the character `1`.

7 Replace all instances of the string `#Protocol#` with the string `/ssl`.

8 Save and close the file.

What to do next

Configure the Guest Agent Properties Files.

Configure the Guest Agent Properties Files

You must manually configure the guest agent properties files.

Prerequisites

Configure the doagentc.bat File.

Procedure

1 Navigate to the `VRMGuestAgent` directory within your WinPE Image.

   For example: `C:\Program Files (x86)\VMware\Plugins\VRM Agent\VRMGuestAgent`.

2 Make a copy of the file `gugent.properties` and name it `gugent.properties.template`.

3 Make a copy of the file `gugent.properties.template` and name it `gugentc.properties`.

4 Open `gugent.properties` in a text editor.

5 Replace all instances of the string `GuestAgent.log` the string `X:/VRMGuestAgent/GuestAgent.log`.

6 Save and close the file.

7 Open `gugentc.properties` in a text editor.

8 Replace all instances of the string `GuestAgent.log` the string `C:/VRMGuestAgent/GuestAgent.log`.

9 Save and close the file.

Preparing for Virtual Machine Image Provisioning

Before you provision instances with OpenStack, you must have virtual machine images and flavors configured in the OpenStack provider.

Virtual Machine Images

You can select an virtual machine image from a list of available images when creating blueprints for OpenStack resources.

A virtual machine image is a template that contains a software configuration, including an operating system. Virtual machine images are managed by the OpenStack provider and are imported during data collection.

If an image that is used in a blueprint is later deleted from the OpenStack provider, it is also removed from the blueprint. If all the images have been removed from a blueprint, the blueprint is deactivated and cannot be used for machine requests until it is edited to add at least one image.

OpenStack Flavors

You can select one or more flavors when creating OpenStack blueprints.

OpenStack flavors are virtual hardware templates that define the machine resource specifications for instances provisioned in OpenStack. Flavors are managed by the OpenStack provider and are imported during data collection.

Preparing for Amazon Machine Image Provisioning

Prepare your Amazon Machine Images and instance types for provisioning in vRealize Automation.

Understanding Amazon Machine Images

You can select an Amazon machine image from a list of available images when creating Amazon machine blueprints.

An Amazon machine image is a template that contains a software configuration, including an operating system. They are managed by Amazon Web Services accounts. vRealize Automation manages the instance types that are available for provisioning.

The Amazon machine image and instance type must be available in an Amazon region. Not all instance types are available in all regions.

You can select an Amazon machine image provided by Amazon Web Services, a user community, or the AWS Marketplace site. You can also create and optionally share your own Amazon machine images. A single Amazon machine image can be used to launch one or many instances.

The following considerations apply to Amazon machine images in the Amazon Web Services accounts from which you provision cloud machines:

- Each blueprint must specify an Amazon machine image.

  A private Amazon machine image is available to a specific account and all its regions. A public Amazon machine image is available to all accounts, but only to a specific region in each account.

- When the blueprint is created, the specified Amazon machine image is selected from regions that have been data-collected. If multiple Amazon Web Services accounts are available, the business group manager must have rights to any private Amazon machine images. The Amazon machine image region and the specified user location restrict provisioning request to reservations that match the corresponding region and location.

- Use reservations and policies to distribute Amazon machine images in your Amazon Web Services accounts. Use policies to restrict provisioning from a blueprint to a particular set of reservations.

- vRealize Automation cannot create user accounts on a cloud machine. The first time a machine owner connects to a cloud machine, she must log in as an administrator and add her vRealize Automation user credentials or an administrator must do that for her. She can then log in using her vRealize Automation user credentials.

  If the Amazon machine image generates the administrator password on every boot, the Edit Machine Record page displays the password. If it does not, you can find the password in the Amazon Web Services account. You can configure all Amazon machine images to generate the administrator password on every boot. You can also provide administrator password information to support users who provision machines for other users.

- To allow remote Microsoft Windows Management Instrumentation (WMI) requests on cloud machines provisioned in Amazon Web Services accounts, enable a Microsoft Windows Remote Management (WinRM) agent to collect data from Windows machines managed by vRealize Automation. See *Installing vRealize Automation*.

- A private Amazon machine image can be seen across tenants.

For related information, see *Amazon Machine Images (AMI)* topics in Amazon documentation.

Understanding Amazon Instance Types

An IaaS architect selects one or more Amazon instance types when creating Amazon EC2 blueprints. An IaaS administrator can add or remove instance types to control the choices available to the architects.

An Amazon EC2 instance is a virtual server that can run applications in Amazon Web Services. Instances are created from an Amazon machine image and by choosing an appropriate instance type.

To provision a machine in an Amazon Web Services account, an instance type is applied to the specified Amazon machine image. The available instance types are listed when architects create the Amazon EC2 blueprint. Architects select one or more instance types, and those instance types become choices available to the user when they request to provision a machine. The instance types must be supported in the designated region.

For related information, see *Selecting Instance Types* and *Amazon EC2 Instance Details* topics in Amazon documentation.

Add an Amazon Instance Type

Several instance types are supplied with vRealize Automation for use with Amazon blueprints. An administrator can add and remove instance types.

The machine instance types managed by IaaS administrators are available to blueprint architects when they create or edit an Amazon blueprint. Amazon machine images and instance types are made available through the Amazon Web Services product.

Prerequisites

Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1. Click **Infrastructure > Administration > Instance Types**.
2. Click **New**.
3. Add a new instance type, specifying the following parameters.

   Information about the available Amazon instances types and the setting values that you can specify for these parameters is available from Amazon Web Services documentation in *EC2 Instance Types - Amazon Web Services (AWS)* at aws.amazon.com/ec2 and *Instance Types* at docs.aws.amazon.com.

   - Name
   - API name
   - Type Name
   - IO Performance Name
   - CPUs
   - Memory (GB)
   - Storage (GB)
   - Compute Units
4. Click the **Save** icon (✓).

Results

When IaaS architects create Amazon Web Services blueprints, they can use your custom instance types.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Scenario: Prepare vSphere Resources for Machine Provisioning

As the vSphere administrator creating templates for vRealize Automation, you want to use the vSphere Web Client to prepare for cloning CentOS machines in vRealize Automation.

You want to convert an existing CentOS reference machine into a vSphere template so you and your architects can create blueprints for cloning CentOS machines in vRealize Automation. To prevent any conflicts that might arise from deploying multiple virtual machines with identical settings, you also want to create a general customization specification that you and your architects can use to create clone blueprints for Linux templates.

Prerequisites

Identify or create a Linux CentOS reference machine with VMware Tools installed. Include at least one Network Adapter to provide internet connectivity.

Procedure

1 Scenario: Convert Your CentOS Reference Machine into a Template for Rainpole

   Using the vSphere Client, you convert your existing CentOS reference machine into a vSphere template for your vRealize Automation IaaS architects to reference as the base for their clone blueprints.

2 Scenario: Create a Customization Specification for Cloning Linux Machines

   Using the vSphere Client, you create a standard customization specification for your vRealize Automation IaaS architects to use when they create clone blueprints for Linux machines.

Scenario: Convert Your CentOS Reference Machine into a Template for Rainpole

Using the vSphere Client, you convert your existing CentOS reference machine into a vSphere template for your vRealize Automation IaaS architects to reference as the base for their clone blueprints.

Procedure

1 Log in to your reference machine as the root user and prepare the machine for conversion.

   a Remove udev persistence rules.

   ```
   /bin/rm -f /etc/udev/rules.d/70*
   ```

   b Enable machines cloned from this template to have their own unique identifiers.

   ```
   /bin/sed -i '/^\(HWADDR\|UUID\)=/d'
   /etc/sysconfig/network-scripts/ifcfg-eth0
   ```

   c Power down the machine.

   ```
   shutdown -h now
   ```

2 Log in to the vSphere Web Client as an administrator.

3 Click the **VM Options** tab.

4 Right-click your reference machine and select **Edit Settings**.

5 Enter `Rainpole_centos_63_x86` in the **VM Name** text box.

6 Even though your reference machine has a CentOS guest operating system, select **Red Hat Enterprise Linux 6 (64-bit)** from the **Guest OS Version** drop-down menu.

   If you select CentOS, your template and customization specification might not work as expected.

7 Right-click your **Rainpole_centos_63_x86** reference machine in the vSphere Web Client and select **Template > Convert to Template**.

Results

vCenter Server marks your Rainpole_centos_63_x86 reference machine as a template and displays the task in the Recent Tasks pane.

What to do next

To prevent any conflicts that might arise from deploying multiple virtual machines with identical settings, you create a general customization specification that you and your Rainpole architects can use to create clone blueprints for Linux templates.

Scenario: Create a Customization Specification for Cloning Linux Machines

Using the vSphere Client, you create a standard customization specification for your vRealize Automation IaaS architects to use when they create clone blueprints for Linux machines.

Procedure

1 On the home page, click **Customization Specification Manager** to open the wizard.

2 Click the **New** icon.

3 Specify properties.

   a Select **Linux** from the **Target VM Operating System** drop-down menu.

   b Enter `Linux` in the **Customization Spec Name** text box.

   c Enter `Rainpole Linux cloning with vRealize Automation` in the **Description** text box.

   d Click **Next**.

4 Set computer name.

   a Select **Use the virtual machine name**.

   b Enter the domain on which cloned machines are going to be provisioned in the **Domain name** text box.

   c Click **Next**.

5 Configure time zone settings.

6 Click **Next**.

7 Select **Use standard network settings for the guest operating system, including enabling DHCP on all network interfaces**.

8 Follow the prompts to enter the remaining required information.

9 On the **Ready to complete** page, review your selections and click **Finish**.

Preparing for Software Provisioning

Use Software to deploy applications and middleware as part of the vRealize Automation provisioning process for vSphere, vCloud Director,vCloud Air, Amazon Web Services, and Microsoft Azure machines.

You can deploy Software on machines if your blueprint supports Software and if you install the guest agent and software bootstrap agent on your reference machines before you convert them into templates, snapshots, or machine images.

For related information about specifying ports when preparing to provision machines, see the Reference Architecture section of the product documentation.

Table 3-17. Provisioning Methods that Support Software

Machine Type	Preparation
vSphere	A clone blueprint provisions a complete and independent virtual machine based on a vCenter Server virtual machine template. If you want your templates for cloning to support Software components, install the guest agent and software bootstrap agent on your reference machine as you prepare a template for cloning. See Checklist for Preparing to Provision by Cloning.
vSphere	A linked clone blueprint provisions a space-efficient copy of a vSphere machine based on a snapshot, using a chain of delta disks to track differences from the parent machine. If you want your linked clone blueprints to support Software components, install the guest agent and software bootstrap agent on the machine before you take the snapshot. If your snapshot machine was cloned from a template that supports Software, the required agents are already installed.
vCloud Director	A clone blueprint provisions a complete and independent virtual machine based on a vCenter Server virtual machine template. If you want your templates for cloning to support Software components, install the guest agent and software bootstrap agent on your reference machine as you prepare a template for cloning. See Checklist for Preparing to Provision by Cloning.
vCloud Air	A clone blueprint provisions a complete and independent virtual machine based on a vCenter Server virtual machine template. If you want your templates for cloning to support Software components, install the guest agent and software bootstrap agent on your reference machine as you prepare a template for cloning. See Checklist for Preparing to Provision by Cloning.
Amazon Web Services	An Amazon machine image is a template that contains a software configuration, including an operating system. If you want to create an Amazon machine image that supports Software, connect to a running Amazon Web Services instance that uses an EBS volume for the root device. Install the guest agent and software bootstrap agent on the reference machine, then create an Amazon Machine Image from your instance. For the guest agent and Software bootstrap agent to function on provisioned machines, you must configure network-to-VPC connectivity. For information about creating Amazon EBS-backed AMIs, see Amazon Web Services documentation.
Microsoft Azure	For information, see Software Component Settings, Create a Blueprint for Microsoft Azure, and Microsoft Azure product documentation.

Preparing to Provision Machines with Software

To support Software components, you must install the guest agent and Software bootstrap agent on your reference machine before you convert to a template for cloning, create an Amazon machine image, or take a snapshot.

Prepare a Windows Reference Machine to Support Software

You use a single script to install the Java Runtime Environment, guest agent, and Software bootstrap agent on a Windows reference machine. From the reference machine, you can create a template for cloning, a snapshot, or an Amazon machine image that supports Software components.

Software supports scripting with Windows CMD and PowerShell 2.0.

Important The startup process must not be interrupted. Configure the virtual machine so that nothing pauses the virtual machine startup process before reaching the login prompt. For example, verify that no processes or scripts prompt for user interaction while the virtual machine starts.

Prerequisites

- Identify or create a Windows reference machine.
- Establish secure trust between the reference machine and your IaaS Manager Service host. See Configuring the Guest Agent to Trust a Server.
- If you plan to remotely access the machine for troubleshooting or other reasons, install Remote Desktop Services (RDS).
- Remove network configuration artifacts from the network configuration files.

Procedure

1. Log in to the Windows reference server as an administrator.
2. Open a browser to the software download page on the vRealize Automation appliance.

   https://*vrealize-automation-appliance-FQDN*/software
3. Save the template ZIP to the Windows server.

   ```
   prepare_vra_template_windows.zip
   ```
4. Extract the ZIP contents to a folder, and run the batch file.

   ```
   .\prepare_vra_template.bat
   ```
5. Follow the prompts.
6. When finished, shut down the Windows virtual machine.

Results

The script removes any previous guest or Software bootstrap agents, and installs the supported versions of the Java Runtime Environment, the guest agent, and the Software bootstrap agent.

What to do next

Convert the reference machine into a template for cloning, a snapshot, or an Amazon machine image. Each supports Software components, and infrastructure architects can use them when creating blueprints.

Prepare a Linux Reference Machine to Support Software

You use a single script to install the Java Runtime Environment, guest agent, and Software bootstrap agent on your Linux reference machine. From the reference machine, you can create a template for cloning, a snapshot, or an Amazon machine image that supports Software components.

Software supports scripting with Bash.

Important The boot process must not be interrupted. Configure the virtual machine so that nothing pauses the virtual machine boot process before reaching the login prompt. For example, verify that no processes or scripts prompt for user interaction while the virtual machine starts.

Prerequisites

- Identify or create a Linux reference machine.
- Verify that the following commands are available, depending on your Linux system:
  - `yum` or `apt-get`
  - `wget` or `curl`
  - `python`
  - `dmidecode` as required by cloud providers
  - Common requirements such as `sed`, `awk`, `perl`, `chkconfig`, `unzip`, and `grep` depending on your Linux distribution

  You might also use an editor to inspect the downloaded `prepare_vra_template.sh` script, which exposes the commands that it uses.
- If you plan to remotely access the machine for troubleshooting or other reasons, install OpenSSH.
- Remove network configuration artifacts from the network configuration files.

Procedure

1. Log in to your reference machine as root.
2. Download the template tar.gz package from the vRealize Automation appliance.

   ```
   wget https://vrealize-automation-appliance-FQDN/software/download/
   prepare_vra_template_linux.tar.gz
   ```

   If your environment is using self-signed certificates, you might need the `--no-check-certificate` option.

   ```
   wget --no-check-certificate https://vrealize-automation-appliance-FQDN/software/download/
   prepare_vra_template_linux.tar.gz
   ```
3. Untar the package.

   ```
   tar -xvf prepare_vra_template_linux.tar.gz
   ```
4. In the untar output, find the installer script, and make it executable.

   ```
   chmod +x prepare_vra_template.sh
   ```

5 Run the installer script.

```
./prepare_vra_template.sh
```

If you need information about non-interactive options and expected values, see the script help.

```
./prepare_vra_template.sh --help
```

6 Follow the prompts.

A confirmation appears when installation succeeds. If errors and logs appear, resolve the errors and rerun the script.

7 When finished, shut down the Linux virtual machine.

Results

The script removes any previous guest or Software bootstrap agents, and installs the supported versions of the Java Runtime Environment, the guest agent, and the Software bootstrap agent.

What to do next

On your hypervisor or cloud provider, turn the reference machine into a template for cloning, a snapshot, or an Amazon machine image. Each supports Software components, and infrastructure architects can use them when creating blueprints.

Updating Existing Virtual Machine Templates in vRealize Automation

If you are updating your templates, Amazon Machine Images, or snapshots for the latest version of the Windows Software bootstrap agent, or if you are manually updating to the latest Linux Software bootstrap agent instead of using the `prepare_vra_template.sh script`, you need to remove any existing versions and delete any logs.

Linux

For Linux reference machines, running the `prepare_vra_template.sh script` script resets the agent and removes any logs for you before reinstalling. However, if you intend to manually install, you need to log in to the reference machine as the root user and run the command to reset and remove the artifacts.

```
/opt/vmware-appdirector/agent-bootstrap/agent_reset.sh
```

Windows

For Windows reference machines, you remove the existing Software agent bootstrap and vRealize Automation 6.0 or later guest agent, and delete any existing runtime log files. In a PowerShell command window, run the commands to remove the agent and artifacts.

```
c:\opt\vmware-appdirector\agent-bootstrap\appd_bootstrap_removal.bat
```

Prepare a vSphere Template for Clone Machine and Software Component Blueprints

As a vCenter Server administrator, you want to prepare a vSphere template that your vRealize Automation architects can use to clone, for example, Linux CentOS machines. You want to ensure that your template supports blueprints with software components, so you install the guest agent and the software bootstrap agent before you turn your reference machine into a template.

Prerequisites

- Identify or create a Linux CentOS reference machine with VMware Tools installed. Include at least one Network Adapter to provide internet connectivity in case blueprint architects do not add this functionality at the blueprint level. For information about creating virtual machines, see the vSphere documentation.
- You must be connected to a vCenter Server to convert a virtual machine to a template. You cannot create templates if you connect the vSphere Client directly to an vSphere ESXi host.

Procedure

1. Scenario: Prepare Your Reference Machine for Guest Agent Customizations and Software Components

   So that your template can support software components, you install the software bootstrap agent and its prerequisite, the guest agent, on your reference machine. The agents ensure that vRealize Automation architects who use your template can include software components in their blueprints.

2. Scenario: Convert Your CentOS Reference Machine into a Template

   After you install the guest agent and software bootstrap agent onto your reference machine, you turn your reference machine into a template that vRealize Automation architects can use to create clone machine blueprints.

3. Scenario: Create a Customization Specification for vSphere Cloning

   Create a customization specification for your blueprint architects to use with your cpb_centos_63_x84 template.

Results

You created a template and customization specification from your reference machine that blueprint architects can use to create vRealize Automation blueprints that clone Linux CentOS machines. Because you installed the Software bootstrap agent and the guest agent on your reference machine, architects can use your template to create elaborate catalog item blueprints that include Software components or other guest agent customizations such as running scripts or formatting disks. Because you installed VMware Tools, architects and catalog administrators can allow users to perform actions against machines, such as reconfigure, snapshot, and reboot.

What to do next

After you configure vRealize Automation users, groups, and resources, you can use your template and customization specification to create a machine blueprint for cloning. See Configure a Machine Blueprint.

Scenario: Prepare Your Reference Machine for Guest Agent Customizations and Software Components

So that your template can support software components, you install the software bootstrap agent and its prerequisite, the guest agent, on your reference machine. The agents ensure that vRealize Automation architects who use your template can include software components in their blueprints.

To simplify the process, you download and run a vRealize Automation script that installs both agents, instead of downloading and installing separate packages.

The script also connects to the Manager Service instance and downloads the SSL certificate, which establishes trust between the Manager Service and machines deployed from the template. Note that having the script download the certificate is less secure than manually obtaining the Manager Service SSL certificate and installing it on your reference machine in `/usr/share/gugent/cert.pem`.

Procedure

1 Open a browser to the vRealize Automation appliance software page.

   https://*vrealize-automation-appliance-FQDN*/software

2 Under Linux Software Installers, download the gzipped tar file.

   ```
   prepare_vra_template_linux.tar.gz
   ```

3 Move the tar file to a temporary directory on the Linux reference machine.

   To transfer the file, you can run a tool such as WinSCP, or use any other method with which you are familiar.

4 Log in as root to the command prompt on the Linux reference machine.

   To open a terminal, you can start Remote Console on the machine from within vRealize Automation, or use any other method with which you are familiar.

5 From the temporary directory, extract the tar file.

   ```
   gunzip prepare_vra_template_linux.tar.gz
   ```

6 Extract the tar file contents.

   ```
   tar xvf prepare_vra_template_linux.tar
   ```

7 Change to the script directory.

   ```
   cd prepare_vra_template_linux
   ```

8. Run the script, and follow the prompts.

   ```
   ./prepare_vra_template.sh
   ```

   If you need non-interactive information about options and values, enter `./prepare_vra_template.sh --help`.

Results

A confirmation message appears when installation finishes. If error messages and logs appear, correct the issues and rerun the script.

Scenario: Convert Your CentOS Reference Machine into a Template

After you install the guest agent and software bootstrap agent onto your reference machine, you turn your reference machine into a template that vRealize Automation architects can use to create clone machine blueprints.

After you convert your reference machine to a template, you cannot edit or power on the template unless you convert it back to a virtual machine.

Procedure

1. Log in to your reference machine as the root user and prepare the machine for conversion.

   a. Remove udev persistence rules.

      ```
      /bin/rm -f /etc/udev/rules.d/70*
      ```

   b. Enable machines cloned from this template to have their own unique identifiers.

      ```
      /bin/sed -i '/^\(HWADDR\|UUID\)=/d'
      /etc/sysconfig/network-scripts/ifcfg-eth0
      ```

   c. If you rebooted or reconfigured the reference machine after installing the software bootstrap agent, reset the agent.

      ```
      /opt/vmware-appdirector/agent-bootstrap/agent_reset.sh
      ```

   d. Power down the machine.

      ```
      shutdown -h now
      ```

2. Log in to the vSphere Web Client as an administrator.

3. Right-click your reference machine and select **Edit Settings**.

4. Enter `cpb_centos_63_x84` in the **VM Name** text box.

5. Even though your reference machine has a CentOS guest operating system, select **Red Hat Enterprise Linux 6 (64-bit)** from the **Guest OS Version** drop-down menu.

   If you select CentOS, your template and customization specification might not work as expected.

6 Right-click your reference machine in the vSphere Web Client and select **Template > Convert to Template**.

Results

vCenter Server marks your cpb_centos_63_x84 reference machine as a template and displays the task in the Recent Tasks pane. If you have already brought your vSphere environment under vRealize Automation management, your template is discovered during the next automated data collection. If you have not configured your vRealize Automation yet, the template is collected during that process.

Scenario: Create a Customization Specification for vSphere Cloning

Create a customization specification for your blueprint architects to use with your cpb_centos_63_x84 template.

Procedure

1 Log in to the vSphere Web Client as an administrator.

2 On the home page, click **Customization Specification Manager** to open the wizard.

3 Click the **New** icon.

4 Click the **New** icon.

5 Specify properties.

   a Select **Linux** from the **Target VM Operating System** drop-down menu.

   b Enter `Customspecs` in the **Customization Spec Name** text box.

   c Enter **`cpb_centos_63_x84 cloning with vRealize Automation`** in the **Description** text box.

   d Click **Next**.

6 Set computer name.

   a Select **Use the virtual machine name**.

   b Enter the domain on which cloned machines are going to be provisioned in the **Domain name** text box.

   c Click **Next**.

7 Configure time zone settings.

8 Click **Next**.

9 Select **Use standard network settings for the guest operating system, including enabling DHCP on all network interfaces**.

   Fabric administrators and infrastructure architects handle network settings for provisioned machine by creating and using Network profiles in vRealize Automation.

10 Follow the prompts to enter the remaining required information.

11 On the **Ready to complete** page, review your selections and click **Finish**.

Results

Scenario: Prepare for Importing the Dukes Bank for vSphere Sample Application Blueprint

As a vCenter Server administrator, you want to prepare a vSphere CentOS 6.x Linux template and customization specification that you can use to provision the vRealize Automation Dukes Bank sample application.

You want to ensure that your template supports the sample application software components, so you install the guest agent and the software bootstrap agent onto your Linux reference machine before you convert it to a template and create a customization specification. You deactivate SELinux on your reference machine to ensure your template supports the specific implementation of MySQL used in the Dukes Bank sample application.

Prerequisites

- Identify or create a CentOS 6.x Linux reference machine with VMware Tools installed. For information about creating virtual machines, see the vSphere documentation.
- You must be connected to a vCenter Server to convert a virtual machine to a template. You cannot create templates if you connect the vSphere Client directly to an vSphere ESXi host.

Procedure

1. Scenario: Prepare Your Reference Machine for the Dukes Bank vSphere Sample Application

   You want your template to support the Dukes Bank sample application, so you must install both the guest agent and the software bootstrap agent on your reference machine so vRealize Automation can provision the software components. To simplify the process, you download and run a vRealize Automation script that installs both the guest agent and the software bootstrap agent instead of downloading and installing the packages separately.

2. Scenario: Convert Your Reference Machine into a Template for the Dukes Bank vSphere Application

   After you install the guest agent and software bootstrap agent on your reference machine, you deactivate SELinux to ensure your template supports the specific implementation of MySQL used in the Dukes Bank sample application. You turn your reference machine into a template that you can use to provision the Dukes Bank vSphere sample application.

3. Scenario: Create a Customization Specification for Cloning the Dukes Bank vSphere Sample Application Machines

   You create a customization specification to use with your Dukes Bank machine template.

Results

You created a template and customization specification from your reference machine that supports the vRealize Automation Dukes Bank sample application.

Scenario: Prepare Your Reference Machine for the Dukes Bank vSphere Sample Application

You want your template to support the Dukes Bank sample application, so you must install both the guest agent and the software bootstrap agent on your reference machine so vRealize Automation can provision the software components. To simplify the process, you download and run a vRealize Automation script that installs both the guest agent and the software bootstrap agent instead of downloading and installing the packages separately.

Procedure

1 Log in to your reference machine as the root user.

2 Download the installation script from your vRealize Automation appliance.

```
wget  https://vRealize_VA_Hostname_fqdn/software/download/prepare_vra_template.sh
```

If your environment is using self-signed certificates, you might have to use the wget option `--no-check-certificate` option. For example:

```
wget --no-check-certificate https://vRealize_VA_Hostname_fqdn/software/download/
prepare_vra_template.sh
```

3 Make the `prepare_vra_template.sh` script executable.

```
chmod +x prepare_vra_template.sh
```

4 Run the `prepare_vra_template.sh` installer script.

```
./prepare_vra_template.sh
```

You can run the help command `./prepare_vra_template.sh --help` for information about non-interactive options and expected values.

5 Follow the prompts to complete the installation.

You see a confirmation message when the installation is successfully completed. If you see an error message and logs in the console, resolve the errors and run the installer script again.

Results

You installed both the software bootstrap agent and its prerequisite, the guest agent, to ensure the Dukes Bank sample application successfully provisions software components. The script also connected to your Manager Service instance and downloaded the SSL certificate to establish trust between the Manager Service and machines deployed from your template. This is a less secure approach than obtaining the Manager Service SSL certificate and manually installing it on your reference machine in `/usr/share/gugent/cert.pem`, and you can manually replace this certificate now if security is a high priority.

Scenario: Convert Your Reference Machine into a Template for the Dukes Bank vSphere Application

After you install the guest agent and software bootstrap agent on your reference machine, you deactivate SELinux to ensure your template supports the specific implementation of MySQL used in the Dukes Bank sample application. You turn your reference machine into a template that you can use to provision the Dukes Bank vSphere sample application.

After you convert your reference machine to a template, you cannot edit or power on the template unless you convert it back to a virtual machine.

Procedure

1 Log in to your reference machine as the root user.

   a Edit your `/etc/selinux/config` file to deactivate SELinux.

   ```
   SELINUX=disabled
   ```

   If you do not deactivate SELinux, the MySQL software component of the Duke's Bank Sample application might not work as expected.

   b Remove udev persistence rules.

   ```
   /bin/rm -f /etc/udev/rules.d/70*
   ```

   c Enable machines cloned from this template to have their own unique identifiers.

   ```
   /bin/sed -i '/^\(HWADDR\|UUID\)=/d'
   /etc/sysconfig/network-scripts/ifcfg-eth0
   ```

   d If you rebooted or reconfigured the reference machine after installing the software bootstrap agent, reset the agent.

   ```
   /opt/vmware-appdirector/agent-bootstrap/agent_reset.sh
   ```

   e Power down the machine.

   ```
   shutdown -h now
   ```

2 Log in to the vSphere Web Client as an administrator.

3 Right-click your reference machine and select **Edit Settings**.

4 Enter `dukes_bank_template` in the **VM Name** text box.

5 If your reference machine has a CentOS guest operating system, select **Red Hat Enterprise Linux 6 (64-bit)** from the **Guest OS Version** drop-down menu.

   If you select CentOS, your template and customization specification might not work as expected.

6 Click **OK**.

7 Right-click your reference machine in the vSphere Web Client and select **Template > Convert to Template**.

Results

vCenter Server marks your dukes_bank_template reference machine as a template and displays the task in the Recent Tasks pane. If you have already brought your vSphere environment under vRealize Automation management, your template is discovered during the next automated data collection. If you have not configured your vRealize Automation yet, the template is collected during that process.

Scenario: Create a Customization Specification for Cloning the Dukes Bank vSphere Sample Application Machines

You create a customization specification to use with your Dukes Bank machine template.

Procedure

1 Log in to the vSphere Web Client as an administrator.

2 On the home page, click **Customization Specification Manager** to open the wizard.

3 Click the **New** icon.

4 Specify properties.

   a Select **Linux** from the **Target VM Operating System** drop-down menu.

   b Enter `Customspecs_sample` in the **Customization Spec Name** text box.

   c Enter `Dukes Bank customization spec` in the **Description** text box.

   d Click **Next**.

5 Set computer name.

   a Select **Use the virtual machine name**.

   b Enter the domain on which you want to provision the Dukes Bank sample application in the **Domain name** text box.

   c Click **Next**.

6 Configure time zone settings.

7 Click **Next**.

8 Select **Use standard network settings for the guest operating system, including enabling DHCP on all network interfaces**.

   Fabric administrators and infrastructure architects handle network settings for provisioned machine by creating and using Network profiles in vRealize Automation.

9 Follow the prompts to enter the remaining required information.

10 On the **Ready to complete** page, review your selections and click **Finish**.

Results

You created a template and customization specification that you can use to provision the Dukes Bank sample application.

What to do next

1 Create an external network profile to provide a gateway and a range of IP addresses. See Create an External Network Profile by Using A Third-Party IPAM Provider.

2 Map your external network profile to your vSphere reservation. See Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer. The sample application cannot provision successfully without an external network profile.

3 Import the Duke's Bank sample application into your environment. See Scenario: Importing the Dukes Bank for vSphere Sample Application and Configuring for Your Environment.

Tenant and Resource Preparations for Blueprint Provisioning

4

You can configure multiple tenant environments, each with their own groups of users and unique access to resources that you bring under vRealize Automation management.

This chapter includes the following topics:

- Configuring Tenant Settings
- Configuring Resources
- User Preferences for Notifications and Delegates

Configuring Tenant Settings

Tenant administrators configure tenant settings such as user authentication, and manage user roles and business groups. System administrators and tenant administrators configure options such as email servers to handle notifications, and branding for the vRealize Automation console.

You can use the Configuring Tenant Settings Checklist to see a high-level overview of the sequence of steps required to configure tenant settings.

Table 4-1. Checklist for Configuring Tenant Settings

Task	vRealize Automation Role	Details
❑ Create local user accounts and assign a tenant administrator.	System administrator	Configure Access to the Default Tenant
❑ Configure Directories Management to set up tenant identity management and access control settings.	Tenant administrator	Choosing Directories Management Configuration Options
❑ Create business groups and custom groups, and grant user access rights to the vRealize Automation console.	Tenant administrator	Configuring Groups and User Roles
❑ (Optional) Create additional tenants so users can access the appropriate applications and resources they need to complete their work assignments.	System administrator	Create Additional Tenants

Table 4-1. Checklist for Configuring Tenant Settings (continued)

Task	vRealize Automation Role	Details
❑ (Optional) Configure custom branding on the tenant login and application pages of the vRealize Automation console.	■ System administrator ■ Tenant administrator	Configuring Custom Branding
❑ (Optional) Configure vRealize Automation to send users notifications when specific events occur.	■ System administrator ■ Tenant administrator	Checklist for Configuring Notifications
❑ (Optional) Configure vRealize Orchestrator to support XaaS and other extensibility.	■ System administrator ■ Tenant administrator	Configuring vRealize Orchestrator
❑ (Optional) Create a custom remote desktop protocol file that IaaS architects use in blueprints to configure RDP settings.	System administrator	Create a Custom RDP File to Support RDP Connections for Provisioned Machines
❑ (Optional) Define datacenter locations that your fabric administrators and IaaS architects can leverage to allow users to select an appropriate location for provisioning when they request machines.	System administrator	For an example of adding datacenter locations, see Scenario: Add Datacenter Locations for Cross Region Deployments .

Choosing Directories Management Configuration Options

You can use vRealize Automation Directories Management features to configure an Active Directory link in accordance with your user authentication requirements.

Directories Management provides many options to support a highly customized user authentication.

Table 4-2. Choosing Directories Management Configuration Options

Configuration Option	Procedure
Configure a link to your Active Directory.	1 Configure a link to your Active Directory. See Configure an Active Directory over LDAP/IWA Link . 2 If you configured vRealize Automation for high availability, see Configure Directories Management for High Availability.
(Optional) Enhance security of a user ID and password based directory link by configuring bi-directional integration with Active Directory Federated Services.	Configure a Bi Directional Trust Relationship Between vRealize Automation and Active Directory
(Optional) Add users and groups to an existing Active Directory Link .	Add Users or Groups to an Active Directory Connection.

Table 4-2. Choosing Directories Management Configuration Options (continued)

Configuration Option	Procedure
(Optional) Edit the default policy to apply custom rules for an Active Directory link.	Manage the User Access Policy.
(Optional) Configure network ranges to restrict the IP addresses through which users can log in to the system, manage login restrictions (timeout, number of login attempts before lock-out).	Add or Edit a Network Range.

Directories Management Overview

Tenant administrators can configure tenant identity management and access control settings using the Directories Management options on the vRealize Automation application console.

You can manage the following settings from the **Administration > Directories Management** tab.

Table 4-3. Directories Management Settings

Setting	Description
Directories	The Directories page enables you to create and manage Active Directory links to support vRealize Automation tenant user authentication and authorization. You create one or more directories and then sync those directories with your Active Directory deployment. This page displays the number of groups and users that are synced to the directory and the last sync time. You can click **Sync Now**, to manually start the directory sync. See Using Directories Management to Create an Active Directory Link. When you click on a directory and then click the **Sync Settings** button, you can edit the sync settings, navigate the Identity Providers page, and view the sync log. From the directories sync settings page you can schedule the sync frequency, see the list of domains associated with this directory, change the mapped attributes list, update the user and groups list that syncs, and set the safeguard targets.
Connectors	The Connectors page lists deployed connectors for your enterprise network. A connector syncs user and group data between Active Directory and the Directories Management service, and when it is used as the identity provider, authenticates users to the service. Each vRealize Automation appliance contains a connector by default. See Managing Connectors and Connector Clusters.
User Attributes	The User Attributes page lists the default user attributes that sync in the directory and you can add other attributes that you can map to Active Directory attributes. See Select Attributes to Sync with Directory .
Network Ranges	This page lists the network ranges that are configured for your system. You configure a network range to allow users access through those IP addresses. You can add additional network ranges and you can edit existing ranges. See Add or Edit a Network Range.

Table 4-3. Directories Management Settings (continued)

Setting	Description
Identity Providers	The Identity Providers page lists identity providers that are available on your system. vRealize Automation systems contain a connector that serves as the default identity provider and that suffices for many user needs. You can add third-party identity provider instances or have a combination of both. See Configure a Third Party Identity Provider Connection.
Policies	The Policies page lists the default access policy and any other web application access policies you created. Policies are a set of rules that specify criteria that must be met for users to access their application portals or to launch Web applications that are enabled for them. The default policy should be suitable for most vRealize Automation deployments, but you can edit it if needed. See Manage the User Access Policy.

Important Concepts Related to Active Directory

Several concepts related to Active Directory are integral to understanding how Directories Management integrates with your Active Directory environments.

Connector

The connector, a component of the service, performs the following functions.

- Syncs user and group data between Active Directory and the service.
- When being used as an identity provider, authenticates users to the service.

  The connector is the default identity provider. For the authentication methods the connector supports, see *VMware Identity Manager Administration*. You can also use third-party identity providers that support the SAML 2.0 protocol. Use a third-party identity provider for an authentication type the connector does not support or for an authentication type the connector does support, if the third-party identity provider is preferable based on your enterprise security policy.

  Note Even if you use third-party identity providers, you must configure the connector to sync user and group data.

Directory

The Directories Management service has its own concept of a directory, which uses Active Directory attributes and parameters to define users and groups. You create one or more directories and then sync those directories with your Active Directory deployment. You can create the following directory types in the service.

- Active Directory over LDAP. Create this directory type if you plan to connect to a single Active Directory domain environment. For the Active Directory over LDAP directory type, the connector binds to Active Directory using simple bind authentication.
- Active Directory, Integrated Windows Authentication. Create this directory type if you plan to connect to a multi-domain or multi-forest Active Directory environment. The connector binds to Active Directory using Integrated Windows Authentication.

The type and number of directories that you create varies depending on your Active Directory environment, such as single domain or multi-domain, and on the type of trust used between domains. In most environments, you create one directory.

The service does not have direct access to Active Directory. Only the connector has direct access to Active Directory. Therefore, you associate each directory created in the service with a connector instance.

Worker

When you associate a directory with a connector instance, the connector creates a partition for the associated directory called a worker. A connector instance can have multiple workers associated with it. Each worker acts as an identity provider. You define and configure authentication methods per worker.

The connector syncs user and group data between Active Directory and the service through one or more workers.

You cannot have two workers of the Integrated Windows Authentication type on the same connector instance.

Active Directory Environments

You can integrate the service with an Active Directory environment that consists of a single Active Directory domain, multiple domains in a single Active Directory forest, or multiple domains across multiple Active Directory forests.

Single Active Directory Domain Environment

A single Active Directory deployment allows you to sync users and groups from a single Active Directory domain.

See Configure an Active Directory over LDAP/IWA Link . For this environment, when you add a directory to the service, select the Active Directory over LDAP option.

Multi-Domain, Single Forest Active Directory Environment

A multi-domain, single forest Active Directory deployment allows you to sync users and groups from multiple Active Directory domains within a single forest.

You can configure the service for this Active Directory environment as a single Active Directory, Integrated Windows Authentication directory type or, alternatively, as an Active Directory over LDAP directory type configured with the global catalog option.

- The recommended option is to create a single Active Directory, Integrated Windows Authentication directory type.

  See Configure an Active Directory over LDAP/IWA Link . When you add a directory for this environment, select the Active Directory (Integrated Windows Authentication) option.

Multi-Forest Active Directory Environment with Trust Relationships

A multi-forest Active Directory deployment with trust relationships allows you to sync users and groups from multiple Active Directory domains across forests where two-way trust exists between the domains.

See Configure an Active Directory over LDAP/IWA Link . When you add a directory for this environment, select the Active Directory (Integrated Windows Authentication) option.

Multi-Forest Active Directory Environment Without Trust Relationships

A multi-forest Active Directory deployment without trust relationships allows you to sync users and groups from multiple Active Directory domains across forests without a trust relationship between the domains. In this environment, you create multiple directories in the service, one directory for each forest.

See Configure an Active Directory over LDAP/IWA Link . The type of directories you create in the service depends on the forest. For forests with multiple domains, select the Active Directory (Integrated Windows Authentication) option. For a forest with a single domain, select the Active Directory over LDAP option.

Using Directories Management to Create an Active Directory Link

After you create vRealize Automation tenants, you must log in to the system console as a tenant administrator and create an Active Directory link to support user authentication.

There are three Active Directory communication protocol options when configuring an Active Directory connection using Directories Management.

- Active Directory over LDAP - An Active Directory over LDAP protocol supports DNS Service Location lookup by default.
- Active Directory (Integrated Windows Authentication) - With Active Directory (Integrated Windows Authentication), you configure the domain to join. Active Directory over LDAP is appropriate for single domain deployments. Use Active Directory (Integrated Windows Authentication) for all multi-domain and multi-forest deployments.
- OpenLDAP - You can use the open source version of LDAP to support Directories Management user authentication.

After you select a communication protocol and configure an Active Directory link, you can specify the domains to use with the Active Directory configuration and then select the users and groups to sync with the specified configuration.

Configure an Active Directory over LDAP/IWA Link

You can configure an Active Directory over LDAP/IWA link to support user authentication using the Directories Management feature to configure a link to Active Directory to support user authentication for all tenants and select users and groups to sync with the Directories Management directory.

For information and instructions about using OpenLDAP with Directories Management, see Configure an OpenLDAP Directory Connection.

For Active Directory (Integrated Windows Authentication), when you have multi-forest Active Directory configured and the Domain Local group contains members from domains in different forests, make sure that the Bind user is added to the Administrators group of the domain in which the Domain Local group resides. If you fail to do this, these members will be missing from the Domain Local group.

Note Configure Active Directory IWA directories for the default tenant first, and then you can add them to other tenants.

Prerequisites

- Select the required default attributes and add additional attributes on the User Attributes page. See Select Attributes to Sync with Directory .
- List of the Active Directory groups and users to sync from Active Directory.
- If your Active Directory requires access over SSL or STARTTLS, the Root CA certificate of the Active Directory domain controller is required.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1. Select **Administration > Directories Management > Directories**.
2. Click **Add Directory** and select **Add Active Directory over LDAP/IWA**.
3. On the Add Directory page, specify the IP address for the Active Directory server in the **Directory Name** text box.
4. Select the appropriate Active Directory communication protocol using the radio buttons under the **Directory Name** text box.

Option	Description
Windows Authentication	Select **Active Directory (Integrated Windows Authentication)**. For Active Directory Integrated Windows Authentication, required information includes the domain's Bind user UPN address and password.
LDAP	Select **Active Directory over LDAP**. For Active Directory over LDAP, information required includes the Base DN, Bind DN, and Bind DN password.

5 Configure the connector that synchronizes users from the Active Directory to the VMware Directories Management directory in the Directory Sync and Authentication section.

Option	Description
Sync Connector	Select the appropriate connector to use for your system. Each vRealize Automation appliance contains a default connector. Consult your system administrator if you need help in choosing the appropriate connector.
Authentication	Click the appropriate radio button to indicate whether the selected connector also performs authentication. If you are using Active Directory (Integrated Windows Authentication), with a third party identity provider to authenticate users, click **No**. After you configure the Active Directory connection to sync users and groups use the Identity Providers page to add the third-party identity provider for authentication. For information about using authentication adapters such as PasswordIpddAdapter, SecurIDAdapter, and RadiusAuthAdapter, see the *VMware Identity Manager Administration Guide*.
Directory Search Attribute	Select the appropriate account attribute that contains the user name. VMware recommends using the sAMAccount attribute rather than userPrincipleName. If you use userPrincipleName for sync operations, integration with second and third party software that requires a user name may not function correctly. **Note** If you select sAMAccountName when using a global catalog, indicated by selecting the**This Directory has a Global Catalog** check box in the Server Location area, users will be unable to log in.

6 Enter the appropriate information in the Server Location text box if you selected Active Directory over LDAP, or enter information in the Join Domain Details text boxes if you selected Active Directory (Integrated Windows Authentication).

Option	Description
Server Location - Displayed when Active Directory over LDAP is selected	■ If you want to use DNS Service Location to locate Active Directory domains, leave the **This Directory supports DNS Service Location** check box selected. **Note** You cannot change the port assignment to 636 if you select this option. A `domain_krb.properties` file, auto-populated with a list of domain controllers, is created along with the directory. See About Domain Controller Selection. If the Active Directory requires STARTTLS encryption, select the **This Directory requires all connections to use STARTTLS** check box in the Certificates section and copy and paste the Active Directory Root CA certificate in the **SSL Certificate** field. ■ If the specified Active Directory does not use DNS Service Location lookup, deselect the check box beside **This Directory supports DNS Service Location** in the Server Location fields and enter the Active Directory server host name and port number in the appropriate text boxes. Select the **This Directory has a Global Catalog** check box if the associated Active Directory uses a global catalog. A global catalog contains a representation of all objects in every domain in a multi-domain Active Directory forest. To configure the directory as a global catalog, see the Multi-Domain Single Forest Active Directory Environment section in Active Directory Environments . If Active Directory requires access over SSL, select the **This Directory requires all connections to use SSL** check box under the Certificates heading and provide the Active Directory SSL certificate. When you select this option, port 636 is used automatically and cannot be changed. Ensure that the certificate is in PEM format and includes the BEGIN CERTIFICATE and END CERTIFICATE lines.
Join Domain Details - Displayed when Active Directory (Integrated Windows Authentication) is selected	Enter the appropriate credentials in the **Domain Name**, **Domain Admin User Name**, and **Domain Admin Password** text boxes. If the Active Directory requires STARTTLS encryption, select the **This Directory requires all connections to use STARTTLS** check box in the Certificates section and copy and paste the Active Directory Root CA certificate in the **SSL Certificate** field. Ensure that the certificate is in PEM format and includes the BEGIN CERTIFICATE and END CERTIFICATE lines.

Option	Description
	If the directory uses multiple domains, add the Root CA certificates for all domains, one at a time. **Note** If the Active Directory requires STARTTLS and you do not provide the certificate, you cannot create the directory.

7 In the Bind User Details section, enter the appropriate credentials to facilitate directory synchronization.

For Active Directory over LDAP:

Option	Description
Base DN	Enter the search base distinguished name. For example, `cn=users,dc=corp,dc=local`.
Bind DN	Enter the bind distinguished name. For example, `cn=fritz infra,cn=users,dc=corp,dc=local`

For Active Directory (Integrated Windows Authentication):

Option	Description
Bind User UPN	Enter the User Principal Name of the user who can authenticate with the domain. For example, UserName@example.com.
Bind DN Password	Enter the Bind User password.

8 Click **Test Connection** to test the connection to the configured directory.

This button does not appear if you selected Active Directory (Integrated Windows Authentication).

9 Click **Save & Next**.

The Select the Domains page appears with the list of domains.

10 Review and update the domains listed for the Active Directory connection.

- For Active Directory (Integrated Windows Authentication), select the domains that should be associated with this Active Directory connection.
- For Active Directory over LDAP, the available domain is listed with a checkmark.

Note If you add a trusting domain after the directory is created, the service does not automatically detect the newly trusting domain. To enable the service to detect the domain, the connector must leave and then rejoin the domain. When the connector rejoins the domain, the trusting domain appears in the list.

11 Click **Next**.

12 Verify that the Directories Management directory attribute names are mapped to the correct Active Directory attributes.

If the directory attribute names are not mapped correctly, select the correct Active Directory attribute from the drop-down menu.

13 Click **Next**.

14 Click ✚ to select the groups you want to sync from Active Directory to the directory.

When you add a group from Active Directory, if members of that group are not in the Users list, they are added. When you sync a group, any users that lack Domain Users as their primary group in Active Directory are not synced.

Note The Directories Management user authentication system imports data from Active Directory when adding groups and users, and the speed of the system is limited by Active Directory capabilities. As a result, import operations may require significant time depending on the number of groups and users being added. To minimize the potential for delays or problems, limit the number of groups and users to only those required for vRealize Automation operation.

If your system performance degrades or if errors occur, close any unneeded applications and ensure that your system has appropriate memory allocated to Active Directory. If problems persist, increase the Active Directory memory allocation as needed. For systems with a large number of users and groups, you may need to increase the Active Directory memory allocation to as much as 24 GB.

15 Click **Next**.

16 Click ✚ to add additional users.

The appropriate values are as follows:

- Single user: `CN=`*`username`*`,CN=Users,OU=Users,DC=myCorp,DC=com`
- Multiple users: `OU=Users,OU=myUnit,DC=myCorp,DC=com`

To exclude users, click ✚ to create a filter to exclude some types of users. You select the user attribute to filter by, the query rule, and the value.

17 Click **Next**.

18 Review the page to see how many users and groups are syncing to the directory.

If you want to make changes to users and groups, click the Edit links.

Note Ensure that you specify user DNs that are under the Base DN specified previously. If the user DN is outside of the Base DN, users from that DN are synced but will be unable to log in.

19 Click **Push to Workspace** to start the synchronization to the directory.

Results

The connection to the Active Directory is complete and the selected users and groups are added to the directory. You can now assign user and groups to the appropriate vRealize Automation roles by selecting **Administration > Users and Groups > Directory Users and Groups**. See Assign Roles to Directory Users or Groups for more information.

What to do next

If your vRealize Automation environment is configured for high availability, you must specifically configure Directories Management for high availability. See Configure Directories Management for High Availability.

- Set up authentication methods. After users and groups sync to the directory, if the connector is also used for authentication, you can set up additional authentication methods on the connector. If a third party is the authentication identity provider, configure that identity provider in the connector.
- Review the default access policy. The default access policy is configured to allow all appliances in all network ranges to access the Web browser, with a session time out set to eight hours or to access a client app with a session time out of 2160 hours (90 days). You can change the default access policy and when you add Web applications to the catalog, you can create new ones.
- Apply custom branding to the administration console, user portal pages and the sign-in screen.

Configure an OpenLDAP Directory Connection

You can configure an OpenLDAP Directory connection with Directories Management.

Though there are several different LDAP protocols, OpenLDAP is the only protocol that is tested and approved for use with vRealize Automation Directories Management.

To integrate your LDAP directory, you create a corresponding Directories Management directory and sync users and groups from your LDAP directory to the Directories Management directory. You can set up a regular sync schedule for subsequent updates.

You also select the LDAP attributes that you want to sync for users and map them to Directories Management attributes.

Your LDAP directory configuration may be based on default schemas or you may have created custom schemas. You may also have defined custom attributes. For Directories Management to be able to query your LDAP directory to obtain user or group objects, you need to provide the LDAP search filters and attribute names that are applicable to your LDAP directory.

Specifically, you need to provide the following information.

- LDAP search filters for obtaining groups, users, and the bind user

- LDAP attribute names for group membership, UUID, and distinguished name

Note Directories Management uses the default page size of 1500 for LDAP queries. If you configure an OpenLDAP directory connection, then you must enable the simple page results control extension for OpenLDAP to limit the number of results displayed. Failure to use this extension may cause user and group sync errors.

Prerequisites

- Review the configuration on the User Attributes page and add any other attributes that you want to sync. You will map the Directories Management attributes to your LDAP directory attributes when you create the directory. These attributes will be synced for the users in the directory.

  Note When you make changes to user attributes, consider the effect on other directories in the service. If you plan to add both Active Directory and LDAP directories, ensure that you do not mark any attributes as required except for **userName**. The settings on the User Attributes page apply to all directories in the service. If an attribute is marked required, users without that attribute are not synced to the Directories Management service.

- A Bind DN user account. Using a Bind DN user account with a non-expiring password is recommended.
- In your LDAP directory, the UUID of users and groups must be in plain text format.
- In your LDAP directory, a domain attribute must exist for all users and groups.

  You map this attribute to the Directories Management **domain** attribute when you create the Directories Management directory.

- User names must not contain spaces. If a user name contains a space, the user is synced but entitlements are not available to the user.
- If you use certificate authentication, users must have values for userPrincipalName and email address attributes.

Procedure

1. Select **Administration > Directories Management > Directories**.
2. Click **Add Directory** and select **Add LDAP Directory**.

3 Enter the required information in the Add LDAP Directory page.

Option	Description
Directory Name	Enter a name for the Directories Management directory.
Directory Sync and Authentication	a In the **Sync Connector** field, select the connector you want to use to sync users and groups from your LDAP directory to the Directories Management directory. A connector component is always available with the Directories Management service by default. This connector appears in the drop-down list. If you install multiple Directories Management appliances for high availability, the connector component of each appears in the list. You do not need a separate connector for an LDAP directory. A connector can support multiple directories, regardless of whether they are Active Directory or LDAP directories. b In the **Authentication** field, if you want to use this LDAP directory to authenticate users, select **Yes**. If you want to use a third-party identity provider to authenticate users, select **No**. After you add the directory connection to sync users and groups, go to the **Administration > Directories Management > Identity Providers page** to add the third-party identity provider for authentication. c For most configurations, leave the **Custom** default selected in the **Directory Search Attribute** text box. In the **Custom Directory Search Attribute** field, specify the LDAP directory attribute to be used for user and group names. This attribute uniquely identifies entities, such as users and groups, from the LDAP server. For example, **cn**. d If you want to use DNS Service Location lookup for Active Directory, make the following selections. ■ In the Server Location section, select the **This Directory supports DNS Service Location** check box. Directories Management finds and uses optimal domain controllers. If you don't want to use optimized domain controller selection, skip to step e. ■ If the Active Directory requires STARTTLS encryption, select the **This Directory requires all connections to use SSL** check box in the Certificates section, and copy and paste the Active Directory Root CA certificate into the SSL Certificate text box. Ensure that the certificate is in the PEM format and includes the "BEGIN CERTIFICATE" and "END CERTIFICATE" lines. **Note** If the Active Directory requires STARTTLS and you do not provide the certificate, you cannot create the directory. e If you do not want to use DNS Service Location lookup for Active Directory, make the following selections. ■ In the Server Location section, verify that the **This Directory supports DNS Service Location** check box is not selected, and enter the Active Directory server host name and port number. To configure the directory as a global catalog, see the Multi-Domain Single Forest Active Directory Environment section inActive Directory Environments .

Option	Description
	■ If the Active Directory requires access over SSL, select the **This Directory requires all connections to use SSL** check box in the Certificates section, and copy and paste the Active Directory Root CA certificate into the SSL Certificate field. Ensure that the certificate is in the PEM format and includes the "BEGIN CERTIFICATE" and "END CERTIFICATE" lines. **Note** If the Active Directory requires STARTTLS and you do not provide the certificate, you cannot create the directory.
Server Location	Enter the LDAP Directory server host and port number. For the server host, you can specify either the fully-qualified domain name or the IP address. For example, `myLDAPserver.example.com` or `100.00.00.0`. If you have a cluster of servers behind a load balancer, enter the load balancer information instead.
LDAP Configuration	Specify the LDAP search filters and attributes that Directories Management can use to query your LDAP directory. Default values are provided based on the core LDAP schema. **Filter Queries** ■ **Groups**: The search filter for obtaining group objects. For example: `(objectClass=group)` ■ **Bind user**: The search filter for obtaining the bind user object, that is, the user that can bind to the directory. For example: `(objectClass=person)` ■ **Users**: The search filter for obtaining users to sync. For example:`(&(objectClass=user)(objectCategory=person))` **Attributes** ■ **Membership**: The attribute that is used in your LDAP directory to define the members of a group. For example: `member` ■ **Object UUID**: The attribute that is used in your LDAP directory to define the UUID of a user or group. For example: `entryUUID` ■ **Distinguished Name**: The attribute that is used in your LDAP directory for the distinguished name of a user or group. For example: `entryDN`

Option	Description
Certificates	If your LDAP directory requires access over SSL, select the **This Directory requires all connections to use SSL** check box. Then copy and paste the LDAP directory server's root CA SSL certificate into the **SSL Certificate** text box. Ensure the certificate is in PEM format and include the "BEGIN CERTIFICATE" and "END CERTIFICATE" lines. If the directory has multiple domains, add the Root CA certificates for all of the domains, one after the other. Finally, ensure that the correct port number is specified in the **Server Port** field in the Server Location section of the page.
Bind User Details	**Base DN**: Enter the DN from which to start searches. For example, cn=users,dc=example,dc=com All applicable users must reside under the Base DN. If a particular user is not located under the Base DN, that user will be unable to log in even if he is a member of a group that is under the Base DN. **Bind DN**: Enter the DN to use to bind to the LDAP directory. You can also enter user names, but a DN is more appropriate for most deployments. **Note** Using a Bind DN user account with a non-expiring password is recommended. **Bind DN Password**: Enter the password for the Bind DN user.

4 To test the connection to the LDAP directory server, click **Test Connection**.

If the connection is not successful, check the information you entered and make the appropriate changes.

5 Click **Save & Next**.

6 Verify the correct domain is selected on the Select the Domains page, and then click **Next**.

7 In the Map Attributes page, verify that the Directories Management attributes are mapped to the correct LDAP attributes.

These attributes will be synced for users.

Important You must specify a mapping for the **domain** attribute.

You can add attributes to the list from the User Attributes page.

8 Click **Next**.

9 Click **+** to select the groups you want to sync from the LDAP directory to the Directories Management directory on Select the groups (users) you want to sync page.

If you have multiple groups with the same name in your LDAP directory, you must specify unique names for them in the groups page.

When you add a group from Active Directory, if members of that group are not in the Users list, they are added. When you sync a group, any users that lack Domain Users as their primary group in Active Directory are not synced.

The **Sync nested group members** option is enabled by default. When this option is enabled, all the users that belong directly to the group you select as well as all the users that belong to nested groups under it are synced. Note that the nested groups are not synced; only the users that belong to the nested groups are synced. In the Directories Management directory, these users will appear as members of the top-level group that you selected for sync. In effect, the hierarchy under a selected group is flattened and users from all levels appear in Directories Management as members of the selected group.

If this option is deactivated, when you specify a group to sync, all the users that belong directly to that group are synced. Users that belong to nested groups under it are not synced. Disabling this option is useful for large directory configurations where traversing a group tree is resource and time intensive. If you deactivate this option, ensure that you select all the groups whose users you want to sync.

Note The Directories Management user authentication system imports data from Active Directory when adding groups and users, and the speed of the system is limited by Active Directory capabilities. As a result, import operations may require a significant amount of time depending on the number of groups and users being added. To minimize the potential for delays or problems, limit the number of groups and users to only those required for vRealize Automation operation.

If your system performance degrades or if errors occur, close any unneeded applications and ensure that your system has appropriate memory allocated to Directories Management. If problems persist, increase the Directories Management memory allocation as needed. For systems with large numbers of users and groups, you may need to increase the Directories Management memory allocation to as much as 24 GB.

10 Click **Next**.

11 Click **+** to add additional users. For example, enter `CN=username,CN=Users,OU=myUnit,DC=myCorp,DC=com`.

   You can add organizational units as well as individual users here.

   You can create a filter to exclude some types of users. Select the user attribute to filter by, the query rule, and the value.

12 Click **Next**.

13 Review the page to see how many users and groups will sync to the directory and to view the default sync schedule.

   To make changes to users and groups, or to the sync frequency, click the **Edit** links.

14 Click **Sync Directory** to start the directory sync.

Results

The connection to the LDAP directory is established and users and groups are synced from the LDAP directory to the Directories Management directory.

You can now assign user and groups to the appropriate vRealize Automation roles by selecting **Administration > Users and Groups > Directory Users and Groups**. See Assign Roles to Directory Users or Groups for more information.

Limitations of LDAP Directory Integration

There are several important limitations related to LDAP Directory integration in Directories Management.

- You can only integrate a single-domain LDAP directory environment.

  To integrate multiple domains from an LDAP directory, you need to create additional Directories Management directories, one for each domain.

- The following authentication methods are not supported for Directories Management directories of type LDAP directory.
  - Kerberos authentication
  - RSA Adaptive Authentication
  - ADFS as a third-party identity provider
  - SecurID
  - Radius authentication with Vasco and SMS Passcode server
- You cannot join an LDAP domain.
- Integration with View or Citrix-published resources is not supported for Directories Management directories of type LDAP directory.
- User names must not contain spaces. If a user name contains a space, the user is synced but entitlements are not available to the user.
- If you plan to add both Active Directory and LDAP directories, ensure that you do not mark any attributes required in the User Attributes page, except for userName, which can be marked required. The settings in the User Attributes page apply to all directories in the service. If an attribute is marked required, users without that attribute are not synced to the Directories Management service.
- If you have multiple groups with the same name in your LDAP directory, you must specify unique names for them in the Directories Management service. You can specify the names when you select the groups to sync.
- The option to allow users to reset expired passwords is not available.
- The `domain_krb.properties` file is not supported.

Configure Directories Management for High Availability

You can use Directories Management to configure a high availability Active Directory connection in vRealize Automation.

Each vRealize Automation appliance includes a connector that supports user authentication, although only one connector is typically configured to perform directory synchronization. It does not matter which connector you choose as the sync connector. To support Directories Management high availability, you must manually configure a second connector that corresponds to your second vRealize Automation appliance, which connects to your Identity Provider and points to the same Active Directory. With this configuration, if one appliance fails, the other takes over management of user authentication.

In a high availability environment, all nodes must serve the same set of Active Directories, users, authentication methods, etc. The most direct method to accomplish this is to promote the Identity Provider to the cluster by setting the load balancer host as the Identity Provider host. With this configuration, all authentication requests are directed to the load balancer, which forwards the request to either connector as appropriate.

A connector is also used for user synchronization. But only one connector is configured to perform directory synchronization. Synced users are saved to appliance database, which is readable by all clustered nodes. If the connector that is responsible for directory synchronization fails, directory synchronization will stop working. To recover, the tenant admin needs to manually prompt another connector to perform directory synchronization using the vRealize Automation UI. See Enable Directory Sync on a Secondary Connector.

For more information about working with connectors, see Managing Connectors and Connector Clusters.

Prerequisites

- Configure your vRealize Automation deployment with at least two instance of the vRealize Automation appliance.
- Install vRealize Automation in Enterprise mode operating in a single domain with two instances of thevRealize Automation appliance.
- Install and configure an appropriate load balancer to work with your vRealize Automation deployment.
- Configure tenants and Directories Management using one of the connectors supplied with the installed instances of the vRealize Automation appliance. For information about tenant configuration, see Configuring Tenant Settings.

Procedure

1. Log in to the load balancer for your vRealize Automation deployment as a tenant administrator.

   The load balancer URL is `<load balancer address>/vcac/org/`*`tenant_name`*.

2. Select **Administration > Directories Management > Identity Providers**.
3. Click the Identity Provider that is currently in use for your system.

   The existing directory and connector that provide basic identity management for your system appears.

4 On the Identity Provider properties page, click the **Add a Connector** drop-down list, and select the connector that corresponds to your secondary vRealize Automation appliance.

5 Enter the appropriate password in the **Bind DN Password** text box that appears when you select the connector.

6 Click **Add Connector**.

7 The main connector appears in the **IdP Hostname**text box by default. Change the host name to point to the load balancer.

Enable Directory Sync on a Secondary Connector

If your primary connector fails, authentication is handled automatically by another connector instance. In the case of a failure, for directory sync, you must modify the directory settings in Directories Management to use the appropriate secondary connector instance. You can enable directory sync only on one connector at a time.

Procedure

1 Select **Administration > Directories Management > Directories**

2 Select the directory that was associated with the original connector instance.

> **Note** You can view this information on the **Directories > Connectors** page.

3 In the Directory Sync and Authentication section of the Directory page, select another connector instance in the **Sync Connector** drop-down list.

4 In the Bind User Details section, enter your Active Directory bind account password in the **Bind DN Password** text box.

5 Click **Save**.

Configure a Bi Directional Trust Relationship Between vRealize Automation and Active Directory

You can enhance system security of a basic vRealize Automation Active Directory connection by configuring a bi directional trust relationship between your identity provider and Active Directory Federated Services.

To configure a bi-directional trust relationship between vRealize Automation and Active Directory, you must create a custom identity provider and add Active Directory metadata to this provider. Also, you must modify the default policy used by your vRealize Automation deployment. Finally, you must configure Active Directory to recognize your identity provider.

Prerequisites

- Verify that you have configured tenants for your vRealize Automation deployment set up an appropriate Active Directory link to support basic Active Directory user ID and password authentication.
- Active Directory is installed and configured for use on your network.
- Obtain the appropriate Active Directory Federated Services (ADFS) metadata.

- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Obtain the Federation Metadata file.

   You can download this file from https://*servername.domain*/FederationMetadata/2007-06/FederationMetadata.xml

2 Search for the word logout, and edit the location of each instance to point to https://*servername.domain*/adfs/ls/logout.aspx

   For example, the following:

```
SingleLogoutService
                Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
                Location="https://servername.domain/adfs/ls/ "/>
```

   Should be changed to:

```
SingleLogoutService
                Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
                Location="https://servername.domain/adfs/ls/logout.aspx"/>
```

3 Create a new Identity Provider for you deployment.

   a Select **Administration > Directories Management > Identity Providers**.

   b Click **Add Identity Provider** and complete the fields as appropriate.

Option	Description
Identity Provider Name	Enter a name for the new identity provider
Identity Provider Metadata (URL or XML)	Paste the contents of your Active Directory Federated Services metadata file here.
Name ID Policy in SAML Request (Optional)	If appropriate, enter a name for the identity policy SAML request.
Users	Select the domains to which you want users to have access privileges.
Process IDP Metadata	Click to process the metadata file that you added.
Network	Select the network ranges to which you want users to have access.
Authentication Methods	Enter a name for the authentication method used by this identity provider.
SAML Context	Select the appropriate context for your system.
SAML Signing Certificate	Click the link beside the SAML Metadata heading to download the Directories Management metadata.

   c Save the Directories Management metadata file as `sp.xml`.

   d Click **Add**.

4. Add a rule to the default policy.

   a. Select **Administration > Directories Management > Policies**.

   b. Click the default policy name.

   c. Click the **+** icon in the **Policy Rules** heading to add a new rule.

      Use the options on the Add a Policy Rule page to create a rule that specifies the appropriate primary and secondary authentication methods to use for a specific network range and device.

      For example, if your network range is `My Machine`, and you need to access content from `All Device Types` then, for a typical deployment, you must authenticate by using the following method: `ADFS Username and Password`.

   d. Click **OK** to save your policy updates.

   e. On the Default Policy page, drag the new rule to the top of the table so that it takes precedence over existing rules.

5. Using the Active Directory Federated Services management console, or another appropriate tool, set up a relying party trust relationship with the vRealize Automation identity provider.

   To set up this trust, you must import the Directories Management metadata that you previously downloaded. See the Microsoft Active Directory documentation for more information about configuring Active Directory Federated Services for bi-directional trust relationships. As part of this process, you must do the following:

   - Set up a Relying Party Trust. When you set up this trust, you must import the VMware Identity Provider service provider metadata XML file that you copied and saved

   - Create a claim rule that transforms the attributes retrieved from LDAP in the Get Attributes rule into the desired SAML format. After you create the rule, edit the rule by adding the following text:

     ```
     c:[Type == "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress"]
     => issue(Type = "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier",
     Issuer = c.Issuer, OriginalIssuer = c.OriginalIssuer, Value = c.Value, ValueType =
     c.ValueType, Properties["http://schemas.xmlsoap.org/ws/2005/05/identity/claimproperties/
     format"] = "urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress", Properties["http://
     schemas.xmlsoap.org/ws/2005/05/identity/claimproperties/spnamequalifier"] =
     "vmwareidentity.domain.com");
     ```

Configure SAML Federation Between Directories Management and SSO2

You can establish SAML federation between vRealize Automation Directories Management and systems that use SSO2 to support single sign on.

Establish federation between Directories Management and SSO2 by creating a SAML connection between the two parties. Currently, the only supported end-to-end flow is where SSO2 acts as the Identity Provider (IdP) and Directories Management acts as the service provider (SP).

For SSO2 user authentication, the same account must exist in both Directories Management and SSO2. Minimally, the UserPrincipalName (UPN) of the user has to match on both ends. Other attributes can differ as they are required to identify the SAML subject.

For local users in SSO2, such as `admin@vsphere.local`, corresponding accounts must also exist in Directories Management, where at least the UPN of the user matches. Create these accounts manually or with a script using the Directories Management local user creation APIs.

Setting up SAML between SSO2 and Directories Management involves configuration on the Directories Management and SSO components.

Table 4-4. SAML Federation Component Configuration

Component	Configuration
Directories Management	Configure SSO2 as a third-party Identity Provider on Directories Management and update the default authentication policy. You can create an automated script to set up Directories Management.
SSO2 component	Configure Directories Management as a service provider by importing the Directories Management `sp.xml` file. This file enables you to configure SSO2 to use Directories Management as the Service Provider (SP).

Prerequisites

- Configure tenants for your vRealize Automation deployment. See Create Additional Tenants.
- Set up an appropriate Active Directory link to support basic Active Directory user ID and password authentication.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Download SSO2 Identity Provider metadata through the SSO2 user interface.

   a Log in to vCenter as an administrator at `https://<cloudvm–hostname>/` .

   b Click the **Log in to vSphere Web Client** link.

   c On the left navigation pane, select **Administration > Single Sign On > Configuration**.

   d Click **Download** adjacent to the Metadata for your SAML service provider heading.

   The `vsphere.local.xml` file should begin downloading.

   e Copy the contents of the `vsphere.local.xml` file.

2 On the vRealize Automation Directories Management Identity Providers page, create a new Identity Provider.

   a Log in to vRealize Automation as a **tenant administrator**.

   b Select **Administration > Directories Management > Identity Providers**.

c Click **Add Identity Provider** and provide the configuration information.

Option	Action
Identity Provider Name	Enter a name for the new Identity Provider.
Identity Provider Metadata (URI or XML) text box	Paste the contents of your SSO2 `idp.xml` metadata file in the text box and click **Process IDP Metadata**.
Name ID Policy in SAML Request (Optional)	Enter `http://schemas.xmlsoap.org/claims/UPN`.
Users	Select the domains to which you want users to have access privileges.
Network	Select the network ranges from which you want users to have access privileges. If you want to authenticate users from an IP addresses, select **All Ranges**.
Authentication Methods	Enter a name for the authentication method. Then, use the **SAML Context** drop down menu to the right to map the authentication method to `urn:oasis:names:tc:SAML:2.0:ac:classes:Password`.
SAML Signing Certificate	Click the link beside the SAML Metadata heading to download the Directories Management metadata.

d Save the Directories Management metadata file as `sp.xml`.

e Click **Add**.

3 Update the relevant authentication policy using the Directories Management Policies page to redirect authentication to the third party SSO2 identity provider.

a Select **Administration > Directories Management > Policies**.

b Click the default policy name.

c Click the authentication method under the **Policy Rules** heading to edit the existing authentication rule.

d On the Edit a Policy Rule page, change the authentication method from password to the appropriate method.

In this case, the method should be SSO2.

e Click **Save** to save your policy updates.

4 On the left navigation pane, select **Administration > Single Sign On > Configuration**, and click **Update** to upload the `sp.xml` file to vSphere.

Add Users or Groups to an Active Directory Connection

You can add users or groups to an existing Active Directory connection.

The Directories Management user authentication system imports data from Active Directory when adding groups and users. The speed of the data transport is limited by Active Directory capabilities. As a result, actions can take a long time depending on the number of groups and users that are added. To minimize problems, limit the groups and users to only the groups and users required for a vRealize Automation action. If problems occur, close unneeded applications and verify that your deployment has appropriate memory allocated to Active Directory. If problems continue, increase the Active Directory memory allocation. For deployments with large numbers of users and groups, you might need to increase the Active Directory memory allocation to as much as 24 GB.

When you sync a vRealize Automation deployment with a many users and groups, there might be a delay before the SyncLog details are available. The time stamp on the log file can differ from the completed time displayed on the console.

If members of a group are not in the Users list, when you add the group from Active Directory, the members are added to the list. When you sync a group, any users that do not have Domain Users as their primary group in Active Directory are not synced.

Note You cannot cancel a synchronize action after you start the action.

Prerequisites

- Connector installed and the activation code activated. Select the required default attributes and add additional attributes on the User Attributes page.
- List of the Active Directory groups and users to sync from Active Directory.
- For Active Directory over LDAP, information required includes the Base DN, Bind DN, and Bind DN password.
- For Active Directory Integrated Windows Authentication, the information required includes the domain's Bind user UPN address and password.
- If Active Directory is accessed over SSL, a copy of the SSL certificate is required.
- If you have a multi-forest Active Directory integrated with Windows Authentication and the Domain Local group contains members from different forests, do the following. Add the Bind user to the Administrators group of the Domain Local group. If the Bind user is not added, these members are missing from the Domain Local group.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1. Select **Administration > Directories Management > Directories**.
2. Click the desired directory name.
3. Click **Sync Settings** to open a dialog box with synchronization options.
4. Click the appropriate icon depending on whether you want to change the user or group configuration.

   To edit the group configuration:

   - To add groups, click the **+** icon to add a line for group DN definitions and enter the appropriate group DN.
   - If you want to delete a group DN definition, click the **x** icon for the desired group DN.

To edit the user configuration:

- To add users, click the **+** icon to add a line for a user DN definition and enter the appropriate user DN.

If you want to delete a user DN definition, click the **x** icon for the desired user DN.

5 Click **Save** to save your changes without synchronizing your updates immediately. Click **Save & Sync** to save your changes and synchronize your updates immediately.

Select Attributes to Sync with Directory

When you set up the Directories Management directory to sync with Active Directory, you specify the user attributes that sync to the directory. Before you set up the directory, you can specify on the User Attributes page which default attributes are required and, if you want, add additional attributes that you want to map to Active Directory attributes.

When you configure the User Attributes page before the directory is created, you can change default attributes from required to not required, mark attributes as required, and add custom attributes.

For a list of the default mapped attributes, see Managing User Attributes that Sync from Active Directory.

After the directory is created, you can change a required attribute to not be required, and you can delete custom attributes. You cannot change an attribute to be a required attribute.

When you add other attributes to sync to the directory, after the directory is created, go to the directory's Mapped Attributes page to map these attributes to Active Directory Attributes.

Procedure

1 Log in to vRealize Automation as a system or tenant administrator.

2 Click the Administration tab.

3 Select **Directories Management > User Attributes**

4 In the Default Attributes section, review the required attribute list and make appropriate changes to reflect what attributes should be required.

5 In the Attributes section, add the Directories Management directory attribute name to the list.

6 Click **Save**.

The default attribute status is updated and attributes you added are added on the directory's Mapped Attributes list.

7 After the directory is created, go to the Identity Stores page and select the directory.

8 Click **Sync Settings > Mapped Attributes**.

9 In the drop-down menu for the attributes that you added, select the Active Directory attribute to map to.

10 Click **Save.**

Results

The directory is updated the next time the directory syncs to the Active Directory.

Add Memory to Directories Management

You may need to allocate additional memory to Directories Management if you have Active Directory connections that contain a large number of users or groups.

By default, 4 GB of memory is allocated to the Directories Management service. This is sufficient for many small to medium sized deployments. If you have an Active Directory connection that uses a large number of users or groups, you may need to increase this memory allocation. Increased memory allocation is appropriate for systems with more than 100,000 users , each in 30 groups and 750 groups overall. For these system, VMware recommends increasing the Directories Management memory allocation to 6 GB.

Directories Management memory is calculated based on the total memory allocated to the vRealize Automation appliance The following table shows memory allocations for relevant components.

Table 4-5. vRealize Automation Appliance Memory Allocation

Virtual Appliance memory	vRA service memory	vIDM service memory
18 GB	3.3 GB	4 GB
24 GB	4.9 GB	6 GB
30 GB	7.4 GB	9.1 GB

Note These allocations assume that all default services are enabled and running on the virtual appliance. They may change if some services are stopped.

Prerequisites

- An appropriate Active Directory connection is configured and functioning on your vRealize Automation deployment.

Procedure

1. Stop each machine on which a vRealize Automation appliance is running.
2. Increase the virtual appliance memory allocation on each machine.

   If you are using the default memory allocation of 18 GB, VMware recommends increasing the memory allocation to 24 GB.
3. Restart the vRealize Automation appliance machines.

Configure Just-in-Time User Provisioning

You can configure Just-in-Time (JIT) provisioning to support adding users without syncing from your Active Directory.

To support Just-in-Time provisioning, you must add a third party identity provider and then configure a connection to it within your vRealize Automation deployment to integrate Directories Management with other SSO providers via a SAML protocol. In addition, you must create a new directory with the appropriate name, such as JIT Directory.

When you enable Just-in-Time provisioning, you can add Just-in-Time users to a designated custom group. To support this functionality, create a custom group with the appropriate members. See Add Just-in-Time Users with Custom Groups and Rules.

Note As a best practice, do not configure Just-in-Time provisioning on the default vsphere.local tenant.

Prerequisites

Configure an appropriate third party identity provider for use with JIT provisioning.

Procedure

1 Create an identity provider for Just-in-Time provisioning.

   a Select **Administration > Directories management > Identity Providers**

   b Click **Add Identity Provider** and edit the identity provider instance settings as appropriate.

      - For just in time provisioning, create a third party identity provider.
      - In the Create Just-in-Time Directory section, enter names for the directory and one or more domains.
      - You must select a network for the third party identity provider configuration.
      - If you are using an external VMware Identity Manager as your third party identity provider, and you are using `userPrincipleName` to authenticate users, you must change the Name ID mapping configuration for `userPrincipleName` from the default of `x509SubjectName` to `unspecified`.

      See Configure a Third Party Identity Provider Connection for more information about creating identity providers.

2 Configure SAML on the Just-in-Time identity provider.

   a Copy IdP metadata from your identity provider.

   b In vRealize Automation, select your identity provider and paste the IdP metadata into the **Identity Provider Metadata (URL or XML)** text box.

   c Click **Save**.

   d In the **Name ID policy in SAML Request (Optional)** drop-down menu, select the appropriate format.

      For example, if you are using the emal address as the unique user identifier, you would select `urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress`.

e Select the appropriate directory under the Users heading.

f Select the networks for use by this identity provider under the Network heading.

g Specify an appropriate name in the **Authentication Methods** text box.

h In the **SAML Context** drop down, select `urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport`

i Right-click the **Service Provider (SP) Metadata** link, and open it in a separate browser tab.

j Use this metadata to configure the SAML connection on your identity provider.

If you are using VMware Identity Manager see the VMware Identity Manager documentation for complete instructions on configuring SAML.

3 Click **Add**.

The new directory is created using the Directory Name provided.

4 Configure the vRealize Automation Access Policy.

a Select **Administration > Policies**.

b Click the green + icon at the top right of the policy rules table.

c Set the policy rule to apply to applicable ranges and device types.

d Select the authentication method that you created when configuring the third party identity provider for JIT provisioning for the authentication method.

Managing User Attributes that Sync from Active Directory

The Directories Management User Attributes page lists the user attributes that sync to your Active Directory connection.

Changes that you make and save in the User Attributes page are added to the Mapped Attributes page in the Directories Management directory. The attributes changes are updated to the directory with the next sync to Active Directory.

The User Attributes page lists the default directory attributes that you can map to Active Directory attributes. You select the attributes that are required, and you can add other Active Directory attributes to sync to the directory.

Table 4-6. Default Active Directory Attributes to Sync to Directory

Directory Attribute Name	Default Mapping to Active Directory Attribute
userPrincipalName	userPrincipalName
distinguishedName	distinguishedName
employeeId	employeeID
domain	canonicalName. Adds the fully qualified domain name of the object.

Table 4-6. Default Active Directory Attributes to Sync to Directory (continued)

Directory Attribute Name	Default Mapping to Active Directory Attribute
disabled (external user disabled)	userAccountControl. Flagged with UF_Account_Disable. When an account is disabled, users cannot log in to access their applications and resources. The resources that users were entitled to are not removed from the account so that when the flag is removed from the account users can log in and access their entitled resources.
phone	telephoneNumber
lastName	sn
firstName	givenName
email	mail
userName	sAMAccountName

The User Attributes page lists the default directory attributes that you can map to Active Directory attributes. You select the attributes that are required, and you can add other Active Directory attributes to sync to the directory.

Table 4-7. Default Active Directory Attributes to Sync to Directory

Directory Attribute Name	Default Mapping to Active Directory Attribute
userPrincipalName	userPrincipalName
distinguishedName	distinguishedName
employeeId	employeeID
domain	canonicalName. Adds the fully qualified domain name of the object.
disabled (external user disabled)	userAccountControl. Flagged with UF_Account_Disable. When an account is disabled, users cannot log in to access their applications and resources. The resources that users were entitled to are not removed from the account so that when the flag is removed from the account users can log in and access their entitled resources.
phone	telephoneNumber
lastName	sn
firstName	givenName
email	mail
userName	sAMAccountName

Managing Connectors and Connector Clusters

The Connectors page lists deployed connectors for your enterprise network. A connector syncs user and group data between Active Directory and the Directories Management service, and when it is used as the identity provider, authenticates users to the service.

In vRealize Automation, each vRealize Automation appliance contains its own connector, and these connectors are suitable for most deployments.

When you associate a directory with a connector instance, the connector creates a partition for the associated directory called a worker. A connector instance can have multiple associated workers. Each worker acts as an identity provider. The connector syncs user and group data between Active Directory and the service through one or more workers. You define and configure authentication methods on a per worker basis.

You can manage various aspects of an Active Directory link from the Connectors page. This page contains a table and several buttons that enable you to complete various management tasks.

- In the Worker column, select a worker to view the connector details and navigate to the Auth Adapters page to see the status of the available authentication methods. For information about authentication, see Integrating Alternative User Authentication Products with Directories Management.
- In the Identity Provider column, select the IdP to view, edit or deactivate. See Configure a Third Party Identity Provider Connection.
- In the Associated Directory column, access the directory associated with this worker.
- Click **Join Domain** to join the connector to a specific Active Directory domain. For example when you configure Kerberos authentication, you must join the Active Directory domain either containing users or having trust relationship with the domains containing users.
- When you configure a directory with an Integrated Windows Authentication Active Directory, the connector joins the domain according to the configuration details.

Connectors in a Clustered Environment

In a distributed, vRealize Automation deployment, all available connectors perform any required user authorization, while a single designated connector handles all configuration synchronization. Typically, synchronization would include additions, deletions, or changes to the user configuration, and synchronization occurs automatically as long as all connectors are available. There are some specific situations in which automatic synchronization may not occur.

For changes related to directory configuration, such as base dn, vRealize Automation attempts to automatically push updates to all connectors in a cluster. If a connector is inoperable or unreachable for some reason, that connector will not receive the update, even when it resumes online operation. To implement configuration changes to connectors that may not have received them automatically, system administrators must manually save the changes to all applicable connectors.

For directory sync profile related changes, vRealize Automation attempts to automatically push updates to all connectors as well. If the sync connector is operational, the update is saved and pushed to all available authorization connectors. If one or more connectors is unreachable, the system admin receives a warning indicating that not all connectors were updated. If the sync connector is inoperable, the update fails and an error occurs. If the system admin changes the connector designated as the sync connector, the new sync connector receives the latest available profile information, and this information is pushed to all applicable, and available, connectors.

Join a Connector Machine to a Domain

In some cases, you may need to join a machine containing a Directories Management connector to a domain.

For Active Directory over LDAP directories, you can join a domain after creating the directory. For Active Directory (Integrated Windows Authentication) directories, the connector is joined to the domain automatically when you create the directory. In both cases, you must supply the appropriate credentials.

To join a domain, you need Active Directory credentials that have the privilege to "join computer to AD domain". This is configured in Active Directory with the following rights:

- Create Computer Objects
- Delete Computer Objects

When you join a domain, a computer object is created in the default location in Active Directory.

If you do not have the rights to join a domain, or if your company policy requires a custom location for the computer object, you must ask your administrator to create the object and then join the connector machine to the domain.

Procedure

1. Ask your Active Directory administrator to create the computer object in Active Directory in a location determined by your company policy. You must provide the host name of the connector. Ensure that you provide the fully-qualified domain name, for example `server.example.com`.

   You can find the host name in the Host Name column on the Connectors page in the administrative console. Select **Administration > Directories Management > Connectors**.

2. After the computer object is created, click **Join Domain** on the Connectors page to join the domain using any domain user account available in Directories Management.

About Domain Controller Selection

Directories Management maintains a dynamic list of domain controllers that requires no user configuration.

Directories Management periodically refreshes, rediscovers and reorders domain controllers based on LDAP Ping and stores them in a domain_krb.properties file and a custom krb5.conf file. The best domain controller is listed first and hence used for all purposes like authentication and sync operations. If this domain controller fails to respond within 10 ms, then list of domain controllers is refreshed again. This allows Directories Management to consistently use optimal domain controllers, even during domain controller failures.

Managing Access Policies

To provide secure access to the users' apps portal and to launch Web and desktop applications, you configure access policies with rules that specify criteria that must be met to sign in to their apps portal and to use their resources.

Policy rules map the requesting IP address to network ranges and designate the type of devices that users can use to sign in. The rule defines the authentication methods and the number of hours the authentication is valid.

The Directories Management service includes a default policy that controls access to the service as a whole. This policy is set up to allow access to all network ranges, from all device types, with a session time out at eight hours, and the authentication method is password authentication. You can edit the default policy.

Note The policies do not control the length of time that an application session lasts. They control the amount of time that users have to launch an application.

Configuring Access Policy Settings

A policy contains one or more access rules. Each rule consists of settings that you can configure to manage user access to their application portals as a whole or to specified Web applications.

Network Range

For each rule, you determine the user base by specifying a network range. A network range consists of one or more IP ranges. You create network ranges from the Identity & Access Management tab, Setup > Network Ranges page prior to configuring access policy sets.

Device Type

Select the type of device that the rule manages. The client types are Web Browser, Identity Manager Client App, iOS, Android, and All device types.

Add Groups

You can apply different policies for authentication based on group membership of your users. To assign groups of users to log in through a specific authentication flow, you can add groups to the access policy rule. You can sync groups from your enterprise directory or local groups that you created in the admin console. Group names must be unique within a domain.

To use groups in access policy rules, you configure a new policy from the Directories Management > Policies page and select the desired groups for the policy. The policy must be mapped in the User Attributes page and then synced to the directory.

When groups are used in an access policy rule, the user login experience for the user changes. Instead of asking users to select their domain and then enter their credentials, a page displays prompting them to enter their unique identifier. Directories Management finds the user in the internal database, based on the unique identifier and displays the authentication page configured in that rule.

When a group is not selected, the access policy rule applies to all users. When you configure access policy rules that include rules based on groups and a rule for all users, make sure that the rule designated for all users is the last rule listed in the Policy Rules section of the policy.

See the VMware Identity Manager documentation on Login Experience Using Unique Identifier for more information about how rules are applied to users.

Authentication Methods

Set the priority of the authentication methods for the policy rule. The authentication methods are applied in the order they are listed. The first identity provider instances that meets the authentication method and network range configuration in the policy is selected, and the user authentication request is forwarded to the identity provider instance for authentication. If authentication fails, the next authentication method in the list is selected. If Certificate authentication is used, this method must be the first authentication method in the list.

You can configure access policy rules to require users to pass credentials through two authentication methods before they can sign in. If one or both authentication method fails and fallback methods are also configured, users are prompted to enter their credentials for the next authentication methods that are configured. The following two scenarios describe how authentication chaining can work.

- In the first scenario, the access policy rule is configured to require users to authenticate with their password and with their Kerberos credential. Fallback authentication is set up to require the password and the RADIUS credential for authentication. A user enters the password correctly, but fails to enter the correct Kerberos authentication credential. Since the user entered the correct password, the fallback authentication request is only for the RADIUS credential. The user does not need to re-enter the password.
- In the second scenario, the access policy rule is configured to require users to authenticate with their password and their Kerberos credential. Fallback authentication is set up to require RSA SecurID and a RADIUS for authentication. A user enters the password correctly but fails to enter the correct Kerberos authentication credential. The fallback authentication request is for both the RSA SecurID credential and the RADIUS credential for authentication.

Authentication Session Length

For each rule, you set the length that this authentication is valid. The value determines the maximum amount of time users have since their last authentication event to access their portal or to launch a specific Web application. For example, a value of *4* in a Web application rule gives users four hours to launch the web application unless they initiate another authentication event that extends the time.

Custom Access Denied Error Message

When users attempt to sign in and fail because of invalid credentials, incorrect configuration, or system error, an access denied message is displayed. The default message is

```
Access denied as no valid authentication methods were found.
```

You can create a custom error message for each access policy rule that overrides the default message. The custom message can include text and a link for a call to action message. For example, in a policy rules for mobile devices that you want to manage, if a user tries to sign in from an unenrolled device, the follow custom error message could appear:

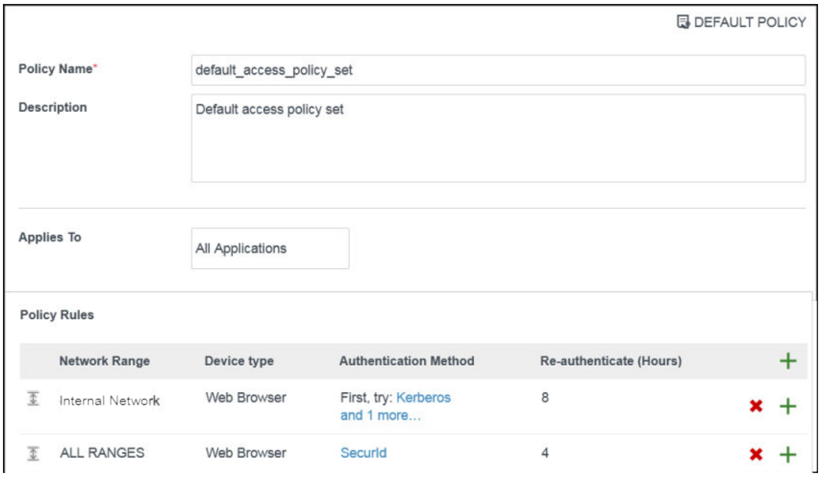
```
Please enroll your device to access corporate resources by clicking the link at the end of this
message.  If your device is already enrolled, contact support for help.
```

Example Default Policy

The following policy serves as an example of how you can configure the default policy to control access to the apps portal. See Manage the User Access Policy.

The policy rules are evaluated in the order listed. You can change the order of the policy by dragging and dropping the rule in the Policy Rules section.

In the following use case, this policy example applies to all applications.



DEFAULT POLICY

Policy Name*: default_access_policy_set

Description: Default access policy set

Applies To: All Applications

Policy Rules

Network Range	Device type	Authentication Method	Re-authenticate (Hours)
Internal Network	Web Browser	First, try: Kerberos and 1 more...	8
ALL RANGES	Web Browser	SecurId	4

1 ■ For the internal network (Internal Network Range), two authentication methods are configured for the rule, Kerberos and password authentication as the fallback method. To access the apps portal from an internal network, the service attempts to authenticate users with Kerberos authentication first, as it is the first authentication method listed in the rule. If that fails, users are prompted to enter their Active Directory password. Users log in using a browser and now have access to their user portals for an eight-hour session.

- For access from the external network (All Ranges), only one authentication method is configured, RSA SecurID. To access the apps portal from an external network, users are required to log in with SecurID. Users log in using a browser and now have access to their apps portals for a four-hour session.

2 When a user attempts to access a resource, except for Web applications covered by a Web-application-specific policy, the default portal access policy applies.

   For example, the re-authentication time for such resources matches the re-authentication time of the default access policy rule. If the time for a user who logs in to the apps portal is eight hours according to the default access policy rule, when the user attempts to launch a resource during the session, the application launches without requiring the user to re-authenticate.

Configure a Group-based Access Policy

You can configure a group based access policy to control login privileges based on group assignments.

Directories Management contains default access policies that support all groups and all network ranges. You can modify these policies to be more restrictive, or you can create new policies to support different login policies.

Procedure

1 Add groups to the desired policy.

   a Select **Administration > Directories Management > Policies**.

   b You can open the default access policy or create a new one.

   c Edit the policy rule that is configured with a device type of Web Browser.

      To edit a policy, click on its Authentication method. By default, there are two policy rules that apply to all IP addresses and all users.

      The Edit Policy Rule page opens for the selected policy. You can edit various parameters such as the Network Range, Device Type, Authentication methods, and other rule parameters for the policy.

   d Click **Edit Groups** on the Edit Policy Rule page to view all groups available for use with the policy.

      This page shows all of the groups associated with the tenant.

   e Select the groups that you want to associate with the policy.

   f Click **OK**.

      The selected groups appear on the Edit Policy Rule page.

g Click **OK** on the Edit Policy Rule page to save the changes to the policy rule.

  The Policies page appears showing the number of groups selected for the policy.

h Click **Save** on the Policies page.

2 Configure a Network Range for the group policy.

a Select **Administration > Directories Management > Network Ranges**

  By default, there is a predefined setting of `All Ranges` which covers all IP addresses for all network ranges. You can create a new network range or edit one of the existing ones.

b Click **Add Network Range**.

  The Edit Network Range page opens.

c Enter a **Name** for the new network range and add a **Description** if needed.

Results

When users log in to vRealize Automation, they must select the domain and then enter a valid username and password. If a group is specified in the applicable policy, valid users must still enter a username and password.

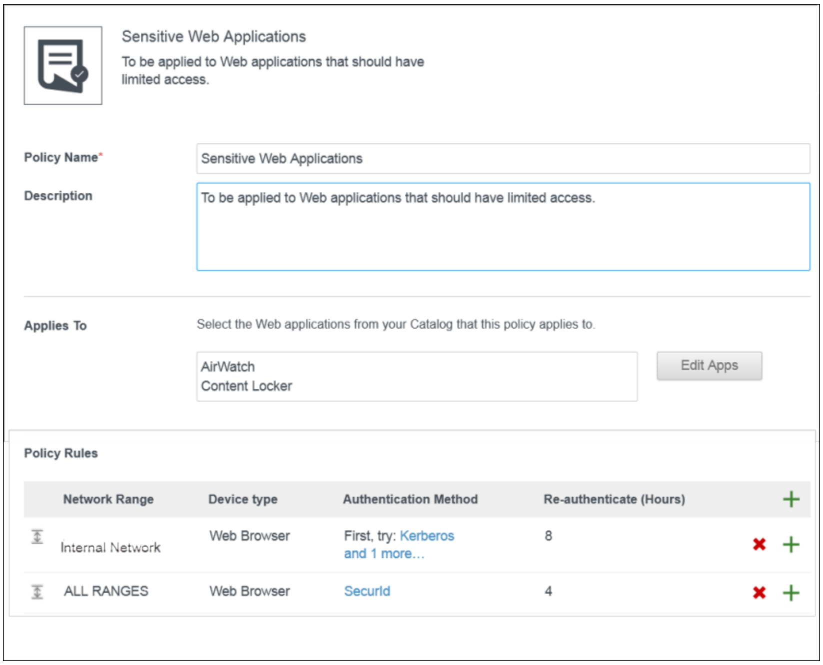
Managing Web and Desktop Application-Specific Policies

When you add Web and desktop applications to the catalog, you can create application-specific access policies. For example, you can create a policy with rules for a Web application that specifies which IP addresses have access to the application, using which authentication methods, and for how long until reauthentication is required.

The following Web-application-specific policy provides an example of a policy you can create to control access to specified Web applications.

Example 1 Strict Web-Application-Specific Policy

In this example, a new policy is created and applied to a sensitive Web application.



Sensitive Web Applications

To be applied to Web applications that should have limited access.

Policy Name*	Sensitive Web Applications
Description	To be applied to Web applications that should have limited access.
Applies To	Select the Web applications from your Catalog that this policy applies to. AirWatch Content Locker

Policy Rules

Network Range	Device type	Authentication Method	Re-authenticate (Hours)
Internal Network	Web Browser	First, try: Kerberos and 1 more...	8
ALL RANGES	Web Browser	SecurId	4

1 To access the service from outside the enterprise network, the user is required to log in with RSA SecurID. The user signs in using a browser and now has access to the apps portal for a four-hour session as provided by the default access rule.

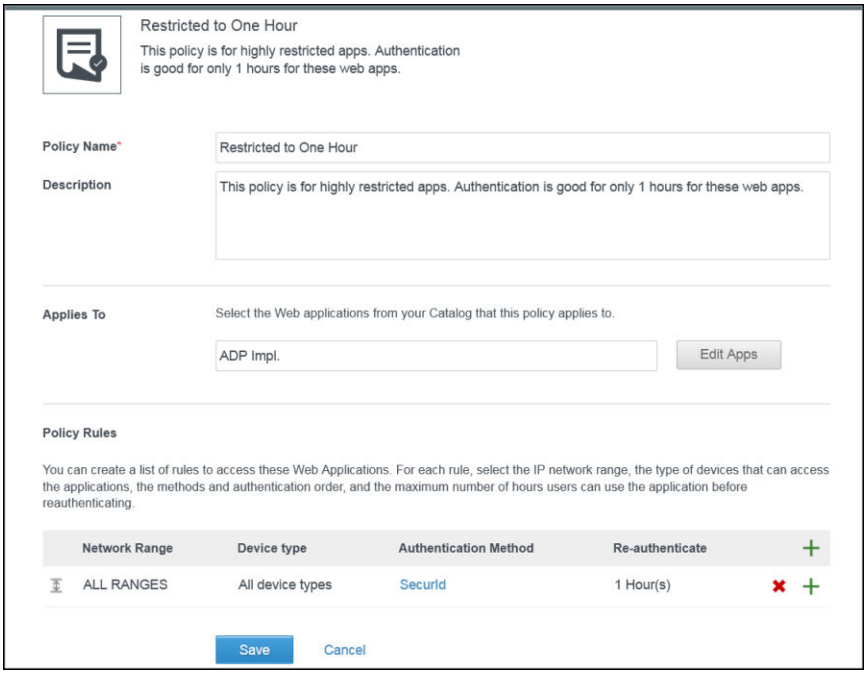
2 After four hours, the user tries to start a Web application with the Sensitive Web Applications policy set applied.

3 The service checks the rules in the policy and applies the policy with the ALL RANGES network range because the user request is coming from a Web browser and from the ALL RANGES network range.

The user signed in with the RSA SecurID authentication method, but the session just expired. The user is redirected for reauthentication. The reauthentication provides the user with another four-hour session and the ability to start the application. For the next four hours, the user can continue to run the application without having to reauthenticate.

Example 2 Stricter Web-Application-Specific Policy

For a stricter rule to apply to extra sensitive Web applications, you could require reauthentication with SecurId on any device after one hour. The following is an example of how this type of a policy access rule is implemented.



Restricted to One Hour

This policy is for highly restricted apps. Authentication is good for only 1 hours for these web apps.

Policy Name*	Restricted to One Hour
Description	This policy is for highly restricted apps. Authentication is good for only 1 hours for these web apps.

Applies To

Select the Web applications from your Catalog that this policy applies to.

ADP Impl.

Policy Rules

You can create a list of rules to access these Web Applications. For each rule, select the IP network range, the type of devices that can access the applications, the methods and authentication order, and the maximum number of hours users can use the application before reauthenticating.

Network Range	Device type	Authentication Method	Re-authenticate
ALL RANGES	All device types	SecurId	1 Hour(s)

1 User logs in from inside the enterprise network using the Kerberos authentication method.

Now, the user can access the apps portal for eight hours, as set up in Example 1.

2 The user immediately tries to start a Web application with the Example 2 policy rule applied, which requires RSA SecurID authentication.

3 The user is redirected to RSA SecurID authentication sign-in page.

4 After the user successfully signs in, the service launches the application and saves the authentication event.

The user can continue to run this application for up to one hour but is asked to reauthenticate after an hour, as dictated by the policy rule.

Manage the User Access Policy

vRealize Automation is supplied with a default user access policy that you can use as is or edit as needed to manage tenant access to applications.

vRealize Automation is supplied with a default user access policy, and you cannot add new policies. You can edit the existing policy to add rules.

Prerequisites

- Select or configure the appropriate identity providers for your deployment. See Configure a Third Party Identity Provider Connection.
- Configure the appropriate network ranges for your deployment. See Add or Edit a Network Range.

- Configure the appropriate authentication methods for your deployment. See Integrating Alternative User Authentication Products with Directories Management.
- If you plan to edit the default policy (to control user access to the service as a whole), configure it before creating Web-application-specific policy.
- Add Web applications to the Catalog. The Web applications must be listed in the Catalog page before you can add a policy.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1. Select **Administration > Directories Management > Policies**.
2. Click **Edit Policy** to add a new policy.
3. Add a policy name and description in the respective text boxes.
4. In the Applies To section, click **Select** and in the page that appears, select the Web applications that are associated with this policy.
5. In the Policy Rules section, click **+** to add a rule.

   The Add a Policy Rule page appears.

   a. Select the network range to apply to this rule.
   b. Select the type of device that can access the web applications for this rule.
   c. Select the authentication methods to use in the order the method should be applied.
   d. Specify the number of hours a Web application session open.
   e. Click **Save**.
6. Configure additional rules as appropriate.
7. Click **Save**.

Configuring Additional Identity Provider Connections

You can configure additional identity provider connections as needed to support different identity management scenarios, including additional built-in identity providers and third-party identity providers.

You can create three types of identity provider connections using Directories Management.

- Create Third-Party IDP - Use this item to create a connection to an external third-party identity provider. Ensure that you have following before adding a third-party identity provider instance.
  - Verify that the third-party instances are SAML 2.0 compliant and that the service can reach the third-party instance.

  - Obtain the appropriate third-party metadata information to add when you configure the identity provider in the administration console. The metadata information you obtain from the third-party instance is either the URL to the metadata or the actual metadata.

- Create Workspace IDP - When you enable a connector to authenticate users during Directories Management configuration, a Workspace IDP is created as the identity provider and password authentication is enabled. You can configure additional workspace identity providers behind different load balancers.
- Create Built-in IDP - Built in Identity Providers use the internal Directories Management mechanisms to support authentication. You can configure built-in identity providers to use authentication methods that do not require the use of an on premises connector. When you configure the built-in provider, you associate the authentication methods to use with the provider.
- Configure a Third Party Identity Provider Connection

  vRealize Automation is supplied with a default identity provider connection instance. Users may want to create additional identity provider connections to support just-in-time user provisioning or other custom configurations.
- Configure Additional Workspace Identity Providers

  When you configure a Directories Management connector to authenticate users, a Workspace IDP is created and password authentication is enabled.
- Configure a Built-in Identity Provider Connection

  You can configure multiple built-in identity providers and associate authentication methods with them.

Configure a Third Party Identity Provider Connection

vRealize Automation is supplied with a default identity provider connection instance. Users may want to create additional identity provider connections to support just-in-time user provisioning or other custom configurations.

vRealize Automation is supplied with an default identity provider. In most cases, the default provider is sufficient for customer needs. If you use an existing enterprise identity management solution, you can set up a custom identity provider to redirect users to your existing identity solution.

When using a custom identity provider, Directories Management uses SAML metadata from that provider to establish a trust relationship with the provider. After this relationship is established, Directories Management maps the users from the SAML assertion to the list of internal vRealize Automation users based the subject name ID.

Prerequisites

- Configure the network ranges that you want to direct to this identity provider instance for authentication. See Add or Edit a Network Range.

- Access to the third-party metadata document. This can be either the URL to the metadata or the actual metadata.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Directories Management > Identity Providers**.

This page displays all configured Identity Providers.

2 Click **Add Identity Provider**.

A menu appears with Identity Provider options.

3 Select **Create Third Party IDP**.

4 Enter the appropriate information to configure the identity provider.

Option	Description
Identity Provider Name	Enter a name for this identity provider instance.
SAML Metadata	Add the third party IdPs XML-based metadata document to establish trust with the identity provider. 1 Enter the SAML metadata URL or the xml content into the text box. 2 Click **Process IdP Metadata**. The NameID formats supported by the IdP are extracted from the metadata and added to the Name ID Format table. 3 In the Name ID value column, select the user attribute in the service to map to the ID formats displayed. You can add custom third-party name ID formats and map them to the user attribute values in the service. 4 (Optional) Select the NameIDPolicy response identifier string format.
Users	Select the Directories Management directories of the users that can authenticate using this identity provider.
Just-in-Time User Provisioning	Select the appropriate options to support just-in-time provisioning using an appropriate third party identity provider. Enter the **Directory Name** to use for just-in-time provisioning. Enter one or more **Domains** that exist within the external identity provider that you will use for just-in-time provisioning.
Network	The existing network ranges configured in the service are listed. Select the network ranges for the users, based on their IP addresses, that you want to direct to this identity provider instance for authentication.
Authentication Methods	Add the authentication methods supported by the third-party identity provider. Select the SAML authentication context class that supports the authentication method.
SAML Signing Certificate	Click **Service Provider (SP) Metadata** to see URL to Directories Management SAML service provider metadata URL . Copy and save the URL. This URL is configured when you edit the SAML assertion in the third-party identity provider to map Directories Management users.
Hostname	If the **Hostname** field displays, enter the hostname where the identity provider is redirected to for authentication. If you are using a non-standard port other than 443, you can set this as Hostname:Port. For example, myco.example.com:8443.

5 Click **Add**.

What to do next

- Copy and save the Directories Management service provider metadata that is required to configure the third-party identity provider instance. This metadata is available either in the SAML Signing Certificate section of the Identity Provider page.
- Add the authentication method of the identity provider to the services default policy.

See the *Setting Up Resources in Directories Management* guide for information about adding and customizing resources that you add to the catalog.

Configure Additional Workspace Identity Providers

When you configure a Directories Management connector to authenticate users, a Workspace IDP is created and password authentication is enabled.

You can configure additional connectors to operate behind multiple load balancers. When your deployment includes more than one load balancer, you can configure additional Workspace identity providers for authentication in each load balancer configuration.

Procedure

1 Select **Administration > Directories Management > Identity Providers**.

This page displays all configured Identity Providers.

2 Click **Add Identity Provider**.

A menu appears with Identity Provider options.

3 Select **Create Workspace IDP**.

4 Enter the appropriate information to configure the identity provider.

Option	Description
Identity Provider Name	Enter the name for this built-in identity provider instance.
Users	Select the users to authenticate. The configured directories are listed.
Users	Select the group of users who can authenticate using this Workspace identity provider.
Network	The existing network ranges configured in the service are listed. Select the network range for the users based on the IP addresses that you want to direct to this identity provider instance for authentication.
Authentication Methods	Authentication methods that are configured for the service are displayed. Select the check box for the authentication methods to associate with this identity provider. For device compliance and Password, with AirWatch and AirWatch Connector, ensure that the option is enabled on the AirWatch configuration page.

5 Click **Add**.

Configure a Built-in Identity Provider Connection

You can configure multiple built-in identity providers and associate authentication methods with them.

Prerequisites

If you are using Built-in Keberos authentication, download the KDC issuer certificate to use in the AirWatch configuration of the iOS device management profile.

Procedure

1 Select **Administration > Directories Management > Identity Providers**.

This page displays all configured Identity Providers.

2 Click **Add Identity Provider**.

A menu appears with Identity Provider options.

3 Select **Create Built-in IDP**.

4 Enter the appropriate information to configure the identity provider.

Option	Description
Identity Provider Name	Enter the name for this built-in identity provider instance.
Users	Select the users to authenticate. The configured directories are listed.
Network	The existing network ranges configured in the service are listed. Select the network range for the users based on the IP addresses that you want to direct to this identity provider instance for authentication.
Authentication Methods	The authentication methods that are configured for the service are displayed. Select the check box for the authentication methods to associate with this identity provider. For device compliance and Password, with AirWatch and AirWatch Connector, ensure that the appropriate option is enabled on the AIrWatch configuration page.

5 Click **Add**.

Integrating Alternative User Authentication Products with Directories Management

Typically, when you initially configure Directories Management, you use the connectors supplied with your existing vRealize Automation infrastructure to create an Active Directory connection for user ID and password based authentication and management. Alternatively, you can integrate Directories Management with other authentication solutions such as Kerberos or RSA SecurID.

The identity provider instance can be the Directories Management connector instance, third-party identity provider instances, or a combination of both.

The identity provider instance that you use with the Directories Management service creates an in-network federation authority that communicates with the service using SAML 2.0 assertions.

When you initially deploy the Directories Management service, the connector is the initial identity provider for the service. Your existing Active Directory infrastructure is used for user authentication and management.

The following authentication methods are supported. You configure these authentication methods from the administration console.

Table 4-8. User Authentication Types Supported by Directories Management

Authentication Types	Description
Password (on-premise deployment)	Without any configuration after Active Directory is configured, Directories Management supports Active Directory password authentication. This method authenticates users directly against Active Directory.
Kerberos for desktops	Kerberos authentication provides domain users with single sign-in access to their apps portal. Users do not need to sign in again after they sign in to the network.
Certificate (on-premise deployment)	Certificate-based authentication can be configured to allow clients to authenticate with certificates on their desktop and mobile devices or to use a smart card adapter for authentication. Certificate-based authentication is based on what the user has and what the person knows. An X.509 certificate uses the public key infrastructure standard to verify that a public key contained within the certificate belongs to the user.
RSA SecurID (on-premise deployment)	When RSA SecurID authentication is configured, Directories Management is configured as the authentication agent in the RSA SecurID server. RSA SecurID authentication requires users to use a token-based authentication system. RSA SecurID is an authentication method for users accessing Directories Management from outside the enterprise network.
RADIUS (on-premise deployment)	RADIUS authentication provides two-factor authentication options. You set up the RADIUS server that is accessible to the Directories Management service. When users sign in with their user name and passcode, an access request is submitted to the RADIUS server for authentication.
RSA Adaptive Authentication (on-premise deployment)	RSA authentication provides a stronger multi-factor authentication than only user name and password authentication against Active Directory. When RSA Adaptive Authentication is enabled, the risk indicators specified in the risk policy set up in the RSA Policy Management application. The Directories Management service configuration of adaptive authentication is used to determine the required authentication prompts.
Mobile SSO (for iOS)	Mobile SSO for iOS authentication is used for single sign-on authentication for AirWatch-managed iOS devices. Mobile SSO (for iOS) authentication uses a Key Distribution Center (KDC) that is part of the Directories Management service. You must initiate the KDC service in the VMware Identity Manager service before you enable this authentication method.
Mobile SSO (for Android)	Mobile SSO for Android authentication is used for single sign-on authentication for AirWatch-managed Android devices. A proxy service is set up between the Directories Management service and AirWatch to retrieve the certificate from AirWatch for authentication.
Password (AirWatch Connector)	The AirWatch Cloud Connector can be integrated with the Directories Management service for user password authentication. You configure the Directories Managementservice to sync users from the AirWatch directory.

Users are authenticated based on the authentication methods, the default access policy rules, network ranges, and the identity provider instance you configure. After the authentication methods are configured, you create access policy rules that specify the authentication methods to be used by device type.

- Configuring SecurID for Directories Management

  When you configure RSA SecurID server, you must add the Directories Management service information as the authentication agent on the RSA SecurID server and configure the RSA SecurID server information on the Directories Management service.

- Configuring RADIUS for Directories Management

  You can configure Directories Management so that users are required to use RADIUS (Remote Authentication Dial-In User Service) authentication. You configure the RADIUS server information on the Directories Management service.

- Configuring a Certificate or Smart Card Adapter for Use with Directories Management

  You can configure x509 certificate authentication to allow clients to authenticate with certificates on their desktop and mobile devices or to use a smart card adapter for authentication. Certificate-based authentication is based on what the user has (the private key or smart card), and what the person knows (the password to the private key or the smart-card PIN.) An X.509 certificate uses the public key infrastructure (PKI) standard to verify that a public key contained within the certificate belongs to the user. With smart card authentication, users connect the smart card with the computer and enter a PIN.

- Configuring a Third-Party Identity Provider Instance to Authenticate Users

  You can configure a third-party identity provider to be used to authenticate users in the Directories Management service.

- Managing Authentication Methods to Apply to Users

  The Directories Management service attempts to authenticate users based on the authentication methods, the default access policy, network ranges, and the identity provider instances you configure.

- Configuring Kerberos for Directories Management

  Kerberos authentication provides users who are successfully signed in to their Active Directory domain to access their apps portal without additional credential prompts. You enable Windows authentication to allow the Kerberos protocol to secure interactions between users' browsers and the Directories Management service. You do not need to directly configure Active Directory to make Kerberos function with your deployment.

Configuring SecurID for Directories Management

When you configure RSA SecurID server, you must add the Directories Management service information as the authentication agent on the RSA SecurID server and configure the RSA SecurID server information on the Directories Management service.

When you configure SecurID to provide additional security, you must ensure that your network is properly configured for your Directories Management deployment. For SecurID specifically, you must ensure that the appropriate port is open to enable SecurID to authenticate users outside your network.

After you run the Directories Management Setup wizard and configured your Active Directory connection, you have the information necessary to prepare the RSA SecurID server. After you prepare the RSA SecurID server for Directories Management, you enable SecurID in the administration console.

- Prepare the RSA SecurID Server

  The RSA SecurID server must be configured with information about the Directories Management appliance as the authentication agent. The information required is the host name and the IP addresses for network interfaces.

- Configure RSA SecurID Authentication

  After Directories Management is configured as the authentication agent in the RSA SecurID server, you must add the RSA SecurID configuration information to the connector.

Prepare the RSA SecurID Server

The RSA SecurID server must be configured with information about the Directories Management appliance as the authentication agent. The information required is the host name and the IP addresses for network interfaces.

Prerequisites

- Verify that one of the following RSA Authentication Manager versions is installed and functioning on the enterprise network: RSA AM 6.1.2, 7.1 SP2 and later, and 8.0 and later. The Directories Management server uses AuthSDK_Java_v8.1.1.312.06_03_11_03_16_51 (Agent API 8.1 SP1), which only supports the preceding versions of RSA Authentication Manager (the RSA SecurID server). For information about installing and configuring RSA Authentication Manager (RSA SecurID server), see RSA documentation.

Procedure

1 On a supported version of the RSA SecurID server, add the Directories Management connector as an authentication agent. Enter the following information.

Option	Description
Hostname	The host name of Directories Management.
IP address	The IP address of Directories Management.
Alternate IP address	If traffic from the connector passes through a network address translation (NAT) device to reach the RSA SecurID server, enter the private IP address of the appliance.

2 Download the compressed configuration file and extract the `sdconf.rec` file.

Be prepared to upload this file later when you configure RSA SecurID in Directories Management.

What to do next

Go to the administration console and in the Identity & Access Management tab Setup pages, select the connector and in the AuthAdapters page configure SecurID.

Configure RSA SecurID Authentication

After Directories Management is configured as the authentication agent in the RSA SecurID server, you must add the RSA SecurID configuration information to the connector.

Prerequisites

- Verify that RSA Authentication Manager (the RSA SecurID server) is installed and properly configured.
- Download the compressed file from the RSA SecurID server and extract the server configuration file.

Procedure

1 As a tenant administrator, navigate to **Administration > Directories Management > Connectors**

2 On the Connectors page, select the Worker link for the connector that is being configured with RSA SecurID.

3 Click **Auth Adapters** and then click **SecurIDIdpAdapter**.

You are redirected to the identity manager sign in page.

4 In the Authentication Adapters page SecurIDIdpAdapter row, click **Edit**.

5 Configure the SecurID Authentication Adapter page.

Information used and files generated on the RSA SecurID server are required when you configure the SecurID page.

Option	Action
Name	A name is required. The default name is SecurIDIdpAdapter. You can change this.
Enable SecurID	Select this box to enable SecurID authentication.
Number of authentication attempts allowed	Enter the maximum number of failed login attempts when using the RSA SecurID token. The default is five attempts.

Option	Action
Connector Address	Enter the IP address of the connector instance. The value you enter must match the value you used when you added the connector appliance as an authentication agent to the RSA SecurID server. If your RSA SecurID server has a value assigned to the Alternate IP address prompt, enter that value as the connector IP address. If no alternate IP address is assigned, enter the value assigned to the IP address prompt.
Agent IP Address	Enter the value assigned to the **IP address** prompt in the RSA SecurID server.
Server Configuration	Upload the RSA SecurID server configuration file. First, you must download the compressed file from the RSA SecurID server and extract the server configuration file, which by default is named `sdconf.rec`.
Node Secret	Leaving the node secret field blank allows the node secret to auto generate. It is recommended that you clear the node secret file on the RSA SecurID server and intentionally do not upload the node secret file. Ensure that the node secret file on the RSA SecurID server and on the server connector instance always match. If you change the node secret at one location, change it at the other location.

6 Click **Save**.

What to do next

Add the authentication method to the default access policy. Navigate to **Administration > Directories Management > Policies** and click **Edit Default Policy** to edit the default policy rules to add the SecurID authentication method to the rule in the correct authentication order.

Configuring RADIUS for Directories Management

You can configure Directories Management so that users are required to use RADIUS (Remote Authentication Dial-In User Service) authentication. You configure the RADIUS server information on the Directories Management service.

RADIUS support offers a wide range of alternative two-factor token-based authentication options. Because two-factor authentication solutions, such as RADIUS, work with authentication managers installed on separate servers, you must have the RADIUS server configured and accessible to the identity manager service.

When users sign in to their Workspace ONE portal and RADIUS authentication is enabled, a special login dialog box appears in the browser. Users enter their RADUS authentication user name and passcode in the login dialog box. If the RADIUS server issues an access challenge, the identity manager service displays a dialog box prompting for a second passcode. Currently support for RADIUS challenges is limited to prompting for text input.

After a user enters credentials in the dialog box, the RADIUS server can send an SMS text message or email, or text using some other out-of-band mechanism to the user's cell phone with a code. The user can enter this text and code into the login dialog box to complete the authentication.

If the RADIUS server provides the ability to import users from Active Directory, end users might first be prompted to supply Active Directory credentials before being prompted for a RADIUS authentication username and passcode.

Prepare the RADIUS Server

Set up the RADIUS server and then configure it to accept RADIUS requests from the Directories Management service.

Refer to your RADIUS vendor's setup guides for information about setting up the RADIUS server. Note your RADIUS configuration information as you use this information when you configure RADIUS in the service. To view the type of RADIUS information required to configure Directories Management see Configure RADIUS Authentication in Directories Management.

You can set up a secondary Radius authentication server to be used for high availability. If the primary RADIUS server does not respond within the server timeout configured for RADIUS authentication, the request is routed to the secondary server. When the primary server does not respond, the secondary server receives all future authentication requests.

Configure RADIUS Authentication in Directories Management

You enable RADIUS software on an authentication manager server. For RADIUS authentication, follow the vendor's configuration documentation.

Prerequisites

Install and configure the RADIUS software on an authentication manager server. For RADIUS authentication, follow the vendor's configuration documentation.

You need to know the following RADIUS server information to configure RADIUS on the service.

- IP address or DNS name of the RADIUS server.
- Authentication port numbers. Authentication port is usually 1812.
- Authentication type. The authentication types include PAP (Password Authentication Protocol), CHAP (Challenge Handshake Authentication Protocol), MSCHAP1, MSCHAP2 (Microsoft Challenge Handshake Authentication Protocol, versions 1 and 2).
- RADIUS shared secret that is used for encryption and decryption in RADIUS protocol messages.
- Specific timeout and retry values needed for RADIUS authentication.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Directories Management > Connectors**.

2 On the Connectors page, select the Worker link for the connector that is being configured for RADIUS authentication.

3 Click **Auth Adapters** and then click **RadiusAuthAdapter**.

   You are redirected to the identity manager sign-in page.

4 Click **Edit** to configure these fields on the Authentication Adapter page.

Option	Action
Name	A name is required. The default name is RadiusAuthAdapter. You can change this.
Enable Radius Adapter	Select this box to enable RADIUS authentication.
Number of authentication attempts allowed	Enter the maximum number of failed login attempts when using RADIUS to log in. The default is five attempts.
Number of attempts to Radius server	Specify the total number of retry attempts. If the primary server does not respond, the service waits for the configured time before retrying again.
Radius server hostname/ address	Enter the host name or the IP address of the RADIUS server.
Authentication port	Enter the Radius authentication port number. This is usually 1812.
Accounting port	Enter 0 for the port number. The accounting port is not used at this time.
Authentication type	Enter the authentication protocol that is supported by the RADIUS server. Either PAP, CHAP, MSCHAP1, OR MSCHAP2.
Shared secret	Enter the shared secret that is used between the RADIUS server and the VMware Identity Manager service.
Server timeout in seconds	Enter the RADIUS server timeout in seconds, after which a retry is sent if the RADIUS server does not respond.
Realm Prefix	(Optional) The user account location is called the realm. If you specify a realm prefix string, the string is placed at the beginning of the user name when the name is sent to the RADIUS server. For example, if the user name is entered as jdoe and the realm prefix DOMAIN-A\ is specified, the user name DOMAIN-A\jdoe is sent to the RADIUS server. If you do not configure these fields, only the user name that is entered is sent.
Realm Suffix	(Optional) If you specify a realm suffix, the string is placed at end of the user name. For example, if the suffix is @myco.com, the username jdoe@myco.com is sent to the RADIUS server.
Login page passphrase hint	Enter the text string to display in the message on the user login page to direct users to enter the correct Radius passcode. For example, if this field is configured with **AD password first and then SMS passcode**, the login page message would read **Enter your AD password first and then SMS passcode.** The default text string is **RADIUS Passcode**.

5 You can enable a secondary RADIUS server for high availability.

Configure the secondary server as described in step 4.

6 Click **Save**.

What to do next

Add the RADIUS authentication method to the default access policy. Select **Administration > Directories Management > Policies** and click **Edit Default Policy** to edit the default policy rules to add the RADIUS authentication method to the rule in the correct authentication order.

Configuring a Certificate or Smart Card Adapter for Use with Directories Management

You can configure x509 certificate authentication to allow clients to authenticate with certificates on their desktop and mobile devices or to use a smart card adapter for authentication. Certificate-based authentication is based on what the user has (the private key or smart card), and what the person knows (the password to the private key or the smart-card PIN.) An X.509 certificate uses the public key infrastructure (PKI) standard to verify that a public key contained within the certificate belongs to the user. With smart card authentication, users connect the smart card with the computer and enter a PIN.

The smart card certificates are copied to the local certificate store on the user's computer. The certificates in the local certificate store are available to all the browsers running on this user's computer, with some exceptions.

Note When certificate authentication is configured and the service appliance is set up behind a load balancer, make sure that the connector is configured with SSL pass-through at the load balancer and not configured to terminate SSL at the load balancer. This configuration ensures that the SSL handshake is between the connector and the client to pass the certificate to the connector. You can configure additional connectors behind another load balancer configured with SSL pass-through and enable and configure certificate-based authentication on those connectors.

Using User Principal Name for Certificate Authentication

You can use certificate mapping in Active Directory. Certificate and smart card logins uses the user principal name (UPN) from Active Directory to validate user accounts. The Active Directory accounts of users attempting to authenticate in the Directories Management service must have a valid UPN that corresponds to the UPN in the certificate.

You can configure the Directories Management to use an email address to validate the user account if the UPN does not exist in the certificate.

You can also enable an alternate UPN type to be used.

Certificate Authority Required for Authentication

To enable logging in using certificate authentication, root certificates and intermediate certificates must be uploaded to the Directories Management.

The certificates are copied to the local certificate store on the user's computer. The certificates in the local certificate store are available to all the browsers running on this user's computer, with some exceptions, and therefore, are available to a Directories Management instance in the browser.

For smart-card authentication, when a user initiates a connection to the Directories Management instance, the Directories Management service sends a list of trusted certificate authorities (CA) to the browser. The browser checks the list of trusted CAs against the available user certificates, selects a suitable certificate, and then prompts the user to enter a smart card PIN. If multiple valid user certificates are available, the browser prompts the user to select a certificate.

If a user cannot authenticate, the root CA and intermediate CA might not be set up correctly, or the service has not been restarted after the root and intermediate CAs were uploaded to the server. In these cases, the browser cannot show the installed certificates, the user cannot select the correct certificate, and certificate authentication fails.

Using Certificate Revocation Checking

You can configure certificate revocation checking to prevent users who have their user certificates revoked from authenticating. Certificates are often revoked when a user leaves an organization, loses a smart card, or moves from one department to another.

Certificate revocation checking with certificate revocation lists (CRLs) and with the Online Certificate Status Protocol (OCSP) is supported. A CRL is a list of revoked certificates published by the CA that issued the certificates. OCSP is a certificate validation protocol that is used to get the revocation status of a certificate.

You can configure both CRL and OCSP in the same certificate authentication adapter configuration. When you configure both types of certificate revocation checking and the Use CRL in case of OCSP failure check box is enabled, OCSP is checked first and if OCSP fails, revocation checking falls back to CRL. Revocation checking does not fall back to OCSP if CRL fails.

Logging in with CRL Checking

When you enable certificate revocation, the Directories Management server reads a CRL to determine the revocation status of a user certificate.

If a certificate is revoked, authentication through the certificate fails.

Logging in with OCSP Certificate Checking

When you configure Certificate Status Protocol (OCSP) revocation checking, Directories Management sends a request to an OCSP responder to determine the revocation status of a specific user certificate. The Directories Management server uses the OCSP signing certificate to verify that the responses it receives from the OCSP responder are genuine.

If the certificate is revoked, authentication fails.

You can configure authentication to fall back to CRL checking if it does not receive a response from the OSCP responder or if the response is invalid.

Configure Certificate Authentication for Directories Management

You enable and configure certificate authentication from the vRealize Automation administration console Directories Management feature.

Note A system administrator must configure an external connector for your vRealize Automation deployment if you are using third party identity providers such as Keberos or smart card authentication.

Prerequisites

- Obtain the Root certificate and intermediate certificates from the CA that signed the certificates presented by your users.
- (Optional) List of Object Identifier (OID)s of valid certificate policies for certificate authentication.
- For revocation checking, the file location of the CRL, the URL of the OCSP server.
- (Optional) OCSP Response Signing certificate file location.
- Consent form content, if enabling a consent form to display before authentication.

Procedure

1. As a tenant administrator, navigate to **Administration > Directories Management > Connectors**
2. On the Connectors page, select the Worker link for the connector that is being configured.
3. Click **Auth Adapters** and then click **CertificateAuthAdapter**.

   You are redirected to the identity manager sign in page.
4. In the CertificateAuthAdapter row, click **Edit**.
5. Configure the Certificate Authentication Adapter page.

   Note An asterisk indicates a required field. All other fields are optional.

Option	Description
***Name**	A name is required. The default name is CertificateAuthAdapter. You can change this name.
Enable certificate adapter	Select the check box to enable certificate authentication.
***Root and intermediate CA certificates**	Select the certificate files to upload. You can select multiple root CA and intermediate CA certificates that are encoded as DER or PEM.
Uploaded CA certificates	The uploaded certificate files are listed in the Uploaded Ca Certificates section of the form. You must restart the service before the new certificates are made available. Click **Restart Web Service** to restart the service and add the certificates to the trusted service. **Note** Restarting the service does not enable certificate authentication. After the service is restarted, continue configuring this page. Clicking **Save** at the end of the page enables certificate authentication on the service.
Use email if no UPN in certificate	If the user principal name (UPN) does not exist in the certificate, select this checkbox to use the emailAddress attribute as the Subject Alternative Name extension to validate user accounts.

Option	Description
Certificate policies accepted	Create a list of object identifiers that are accepted in the certificate policies extensions. Enter the object ID numbers (OID) for the Certificate Issuing Policy. Click **Add another value** to add additional OIDs.
Enable cert revocation	Select the check box to enable certificate revocation checking. This prevents users who have revoked user certificates from authenticating.
Use CRL from certificates	Select the check box to use the certificate revocation list (CRL) published by the CA that issued the certificates to validate a certificate's status, revoked or not revoked.
CRL Location	Enter the server file path or the local file path from which to retrieve the CRL.
Enable OCSP Revocation	Select the check box to use the Online Certificate Status Protocol (OCSP) certificate validation protocol to get the revocation status of a certificate.
Use CRL in case of OCSP failure	If you configure both CRL and OCSP, you can check this box to fall back to using CRL if OCSP checking is not available.
Send OCSP Nonce	Select this check box if you want the unique identifier of the OCSP request to be sent in the response.
OCSP URL	If you enabled OCSP revocation, enter the OCSP server address for revocation checking.
OCSP responder's signing certificate	Enter the path to the OCSP certificate for the responder, */path/to/file.cer*.
Enable consent form before authentication	Select this check box to include a consent form page to appear before users log in to their My Apps portal using certificate authentication.
Consent form content	Type the text that displays in the consent form in this text box.

6 Click **Save**.

What to do next

- Add the certificate authentication method to the default access policy.Navigate to **Administration > Directories Management > Policies** and click **Edit Default Policy** to edit the default policy rules and add Certificate and make it the first authentication method for the default policy. Certificate must be first authentication method listed in the policy rule, otherwise certificate authentication fails.
- When Certificate Authentication is configured, and the service appliance is set up behind a load balancer, make sure that the Directories Management connector is configured with SSL pass-through at the load balancer and not configured to terminate SSL at the load balancer. This configuration ensures that the SSL handshake is between the connector and the client in order to pass the certificate to the connector.

Configuring a Third-Party Identity Provider Instance to Authenticate Users

You can configure a third-party identity provider to be used to authenticate users in the Directories Management service.

Complete the following tasks prior to using the administration console to add the third-party identity provider instance.

- Verify that the third-party instances are SAML 2.0 compliant and that the service can reach the third-party instance.
- Obtain the appropriate third-party metadata information to add when you configure the identity provider in the administration console. The metadata information you obtain from the third-party instance is either the URL to the metadata or the actual metadata.

Configure a Third Party Identity Provider Connection

vRealize Automation is supplied with a default identity provider connection instance. Users may want to create additional identity provider connections to support just-in-time user provisioning or other custom configurations.

vRealize Automation is supplied with an default identity provider. In most cases, the default provider is sufficient for customer needs. If you use an existing enterprise identity management solution, you can set up a custom identity provider to redirect users to your existing identity solution.

When using a custom identity provider, Directories Management uses SAML metadata from that provider to establish a trust relationship with the provider. After this relationship is established, Directories Management maps the users from the SAML assertion to the list of internal vRealize Automation users based the subject name ID.

Prerequisites

- Configure the network ranges that you want to direct to this identity provider instance for authentication. See Add or Edit a Network Range.
- Access to the third-party metadata document. This can be either the URL to the metadata or the actual metadata.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Directories Management > Identity Providers**.

This page displays all configured Identity Providers.

2 Click **Add Identity Provider**.

A menu appears with Identity Provider options.

3 Select **Create Third Party IDP**.

4 Enter the appropriate information to configure the identity provider.

Option	Description
Identity Provider Name	Enter a name for this identity provider instance.
SAML Metadata	Add the third party IdPs XML-based metadata document to establish trust with the identity provider. 1 Enter the SAML metadata URL or the xml content into the text box. 2 Click **Process IdP Metadata**. The NameID formats supported by the IdP are extracted from the metadata and added to the Name ID Format table. 3 In the Name ID value column, select the user attribute in the service to map to the ID formats displayed. You can add custom third-party name ID formats and map them to the user attribute values in the service. 4 (Optional) Select the NameIDPolicy response identifier string format.
Users	Select the Directories Management directories of the users that can authenticate using this identity provider.
Just-in-Time User Provisioning	Select the appropriate options to support just-in-time provisioning using an appropriate third party identity provider. Enter the **Directory Name** to use for just-in-time provisioning. Enter one or more **Domains** that exist within the external identity provider that you will use for just-in-time provisioning.
Network	The existing network ranges configured in the service are listed. Select the network ranges for the users, based on their IP addresses, that you want to direct to this identity provider instance for authentication.
Authentication Methods	Add the authentication methods supported by the third-party identity provider. Select the SAML authentication context class that supports the authentication method.
SAML Signing Certificate	Click **Service Provider (SP) Metadata** to see URL to Directories Management SAML service provider metadata URL . Copy and save the URL. This URL is configured when you edit the SAML assertion in the third-party identity provider to map Directories Management users.
Hostname	If the **Hostname** field displays, enter the hostname where the identity provider is redirected to for authentication. If you are using a non-standard port other than 443, you can set this as Hostname:Port. For example, myco.example.com:8443.

5 Click **Add**.

What to do next

- Copy and save the Directories Management service provider metadata that is required to configure the third-party identity provider instance. This metadata is available either in the SAML Signing Certificate section of the Identity Provider page.
- Add the authentication method of the identity provider to the services default policy.

See the *Setting Up Resources in Directories Management* guide for information about adding and customizing resources that you add to the catalog.

Managing Authentication Methods to Apply to Users

The Directories Management service attempts to authenticate users based on the authentication methods, the default access policy, network ranges, and the identity provider instances you configure.

When users attempt to log in, the service evaluates the default access policy rules to select which rule in the policy to apply. The authentication methods are applied in the order they are listed in the rule. The first identity provider instance that meets the authentication method and network range requirements of the rule is selected. The user authentication request is forwarded to the identity provider instance for authentication. If authentication fails, the next authentication method configured in the rule is applied.

You can add rules that specify the authentication methods to be used by either the device type or by the device type and from a specific network range. For example, you might configure a rule that requires users who sign in using iOS devices from a specific network to authenticate using RSA SecurID. Then configure another rule that requires users who sign in using any type of device from the internal network IP address to authenticate using their password.

Add or Edit a Network Range

You can manage the network ranges to define the IP addresses from which users can log in via an Active Directory link. You add the network ranges you create to specific identity provider instances and to access policy rules.

Define network ranges for your Directories Management deployment based on your network topology.

One network range, called ALL RANGES, is created as the default. This network range includes every IP address available on the Internet, 0.0.0.0 to 255.255.255.255. Even if your deployment has a single identity provider instance, you can change the IP address range and add other ranges to exclude or include specific IP addresses to the default network range. You can create other network ranges with specific IP addresses that you can apply for specific purpose.

Note The default network range, ALL RANGES, and its description, "a network for all ranges," are editable. You can edit the name and description, including changing the text to a different language, by clicking the network range name on the Network Ranges page.

Prerequisites

- You have configured tenants for your vRealize Automation deployment set up an appropriate Active Directory link to support basic Active Directory user ID and password authentication.
- Active Directory is installed and configured for use on your network.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Directories Management > Network Ranges**.

2 Edit an existing network range or add a new network range.

Option	Description
Edit an existing range	Click the network range name to edit.
Add a range	Click **Add Network Range** to add a new range.

3 Complete the form.

Form Item	Description
Name	Enter a name for the network range.
Description	Enter a description for the Network Range.
View Pods	The View Pods option only appears when the View module is enabled. Client Access URL Host. Enter the correct Horizon Client access URL for the network range. Client Access Port. Enter the correct Horizon Client access port number for the network range.
IP Ranges	Edit or add IP ranges until all desired and no undesired IP addresses are included.

What to do next

- Associate each network range with an identity provider instance.
- Associate network ranges with access policy rule as appropriate. See Configuring Access Policy Settings.

Select Attributes to Sync with Directory

When you set up the Directories Management directory to sync with Active Directory, you specify the user attributes that sync to the directory. Before you set up the directory, you can specify on the User Attributes page which default attributes are required and, if you want, add additional attributes that you want to map to Active Directory attributes.

When you configure the User Attributes page before the directory is created, you can change default attributes from required to not required, mark attributes as required, and add custom attributes.

For a list of the default mapped attributes, see Managing User Attributes that Sync from Active Directory.

After the directory is created, you can change a required attribute to not be required, and you can delete custom attributes. You cannot change an attribute to be a required attribute.

When you add other attributes to sync to the directory, after the directory is created, go to the directory's Mapped Attributes page to map these attributes to Active Directory Attributes.

Procedure

1 Log in to vRealize Automation as a system or tenant administrator.

2 Click the Administration tab.

3 Select **Directories Management > User Attributes**

4 In the Default Attributes section, review the required attribute list and make appropriate changes to reflect what attributes should be required.

5 In the Attributes section, add the Directories Management directory attribute name to the list.

6 Click **Save**.

   The default attribute status is updated and attributes you added are added on the directory's Mapped Attributes list.

7 After the directory is created, go to the Identity Stores page and select the directory.

8 Click **Sync Settings > Mapped Attributes**.

9 In the drop-down menu for the attributes that you added, select the Active Directory attribute to map to.

10 Click **Save.**

Results

The directory is updated the next time the directory syncs to the Active Directory.

Applying the Default Access Policy

The Directories Management service includes a default access policy that controls user access to their Workspace ONE portals and their Web applications. You can edit the policy to change the policy rules as necessary.

When you enable authentication methods other than password authentication, you must edit the default policy to add the enabled authentication method to the policy rules.

Each rule in the default access policy requires that a set of criteria be met to allow user access to the applications portal. You apply a network range, select which type of user can access content, and select the authentication methods to use. See Managing Access Policies.

The number of attempts the service makes to log in a user using a given authentication method varies. The service only makes one attempt at authentication for Kerberos or certificate authentication. If the attempt is not successful in logging in a user, the next authentication method in the rule is attempted. The maximum number of failed login attempts for Active Directory password and RSA SecurID authentication is set to five by default. When a user has five failed login attempts, the service attempts to log in the user with the next authentication method on the list. When all authentication methods are exhausted, the service issues an error message.

Apply Authentication Methods to Policy Rules

Only the password authentication method is configured in the default policy rules. You must edit the policy rules to select the other authentication methods you configured and set the order in which the authentication methods are used for authentication.

Prerequisites

Enable and configure the authentication methods that your organization supports. See Integrating Alternative User Authentication Products with Directories Management

Procedure

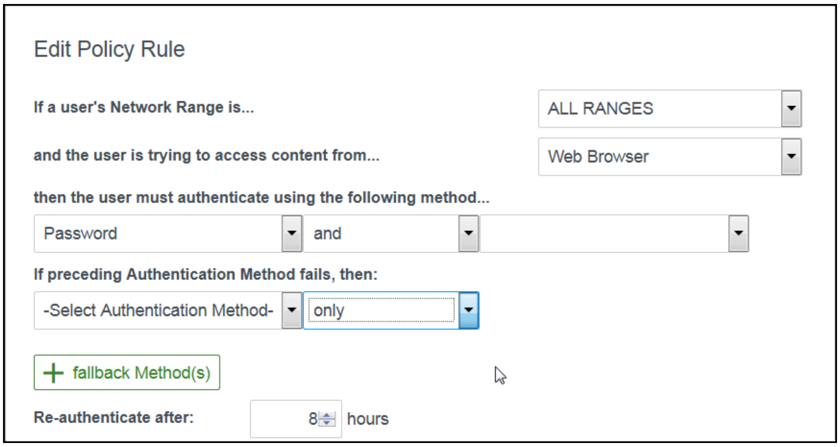
1 Select **Administration > Directories Management > Policies**

2 Click the default access policy to edit.

3 To edit a policy rule, click the authentication method to edit in the Policy Rules, Authentication Method column.

  The add a new policy rule, click the **+** icon.

4 Click **Save** and click **Save** again on the Policy page.



Edit Policy Rule

If a user's Network Range is... ALL RANGES

and the user is trying to access content from... Web Browser

then the user must authenticate using the following method...

Password | and |

If preceding Authentication Method fails, then:

-Select Authentication Method- | only

Re-authenticate after: 8 hours

5 Click **Save** and click **Save** again on the Policy page.

Configuring Kerberos for Directories Management

Kerberos authentication provides users who are successfully signed in to their Active Directory domain to access their apps portal without additional credential prompts. You enable Windows authentication to allow the Kerberos protocol to secure interactions between users' browsers and the Directories Management service. You do not need to directly configure Active Directory to make Kerberos function with your deployment.

Currently, interactions between a user's browser and the service are authenticated by Kerberos on the Windows operating systems only. Accessing the service from other operating systems does not take advantage of Kerberos authentication.

- Configure Kerberos Authentication

  To configure the Directories Management service to provide Kerberos authentication, you must join to the domain and enable Kerberos authentication on the Directories Management connector.

- Configure Internet Explorer to Access the Web Interface

  You must configure the Internet Explorer browser if Kerberos is configured for your deployment and if you want to grant users access to the Web interface using Internet Explorer.

- Configure Firefox to Access the Web Interface

  You must configure the Firefox browser if Kerberos is configured for your deployment and you want to grant users access to the Web interface using Firefox.

- Configure the Chrome Browser to Access the Web Interface

  You must configure the Chrome browser if Kerberos is configured for your deployment and if you want to grant users access to the Web interface using the Chrome browser.

Configure Kerberos Authentication

To configure the Directories Management service to provide Kerberos authentication, you must join to the domain and enable Kerberos authentication on the Directories Management connector.

Prerequisites

- Deploy an NSX Edge in your VCenter and configure an NSX Load Balancer. See *vRealize Automation Load Balancing* for information about setting up a load balancer.
- Join your domain to the master tenant. You must do this prior to creating directory connections in separate tenants.

  a Log in to the default tenant as administrator@vsphere.local.

  b Create a local user TestUser and enter TestUser as your tenant administrator.

  c Select **Administration > Directories Management > Connectors**.

  d Select Join Domain on each appliance connector.

  e On Join Domain. select Custom Domain and enter the domain that you want the tenant to connect to along with credentials and OU to connect.

- Set up directory connections for default tenants and for non-default tenants. Kerberos authentication works with both Integrated Windows Authentication and with Active Directory over LDAP. See Configure an Active Directory over LDAP/IWA Link and Configure an OpenLDAP Directory Connection.
- Ensure that the vRealize Automation node hostname matches the Active Directory domain that it is joining. For example, if vRealize Automation is joining an Active Directory realm called COMPANY.COM, the hostname should be node.company.com.
- Configure a workspace identity provider. Ensure that all nodes in your deployment are registered in your workspace identity provider and that your load balancer name is defined.

  a Select **Administration > Directories Management > Identity Providers**.

  b Select the appropriate Identity Provider link.

  For example, WorkspaceIDP_1.

  c Click the Identity Provider link and find the configured IdP hostname. Record the hostname as you will need it when configuring your web browsers.

d Register all applicable nodes in the workspace IdP, and enter the load balancer FQDN for the hostname.

e Click **Save**.

- Configure your tenant directory for the default tenant. See PLUGINS_ROOT/com.vmware.vra.install.upgrade.doc/GUID-6B4540C3-89BA-42B3-B4EB-3859BF1F17EE.html.

Procedure

1 As a tenant administrator, navigate to **Administration > Directories Management > Connectors**.

2 On the Connectors page, for the connector that is being configured for Kerberos authentication, click **Join Domain**.

3 On the Join Domain page, enter the information for the Active Directory domain.

Option	Description
Domain	Enter the fully qualified domain name of the Active Directory. The domain name you enter must be the same Windows domain as the connector server.
Domain User	Enter the user name of an account in the Active Directory that has permissions to join systems to that Active Directory domain.
Domain Password	Enter the password associated with the AD Username. This password is not stored by Directories Management .

Click **Save**.

The Join Domain page is refreshed and displays a message that you are currently joined to the domain.

4 In the Worker column for the connector click **Auth Adapters**.

5 Click **KerberosIdpAdapter**

You are redirected to the identity manager sign in page.

6 Click **Edit** in the KerberosIdpAdapter row and configure the Kerberos authentication page.

Option	Description
Name	A name is required. The default name is KerberosIdpAdapter. You can change this.
Directory UID Attribute	Enter the account attribute that contains the user name.
Enable Windows Authentication	Select this to extend authentication interactions between users' browsers and Directories Management.

Option	Description
Enable NTLM	Select this to enable NT LAN Manager (NTLM) protocol-based authentication only if your Active Directory infrastructure relies on NTLM authentication.
Enable Redirect	Select this if round-robin DNS and load balancers do not have Kerberos support. Authentication requests are redirected to Redirect Host Name. If this is selected, enter the redirect host name in **Redirect Host Name** text box. This is usually the hostname of the service.

7 Click **Save**.

8 Configure Kerberos authentication on all applicable nodes.

  a Select **Administration > Directories Management > Connectors**.

  This page shows currently configured connectors. By default, only password authentication is configured.

  b Click the worker hyperlink associated with the first vRealize Automation appliance.

  c Click the KerberosIdpAdapter link to open the authentication page.

  You may need to enter your password and relaunch the KerberosIdpAdapter link.

  d Provide the Directory UID attribute and enter the default value sAMAAccountName.

  e Select the **Enable Windows Authentication** and **Enable Redirect** check boxes.

  f Leave **NTLM** unchecked, as it is necessary only for older domain controllers.

  g Enter the name of the VA1 appliance for the Redirect Hostname.

  h Click **Save**.

9 Configure a default access policy. Kerberos configuration requires three access policies: Kerberos, password, local password.

  a Select **Administration > Directories Management > Policies**.

  b Select default_access_policy_set.

  c Click the hyperlinked value Password under the Authentication Methods heading on the web browser line.

  d Click the green + icons to create new authentication methods for Kerberos, password, and Password (Local Directory).

  e For each authentication method, select ALL RANGES as the users network range and Web Browser as the the user's content access method.

  f Change the first authentication method to Kerberos, and set the failback method to password.

  g Click **Save** and then **OK**.

Configure Internet Explorer to Access the Web Interface

You must configure the Internet Explorer browser if Kerberos is configured for your deployment and if you want to grant users access to the Web interface using Internet Explorer.

Kerberos authentication works in conjunction with Directories Management on Windows operating systems.

Note Do not implement these Kerberos-related steps on other operating systems.

Prerequisites

Configure the Internet Explorer browser for each user or provide users with the instructions after you configure Kerberos.

Procedure

1 Verify that you are logged into Windows as a user in the domain.

2 In Internet Explorer, enable automatic log in.

  a Select **Tools > Internet Options > Security**.

  b Click **Custom level**.

  c Select **Automatic login only in Intranet zone**.

  d Click **OK.**

3 Verify that this instance of the connector virtual appliance is part of the local intranet zone.

  a Use Internet Explorer to access the Directories Management sign in URL at *https://myconnectorhost.domain/authenticate/*.

  b Locate the zone in the bottom right corner on the status bar of the browser window.

  If the zone is Local intranet, Internet Explorer configuration is complete.

4 If the zone is not Local intranet, add the Directories Management sign in URL to the intranet zone.

  a Select **Tools > Internet Options > Security > Local intranet > Sites.**

  b Select **Automatically detect intranet network**.

  If this option was not selected, selecting it might be sufficient for adding the to the intranet zone.

  c (Optional) If you selected **Automatically detect intranet network**, click **OK** until all dialog boxes are closed.

  d In the Local Intranet dialog box, click **Advanced.**

  A second dialog box named Local intranet appears.

  e Enter the Directories Management URL in the **Add this Web site to the zone** text box.

  https://myconnectorhost.domain/authenticate/

  f Click **Add > Close > OK**.

5 Verify that Internet Explorer is allowed to pass the Windows authentication to the trusted site.

   a In the Internet Options dialog box, click the **Advanced** tab.

   b Select **Enable Integrated Windows Authentication**.

      This option takes effect only after you restart Internet Explorer.

   c Click **OK**.

6 Log in to the Web interface to check access.

   If Kerberos authentication is successful, the test URL goes to the Web interface.

Results

The Kerberos protocol secures all interactions between this Internet Explorer browser instance and Directories Management. Now, users can use single sign-on to access their Workspace ONE portal.

Configure Firefox to Access the Web Interface

You must configure the Firefox browser if Kerberos is configured for your deployment and you want to grant users access to the Web interface using Firefox.

Kerberos authentication works in conjunction with Directories Management on Windows operating systems.

Prerequisites

Configure the Firefox browser, for each user, or provide users with the instructions, after you configure Kerberos.

Procedure

1 In the URL text box of the Firefox browser, enter `about:config` to access the advanced settings.

2 Click **I'll be careful, I promise!**.

3 Double-click **network.negotiate-auth.trusted-uris** in the Preference Name column.

4 Enter your Directories Management URL in the text box.

   https://myconnectorhost.domain.com

5 Click **OK.**

6 Double-click **network.negotiate-auth.delegation-uris** in the Preference Name column.

7 Enter your Directories Management URL in the text box.

   https://myconnectorhost.domain.com/authenticate/

8 Click **OK.**

9 Test Kerberos functionality by using the Firefox browser to log in to login URL. For example, *https://myconnectorhost.domain.com/authenticate/*.

If the Kerberos authentication is successful, the test URL goes to the Web interface.

Results

The Kerberos protocol secures all interactions between this Firefox browser instance and Directories Management. Now, users can use single sign-on access their Workspace ONE portal.

Configure the Chrome Browser to Access the Web Interface

You must configure the Chrome browser if Kerberos is configured for your deployment and if you want to grant users access to the Web interface using the Chrome browser.

Kerberos authentication works in conjunction with Directories Management on Windows operating systems.

Note Do not implement these Kerberos-related steps on other operating systems.

Prerequisites

- Configure Kerberos.
- Since Chrome uses the Internet Explorer configuration to enable Kerberos authentication, you must configure Internet Explorer to allow Chrome to use the Internet Explorer configuration. See Google documentation for information about how to configure Chrome for Kerberos authentication.

Procedure

1 Test Kerberos functionality by using the Chrome browser.

2 Log in to Directories Management at *https://myconnectorhost.domain.com/authenticate/*.

If Kerberos authentication is successful, the test URL connects with the Web interface.

Results

If all related Kerberos configurations are correct, the relative protocol (Kerberos) secures all interactions between this Chrome browser instance and Directories Management. Users can use single sign-on access their Workspace ONE portal.

Upgrading External Connectors for Directories Management

If you use an external connector with your vRealize Automation Directories Management configuration, you may need to upgrade this connector on occasion.

You may need to upgrade an external connector when upgrading the version of your vRealize Automation deployment or if a new connector build offers a feature you want.

This documentation applies only to users who have deployed additional, stand-alone external connector appliances. In vRealize Automation, external connector appliances are used with smart card authentication, for instance.

By default, the connector uses the VMware Web site for the upgrade procedure, which requires the connector appliance to have Internet connectivity. You must also configure proxy server settings for the connector appliance, if applicable.

If your connector instance does not have an Internet connection, you can perform the upgrade offline. For an offline upgrade, you download the upgrade package and set up a local Web server to host the upgrade file.

Intended Audience

This information is intended for anyone who installs, upgrades, and configures Directories Management. The information is written for experienced Windows or Linux system administrators who are familiar with virtual machine technology.

Preparing to Upgrade an External Connector

To prepare for a connector upgrade, you must check for available upgrades and configuring the proxy server settings for the appliance, if applicable.

- Check Availability of an External Connector Upgrade Online

  If your connector appliance has Internet connectivity, you can check for the availability of upgrades online from the appliance.

- Configure Proxy Server Settings for the External Connector Appliance

  The connector appliance accesses the VMware update servers through the Internet. If your network configuration provides Internet access using an HTTP proxy, you must adjust the proxy settings for the appliance.

Check Availability of an External Connector Upgrade Online

If your connector appliance has Internet connectivity, you can check for the availability of upgrades online from the appliance.

Procedure

1 Log in to the connector appliance as the root user.

2 Run the following command.

```
/usr/local/horizon/update/updatemgr.hzn updateinstaller
```

3 Run the following command to check for an online upgrade.

```
/usr/local/horizon/update/updatemgr.hzn check
```

Configure Proxy Server Settings for the External Connector Appliance

The connector appliance accesses the VMware update servers through the Internet. If your network configuration provides Internet access using an HTTP proxy, you must adjust the proxy settings for the appliance.

Enable your proxy to handle only Internet traffic. To ensure that the proxy is set up correctly, set the parameter for internal traffic to no-proxy within the domain.

Note Proxy servers that require authentication are not supported.

Prerequisites

- Verify that you have the root password for the connector appliance.
- Verify that you have the proxy server information.

Procedure

1 Log in to the connector appliance as the root user.

2 Enter YaST on the command line to run the YaST utility.

3 Select **Network Services** in the left pane, then select **Proxy**.

4 Enter the proxy server URLs in the **HTTP Proxy URL** and **HTTPS Proxy URL** fields.

5 Select **Finish** and exit the YaST utility.

6 Restart the Tomcat server on the connector virtual appliance to use the new proxy settings.

```
service horizon-workspace restart
```

Results

The VMware update servers are now available to the connector appliance.

Upgrade an External Connector Online

You can upgrade a Directories Management external connector online if you have an appropriate connection.

Prerequisites

- Verify that the connector appliance can resolve and reach vapp-updates.vmware.com on port 80 over HTTP.
- Confirm that a connector upgrade exists. Run the appropriate command to check for upgrades. See Check for the Availability of a Directories Management Connector Upgrade Online.
- Verify that at least 2 GB of disk space is available on the primary root partition of the appliance.
- Verify that the connector is properly configured.
- Take a snapshot of your connector appliance to back it up. For information about how to take snapshots, see the vSphere documentation.

- If an HTTP proxy server is required for outbound HTTP access, configure the proxy server settings for the connector appliance. See Configure Proxy Server Settings for the Directories Management Connector Appliance.

Procedure

1. Log in to the connector appliance as the root user.
2. Run the following command.

   ```
   /usr/local/horizon/update/updatemgr.hzn updateinstaller
   ```

3. Run the following command to check that on online upgrade exists.

   ```
   /usr/local/horizon/update/updatemgr.hzn check
   ```

4. Run the following command to update the appliance.

   ```
   /usr/local/horizon/update/updatemgr.hzn update
   ```

   Messages that occur during the upgrade are saved to the `update.log` file at `/opt/vmware/var/log/update.log`.

5. Run the `updatemgr.hzn check` command again to verify that a newer update does not exist.

   ```
   /usr/local/horizon/update/updatemgr.hzn check
   ```

6. Check the version of the upgraded appliance.

   ```
   vamicli version --appliance
   ```

   The new version is displayed.

7. Restart the connector appliance.

   `reboot`

Upgrade an External Connector Offline

If your existing vRealize Automation Directories Management connector appliance cannot connect to the Internet for upgrade, you can perform an offline upgrade. You must set up an upgrade repository on a local Web server and configure the connector appliance to use the local Web server for upgrade.

Prerequisites

- Confirm that a connector upgrade exists. Check the My VMware Downloads site at my.vmware.com for upgrades.
- Verify that at least 2 GB of disk space is available on the primary root partition of the appliance.
- Verify that the connector is properly configured.

- Take a snapshot of your connector appliance to back it up. For information about how to take snapshots, see the vSphere documentation.
- Configure the connector appliance to user a local Web server to host the upgrade file. See Prepare a Local Web Server for Offline Upgrade.

Procedure

1 Prepare a Local Web Server for Offline Upgrade

  Before you start the offline connector upgrade, prepare the local Web server by creating a directory structure that includes a subdirectory for the connector appliance.

2 Configure the Connector and Perform Offline Upgrade

  Configure the connector appliance to point to the local Web server to perform an offline upgrade. Then upgrade the appliance.

Prepare a Local Web Server for Offline Upgrade

Before you start the offline connector upgrade, prepare the local Web server by creating a directory structure that includes a subdirectory for the connector appliance.

Prerequisites

- Download the `identity-manager-connector-`*`versionNumber`*`-`*`buildNumber`*`-updaterepo.zip` file from My VMware. Go to my.vmware.com, navigate to the VMware Identity Manager Download page, and download the file listed under **VMware Identity Manager Connector offline upgrade package**.
- If you use an IIS Web server, configure the Web server to allow special characters in file names. You configure this in the **Request Filtering** section by selecting the **Allow double escaping** option.

Procedure

1 Create a directory on the Web server at http://*YourWebServer*/*VM*/ and copy the downloaded zip file to it.

2 Verify that your Web server includes mime types for `.sig` (text/plain) and `.sha256` (text/plain).

  Without these mime types your Web server fails to check for updates.

3 Unzip the file.

  The contents of the extracted ZIP file are served by http://*YourWebServer*/*VM*/.

  The extracted contents of the file contain the following subdirectories: `/manifest` and `/package-pool`.

4 Run the following `updatelocal.hzn` command to check that the URL has valid update contents.

```
/usr/local/horizon/update/updatelocal.hzn checkurl http://YourWebServer/VM
```

Configure the Connector and Perform Offline Upgrade

Configure the connector appliance to point to the local Web server to perform an offline upgrade. Then upgrade the appliance.

Prerequisites

Prepare a local Web server for offline upgrade.

Procedure

1 Log in to the connector appliance as the root user.

2 Run the following command to configure an upgrade repository that uses a local Web server.

```
/usr/local/horizon/update/updatelocal.hzn seturl http://YourWebServer/VM/
```

Note To undo the configuration and restore the ability to perform an online upgrade, you can run the following command.

```
/usr/local/horizon/update/updatelocal.hzn setdefault
```

3 Perform the upgrade.

a Run the following command.

```
/usr/local/horizon/update/updatemgr.hzn updateinstaller
```

b Run the following command to check the version of the available upgrade.

```
/usr/local/horizon/update/updatemgr.hzn check
```

c Run the following command to update the connector.

```
/usr/local/horizon/update/updatemgr.hzn update
```

Messages that occur during the upgrade are saved to the `update.log` file at `/opt/vmware/var/log/update.log`.

d Run the `updatemgr.hzn check` command again.

```
/usr/local/horizon/update/updatemgr.hzn check
```

e Check the version of the upgraded appliance.

```
vamicli version --appliance
```

The command should display the new version.

f Restart the connector appliance.

For example, from the command line run the following command.

`reboot`

Results

The connector upgrade is complete.

Configuring Settings After Upgrading an External Connector

After upgrading to connector 2016.3.1.0 or later, you may need to configure some settings.

Rejoin Domain with Kerberos Authentication

If you use Kerberos authentication or Active Directory (Integrated Windows Authentication) directories, you must leave the domain and then rejoin it. This is required for all the connector virtual appliances in your deployment.

1 Select **Administration > Directories Management > Connectors**

2 In the Connectors page, for each connector that is being used for Kerberos authentication or an Active Directory (Integrated Windows Authentication) directory, click **Leave Domain**.

3 To join the domain, you need Active Directory credentials with the privileges to join the domain. See Join a Connector Machine to a Domain for more information.

4 If you are using Kerberos authentication, enable the Kerberos authentication adapter again. To access the Auth Adapters page, in the Connectors page click the appropriate link in the **Worker** column and select the **Auth Adapters** tab.

5 Verify that the other authentication adapters you are using are enabled.

Update Domains Page

If you are using Active Directory (Integrated Windows Authentication), or Active Directory over LDAP with the **This Directory supports DNS Service Location** option enabled, save the directory's Domains page.

1 Select **Administration > Directories Management > Directories**

2 Select the applicable directory to edit it.

3 Provide the password for the Bind DN user and click **Save**.

4 Click **Sync Settings** on the left of the page and select the **Domains** tab.

5 Click **Save**.

DNS Service Location and Domain Controllers

Note In connector 2016.3.1.0 and later, a `domain_krb.properties` file is automatically created and auto-populated with domain controllers when a directory with DNS Service Location enabled is created. When you save the Domains page after upgrade, if you had a `domain_krb.properties` file in your original deployment, the file is updated with domains that you may have added subsequently and that were not in the file. If you did not have a `domain_krb.properties` file in your original deployment, the file is created and auto-populated with domain controllers. See About Domain Controller Selection for more information about the `domain_krb.properties` file.

Troubleshooting External Connector Upgrade Errors

You can troubleshoot vRA Directories Management external connector upgrade problems by reviewing the error logs. If the connector does not start, you can revert to a previous instance by rolling back to a snapshot.

- Checking the Upgrade Error Logs

  Resolve errors that occur during upgrade by reviewing the error logs. Upgrade log files are in the `/opt/vmware/var/log` directory.

- Rolling Back to Snapshots of Connector

  If the connector does not start properly after an upgrade, and you cannot resolve the problem by reviewing the upgrade error logs and running the upgrade command again, you can roll back to a previous connector instance.

- Collecting a Log File Bundle

  You can collect a bundle of log files to send to VMware support. You obtain the bundle from the connector configuration page.

Checking the Upgrade Error Logs

Resolve errors that occur during upgrade by reviewing the error logs. Upgrade log files are in the `/opt/vmware/var/log` directory.

If any errors occurred, the connector may not start after upgrade.

Procedure

1. Log in to the connector appliance.
2. Go to the `/opt/vmware/var/log` directory.
3. Open the `update.log` file and review the error messages.
4. Resolve the errors and rerun the upgrade command. The upgrade command resumes from the point where it stopped.

   Note Alternatively, you can revert to a snapshot and run the update again.

Rolling Back to Snapshots of Connector

If the connector does not start properly after an upgrade, and you cannot resolve the problem by reviewing the upgrade error logs and running the upgrade command again, you can roll back to a previous connector instance.

Procedure

- Revert to one of the snapshots you took as a backup of your original connector instance. For information, see the vSphere documentation.

Collecting a Log File Bundle

You can collect a bundle of log files to send to VMware support. You obtain the bundle from the connector configuration page.

The following log files are collected in the bundle.

Table 4-9. Log Files

Component	Location of Log File	Description
Apache Tomcat Logs (catalina.log)	`/opt/vmware/horizon/workspace/logs/catalina.log`	Apache Tomcat records messages that are not recorded in other log files.
Configurator Logs (configurator.log)	`/opt/vmware/horizon/workspace/logs/configurator.log`	Requests that the Configurator receives from the REST client and the Web interface.
Connector Logs (connector.log)	`/opt/vmware/horizon/workspace/logs/connector.log`	A record of each request received from the Web interface. Each log entry also includes the request URL, timestamp, and exceptions. No sync actions are recorded.

Procedure

1. Log in to the connector configuration page at https://*connectorURL*:8443/cfg/logs.
2. Click **Prepare log bundle**.
3. Download the bundle and send it to VMware support.

Scenario: Configure an Active Directory Link for a Highly Available vRealize Automation

As a tenant administrator, you want to configure an Active Directory over LDAP directory connection to support user authentication for your highly available vRealize Automation deployment.

Each vRealize Automation appliance includes a connector that supports user authentication, although only one connector is typically configured to perform directory synchronization. It does not matter which connector you choose to serve as the sync connector. To support Directories Management high availability, you must configure a second connector that corresponds to your second vRealize Automation appliance, which connects to your Identity Provider and points to the same Active Directory. With this configuration, if one appliance fails, the other takes over management of user authentication.

In a high availability environment, all nodes must serve the same set of Active Directories, users, authentication methods, etc. The most direct method to accomplish this is to promote the Identity Provider to the cluster by setting the load balancer host as the Identity Provider host. With this configuration, all authentication requests are directed to the load balancer, which forwards the request to either connector as appropriate.

Prerequisites

- Install a distributed vRealize Automation deployment with appropriate load balancers. See *Installing vRealize Automation*.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1. Select **Administration > Directories Management > Directories**.
2. Click **Add Directory**.
3. Enter your specific Active Directory account settings, and accept the default options.

Option	Sample Input
Directory Name	Add the IP address of your active directory domain name.
Sync Connector	Every vRealize Automation appliance contains a connector. Use any of the available connectors.
Base DN	Enter the Distinguished Name (DN) of the starting point for directory server searches. For example, **cn=users,dc=corp,dc=local**.
Bind DN	Enter the full distinguished name (DN), including common name (CN), of an Active Directory user account that has privileges to search for users. For example, **cn=config_admin infra,cn=users,dc=corp,dc=local**.
Bind DN Password	Enter the Active Directory password for the account that can search for users.

4. Click **Test Connection** to test the connection to the configured directory.

   If the connection fails, check your entries in all fields and consult your system administrator if necessary.

5. Click **Save & Next**.

   The Select the Domains page with the list of domains appears.

6. Leave the default domain selected and click **Next**.

7 Verify that the attribute names are mapped to the correct Active Directory attributes. If not, select the correct Active Directory attribute from the drop-down menu. Click **Next**.

8 Select the groups and users you want to sync.

   a Click the **Add** icon (✚).

   b Enter the user domain and click **Find Groups**.

   For example, `cn=users,dc=corp,dc=local`.

   c Select the **Select All** check box.

   d Click **Select**.

   e Click **Next**.

   f Click ✚ to add additional users. For example, enter as `CN–username,CN=Users,OU–myUnit,DC=myCorp,DC=com`.

   To exclude users, click + to create a filter to exclude some types of users. You select the user attribute to filter by, the query rule, and the value.

   g Click **Next**.

9 Review the page to see how many users and groups are syncing to the directory and click **Sync Directory**.

The directory sync process takes some time, but it happens in the background and you can continue working.

10 Configure a second connector to support high availability.

   a Log in to the load balancer for your vRealize Automation deployment as a tenant administrator.

   The load balancer URL is *`load balancer address`*`/vcac/org/`*`tenant_name`*.

   b Select **Administration > Directories Management > Identity Providers**.

   c Click the Identity Provider that is currently in use for your system.

   The existing directory and connector that provide basic identity management for your system appears.

   d Click the **Add a Connector** drop-down list, and select the connector that corresponds to your secondary vRealize Automation appliance.

   e Enter the appropriate password in the **Bind DN Password** text box that appears when you select the connector.

   f Click **Add Connector**.

   g Edit the host name to point to your load balancer.

Results

You connected your corporate Active Directory to vRealize Automation and configured Directories Management for high availability.

What to do next

To provide enhanced security, you can configure bi-directional trust between your identity provider and your Active Directory. See Configure a Bi Directional Trust Relationship Between vRealize Automation and Active Directory.

Configure External Connectors for Smart Card and Third-party Identity Provider Authentication in vRealize Automation

A system administrator must configure an external connector for your vRealize Automation deployment using Directories Management if you are using third party identity providers with certificate authentication or smart card authentication. Also, the procedure herein broadly applies to all types of certificate authentication.

Directories Management supports multiple identity providers and connector clusters for each configured Active Directory. To use a third-party identity provider or smart card authentication, you can set up either a single external connector or a connector cluster with an appropriate identity provider behind a load balancer that permits SSL passthrough. See Managing Connectors and Connector Clusters for more information.

See Upgrading External Connectors for Directories Management for information about updating an external connector.

There are various certificate configuration options available for use with smart card authentication. See Configuring a Certificate or Smart Card Adapter for Use with Directories Management.

Prerequisites

- Configure an appropriate Active Directory connection for use with your vRealize Automation deployment.
- Download the OVA file required to configure a connector from VMware vRealize Automation Tools and SDK.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Generate a Connector Activation Token

Before you deploy the connector virtual appliance to use for smart card authentication, generate an activation code for the new connector from the vRealize Automation console. The activation code is used to establish communication between Directories Management and the connector.

2 Deploy the Connector OVA File

After downloading a connector OVA file, you can deploy it using the VMware vSphere Client or vSphere Web Client.

3 Configure Connector Settings

After deploying the connector OVA, you must run the Setup wizard to activate the appliance and configure the administrator passwords.

4 Apply Public Certificate Authority

When Directories Management is installed, a default SSL certificate is generated. You can use the default certificate for testing purposes, but you should generate and install commercial SSL certificates for production environments.

5 Create a Workspace Identity Provider

You must create a Workspace identity provider for use with an external connector.

6 Configure Certificate Authentication and Configure Default Access Policy Rules

You must configure your external connection for use with your vRealize Automation Active Directory and domain.

Generate a Connector Activation Token

Before you deploy the connector virtual appliance to use for smart card authentication, generate an activation code for the new connector from the vRealize Automation console. The activation code is used to establish communication between Directories Management and the connector.

You can configure a single connector or a connector cluster. If you want to use a connector cluster, repeat this procedure for each connector that you need.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Directories Management > Connectors**

2 Click **Add Connector**.

3 Enter a name for the new connector in the **Connector ID Name** text box.

4 Click **Generate Activation Code**.

The activation code for the connector is displayed in the **Connector Activation Code** box.

5 Copy the activation code for use in configuring the connector using the OVA file.

6 Click **OK**.

Deploy the Connector OVA File

After downloading a connector OVA file, you can deploy it using the VMware vSphere Client or vSphere Web Client.

You deploy the OVA file using the vSphere Client or the vSphere Web Client.

Prerequisites

- Identify the DNS records and host name to use for your connector OVA deployment.
- If using the vSphere Web Client, use either Firefox or Chrome browsers. Do not use Internet Explorer to deploy the OVA file.
- Download the OVA file required to configure a connector from VMware vRealize Automation Tools and SDK.

Procedure

1 In the vSphere Client or the vSphere Web Client, select **File > Deploy OVF Template**.

2 In the Deploy OVF Template pages, enter the information specific to your deployment of the connector.

Page	Description
Source	Browse to the OVA package location, or enter a specific URL.
OVA Template Details	Verify that you selected the correct version.
License	Read the End User License Agreement and click **Accept**.
Name and Location	Enter a name for the virtual appliance. The name must be unique within the inventory folder and can contain up to 80 characters. Names are case sensitive. Select a location for the virtual appliance.
Host / Cluster	Select the host or cluster to run the deployed template.
Resource Pool	Select the resource pool.
Storage	Select the location to store the virtual machine files.
Disk Format	Select the disk format for the files. For production environments, select a **Thick Provision** format. Use the **Thin Provision** format for evaluation and testing.
Network Mapping	Map the networks in your environment to the networks in the OVF template.
Properties	a In the **Timezone setting** field, select the correct time zone. b The Customer Experience Improvement Program checkbox is selected by default. VMware collects anonymous data about your deployment in order to improve VMware's response to user requirements. Deselect the checkbox if you do not want the data collected. c In the Host Name text box, enter the host name to use. If this is blank, reverse DNS is used to look up the host name. d To configure the static IP address for connector, enter the address for each of the following: Default Gateway, DNS, IP Address, and Netmask. **Important** If any of the four address fields, including Host Name, are left blank, DHCP is used. To configure DHCP, leave the address fields blank.
Ready to Complete	Review your selections and click **Finish**.

Depending on your network speed, the deployment can take several minutes. You can view the progress in the progress dialog box.

3 When the deployment is complete, select the appliance, right-click, and select **Power > Power on**.

The appliance is initialized. You can go to the **Console** tab to see the details. When the virtual appliance initialization is complete, the console screen displays the version and URLs to log in to the Setup wizard to complete the set up.

What to do next

Use the Setup wizard to add the activation code and administrative passwords.

Configure Connector Settings

After deploying the connector OVA, you must run the Setup wizard to activate the appliance and configure the administrator passwords.

Prerequisites

- You have generated an activation code for the connector.
- Ensure the connector appliance is powered on and you know the connector URL.
- Collect a list of password to use for the connector administrator, root account, and sshuser account.

Procedure

1 To run the Setup wizard, enter the connector URL that was displayed in the Console tab after the OVA was deployed.

2 On the Welcome Page, click **Continue**.

3 Create strong passwords for the following connector virtual appliance administrator accounts.

Strong passwords should be at least eight characters long and include uppercase and lowercase characters and at least one digit or special character.

Option	Description
Appliance Administrator	Create the appliance administrator password. The user name is **admin** and cannot be changed. You use this account and password to log in to the connector services to manage certificates, appliance passwords and syslog configuration. **Important** The **admin** user password must be at least 6 characters in length.
Root Account	A default VMware root password was used to install the connector appliance. Create a new root password.
sshuser Account	Create the password to use for remote access to the connector appliance.

4 Click **Continue**.

5 On the Activate Connector page, paste in the activation code and click **Continue**.

6 If you are using a self-signed certificate on the vRealize Automation internal connector, you can get the appropriate certificate by running the following command on the vRealize Automation appliance: `cat /etc/apache2/server-cert.pem`

   Select the **Terminate SSL on a Load Balancer** tab, and then click the link for `/horizon_workspace_rootca.pem`.

   The activation code is verified and communication between the service and the connector instance is established to complete the connector configuration.

What to do next

In the service, set up your environment based on your needs. For example, if you added an additional connector because you want to sync two Integrated Windows Authentication directories, create the directory and associate it with the new connector.

Apply Public Certificate Authority

When Directories Management is installed, a default SSL certificate is generated. You can use the default certificate for testing purposes, but you should generate and install commercial SSL certificates for production environments.

If the Directories Management points to a load balancer, the SSL certificate is applied to the load balancer.

You must check the **Mark this key as exportable** when importing a certificate.

You only need to specify the CN, or certificate authority's site domain name, if you are generating a CSR for a custom certificate.

Prerequisites

Generate a Certificate Signing Request (CSR) and obtain a valid, signed certificate from a CA. If your organization provides SSL certificates that are signed by a CA, you can use these certificates. The certificate must be in the PEM format.

Procedure

1 Log in to the connector appliance administrative page as an admin user at the following location:

   `https://`*`myconnector.mycompany`*`:8443/cfg`

2 In the administrator console, click **Appliance Settings**.

   VA Configuration is selected by default.

3 Click **Manage Configurations**.

4 Enter the VMware Identify Manager server admin user password.

5. Select **Install Certificate**.
6. In the Terminate SSL on the **Identity Manager Appliance** tab, select **Custom Certificate**.
7. In the **SSL Certificate Chain** text box, paste the host, intermediate, and root certificates, in that order.

   The SSL certificate works only if you include the entire certificate chain in the correct order. For each certificate, copy everything between and including the lines -----BEGIN CERTIFICATE----- and -----END CERTIFICATE----

   Ensure that the certificate includes the FQDN hostname.
8. Paste the private key in the Private Key text box. Copy everything between ----BEGIN RSA PRIVATE KEY and ---END RSA PRIVATE KEY.
9. Click **Save**.

Example: Certificate Examples

Certificate Chain Example
-----BEGIN CERTIFICATE-----
jlQvt9WdR9Vpg3WQT5+C3HU17bUOwvhp/r0+ ... ... W53+O05j5xsxzDJfWr1lqBlFF/OklYCPcyK1
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
WdR9Vpg3WQT5+C3HU17bUOwvhp/rjlQvt90+ ... ... O05j5xsxzDJfWr1lqBlFF/OklYCPW53+cyK1
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
dR9Vpg3WQTjlQvt9W5+C3HU17bUOwvhp/r0+ ... ... 5j5xsxzDJfWr1lqW53+O0BlFF/OklYCPcyK1
-----END CERTIFICATE-----

Private Key Example
-----BEGIN RSA PRIVATE KEY-----
jlQvtg3WQT5+C3HU17bU9WdR9VpOwvhp/r0+ ... ... 1lqBlFFW53+O05j5xsxzDJfWr/OklYCPcyK1
-----END RSA PRIVATE KEY-----

Create a Workspace Identity Provider

You must create a Workspace identity provider for use with an external connector.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Directories Management > Identity Providers**.

2 Select **Add Identity Provider**.

3 Select **Create Workspace IDP**.

4 Enter a name for the identity provider in the **Identity Provider Name** field.

5 Select the directory that corresponds to the users that will use the identity provider.

The directory you select determines which connectors are available for the identity provider.

6 Select the external connector or connectors that you configured for smart card authentication.

Note If the deployment is located behind a load balancer, enter the load balancer URL.

7 Select the network to have access to the identity provider.

8 Click **Add**.

Configure Certificate Authentication and Configure Default Access Policy Rules

You must configure your external connection for use with your vRealize Automation Active Directory and domain.

Prerequisites

Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Directories Management > Connectors**.

2 Select the Desired connector in the **Worker** column.

   The selected worker is shown in the **Worker Name** text box on the Connector **Detail** tab and connector type information appears in the **Connector Type** text box.

3 Ensure that the connector is linked to the desired Active Directory by specifying that Directory in the **Associated Directory** text box.

4 Enter the appropriate domain name in the **Associated Domains** text box.

5 Select the **AuthAdapters** tab and enable CertificateAuthAdapter.

6 Configure certificate authentication as appropriate for your deployment.

   See Configure Certificate Authentication for Directories Management.

7 Select **Administration > Directories Management > Policies**.

8 Click **Edit Default Policy**.

9 Add Certificate to the policy rules and make it the first authentication method.

   Certificate must be the first authentication method listed in the policy rule, otherwise certificate authentication fails.

Create a Multi Domain or Multi Forest Active Directory Link

As a system administrator, you need to configure a multi domain or multi forest Active Directory link.

The procedure for configuring a multi domain or multi forest Active Directory link is essentially the same. For a multi forest link, bi-directional trust is required between all applicable domains.

Prerequisites

- Install a distributed vRealize Automation deployment with appropriate load balancers. See *Installing vRealize Automation*.
- Log in to vRealize Automation as a **tenant administrator**.
- Configure the appropriate domains and Active Directory forests for your deployment.

Procedure

1 Select **Administration > Directories Management > Directories**.

2 Click **Add Directory**.

3 On the Add Directory page, specify a name for the Active Directory server in the **Directory Name** text box.

4 Select **Active Directory (Integrated Windows Authentication)** under the **Directory Name** heading.

5 Configure the connector that synchronizes users from the Active Directory to the VMware Directories Management directory in the Directory Sync and Authentication section.

Option	Description
Sync Connector	Select the appropriate connector to use for your system. Each vRealize Automation appliance contains a default connector. Consult your system administrator if you need help in choosing the appropriate connector.
Authentication	Click the appropriate radio button to indicate whether the selected connector also performs authentication.
Directory Search Attribute	Select the appropriate account attribute that contains the user name.

Depending on your deployment configuration, you will have one or more connectors available for use.

6 Enter the appropriate join domain credentials in the **Domain Name**, **Domain Admin User Name**, and **Domain Admin Password** text boxes.

As an example, you might enter something like the following: **Domain Name**: `hs.trcint.com`, **Domain Admin Username**: `devadmin`, **Domain Admin Password**: `xxxx`.

7 In the **Bind User Details** section, enter the appropriate Active Directory (Integrated Windows Authentication) credentials to facilitate directory synchronization.

Option	Description
Bind User UPN	Enter the User Principal Name of the user who can authenticate with the domain. For example, UserName@example.com.
Bind DN Password	Enter the Bind User password.

8 Click **Save & Next**.

The Select the Domains page appears with the list of domains.

9 Click the appropriate check boxes to select the desired domains for your system deployment.

10 Click **Next**.

11 Verify that the Directories Management directory attribute names are mapped to the correct Active Directory attributes.

If the directory attribute names are mapped incorrectly, select the correct Active Directory attribute from the drop-down menu.

12 Click **Next**.

13 Click ✚ to select the groups you want to sync from Active Directory to the directory.

When you add an Active Directory group, if members of that group are not in the Users list, they are added.

> **Note** The Directories Management user authentication system imports data from Active Directory when adding groups and users, and the speed of the system is limited by Active Directory capabilities. As a result, import operations may require a significant amount of time depending on the number of groups and users being added. To minimize the potential for delays or problems, limit the number of groups and users to only those required for vRealize Automation operation. If your system performance degrades or if errors occur, close any unneeded applications and ensure that your system has appropriate memory allocated to Active Directory. If problems persist, increase the Active Directory memory allocation as needed. For systems with large numbers of users and groups, you may need to increase the Active Directory memory allocation to as much as 24 GB.

14 Click **Next**.

15 Click ✚ to add additional users. For example, enter as `CN–username,CN=Users,OU–myUnit,DC=myCorp,DC=com`.

To exclude users, click ✚ to create a filter to exclude some types of users. You select the user attribute to filter by, the query rule, and the value.

16 Click **Next**.

17 Review the page to see how many users and groups are syncing to the directory.

If you want to make changes to users and groups, click the Edit links.

18 Click **Push to Workspace** to start the synchronization to the directory.

What to do next

Configuring Groups and User Roles

Tenant administrators create business groups and custom groups, and grant user access rights to the vRealize Automation console.

Assign Roles to Directory Users or Groups

Tenant administrators grant access rights to users by assigning roles to users or groups.

To allow users or groups to modify and trigger a pipeline, you must assign permissions to those users and groups. When you assign users and groups the role of Release Manager, they can modify and trigger the pipeline. When you assign users and groups the role of Release Engineer, they can trigger the pipeline. For more information, see the *Using vRealize Code Stream* guide.

Prerequisites

Log in to vRealize Automation as a **tenant administrator**.

Procedure

1. Select **Administration > Users & Groups > Directory Users & Groups**.
2. Enter a user or group name in the **Search** box and press Enter.

   Do not use an at sign (@), backslash (\), or slash (/) in a name. You can optimize your search by typing the entire user or group name in the form user@domain.
3. Click the name of the user or group to which you want to assign roles.
4. Select one or more roles from the Add Roles to this User list.

   The Authorities Granted by Selected Roles list indicates the specific authorities you are granting.
5. (Optional) Click **Next** to view more information about the user or group.
6. On the **User Details** page, on the **General** tab, scroll the list of roles to add the user.
   a. To give the user permissions to modify and trigger a pipeline, select the **Release Manager** check box.
   b. To give the user permissions to trigger a pipeline, select the **Release Engineer** check box.
7. Click **Update**.

Results

Users who are currently logged in to the vRealize Automation must log out and log back in to the vRealize Automation before they can navigate to the pages to which they have been granted access.

What to do next

Optionally, you can create your own custom groups from users and groups in your Active Directory connections. See Create a Custom Group.

Create a Custom Group

Tenant administrators can create custom groups by combining other custom groups, identity store groups, and individual identity store users. Custom groups provide more granular control over access within vRealize Automation than business groups which correspond to a line of business, department, or other organizational unit.

Custom groups enable you to grant access rights for tasks on a finer basis than the standard vRealize Automation group assignments. For instance, you may want to create a custom group to allow tenant administrators to control who has specific permissions within the tenant.

You can assign roles to your custom group, but it is not necessary in all cases. For example, you can create a custom group called Machine Specification Approvers, to use for all machine pre-approvals. You can also create custom groups to map to your business groups so that you can manage all groups in one place. In those cases, you do not need to assign roles.

Prerequisites

Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Users & Groups > Custom Groups**.

2 Click **New**.

3 Enter a group name in the **Name** text box.

   Custom group names cannot contain the combination of a semicolon (;) followed by an equal sign (=).

4 (Optional) Enter a description in the **Description** text box.

5 Select one or more roles from the Add Roles to this Group list.

   The Authorities Granted by Selected Roles list indicates the specific authorities you are granting.

6 Click **Next**.

7 Add users and groups to create your custom group.

   a Enter a user or group name in the **Search** box and press Enter.

      Do not use an at sign (@), backslash (\), or slash (/) in a name. You can optimize your search by typing the entire user or group name in the form user@domain.

   b Select the user or group to add to your custom group.

8 Click **Finish**.

Results

Users who are currently logged in to the vRealize Automation must log out and log back in to the vRealize Automation before they can navigate to the pages to which they have been granted access.

Add Just-in-Time Users with Custom Groups and Rules

You can add vRealize Automation users to a deployment without access to Active Directory using just-in-time user provisioning. To invoke just-in-time provisioning for first time users, you must create rules to populate the applicable custom group.

On initial log on, just-in-time users are assigned group membership dynamically based on rules that you create on the Advanced Group Membership wizard page. After initial log on, you can assign group membership in the usual manner. This second page of this wizard contains four selection boxes for creating rules based on a variety of criteria that define just-in-time users.

For example, in the first rule selection box, you can select Domain as a criteria, and then select Matches in the second box. Then, in the third rule box, you can enter a domain. These selections create a rule that establishes just-in-time membership based users that are associated with the specified domain. The third selection box is a free form entry box, and you can enter any information that logically relates to the selections in the first two selection boxes.

Note When configuring Just-In-Time users, the `NameId` format mapping specifies an attribute used to uniquely identify an user. This attribute used as a `NameId` should be unique for the user, and the attribute itself should be provided as part of the SAML claim. Changing the `NameId` attribute or the value of the `NameId` will result in an error during a login attempt. For example, if you map the `NameId` to the user's SAMAccountName by using the `urn:oasis:names:tc:SAML:2.0:nameid–format:transient` NameId format, then you should also provide the SAMAccountName separately. The userName and the value of SAMAccountName should never change.

vRealize Automation supports wildcard matching for configuring just in time users. See Using Wildcard based matching for Just in Time Users for more information about enabling and using wildcard matching.

Note You can create multiple rules to populate just-in time users based on a variety of criteria. If you create multiple rules, you can use the **Match** rule selection box, located above the main rule boxes, to indicate whether vRealize Automation should match any or all of the rules when populating just-in-time users.

Procedure

1 Select **Administration > Users & Groups > Custom Groups** and locate an existing group, for example a group that is appropriate for the just-in-time users.

  See Create a Custom Group for more information.

  Click in the group row but not on the name of the group.

2 Click **Advanced Membership**.

  You can add individual users on the Add Users to Group page if desired.

3 Click **Next** to view the Group Rules page.

4 Use the match and rule selection boxes to create one or more rules as appropriate for your user configuration.

  In the three main rule selection boxes, located below the **Match** rule selection box, click the Down arrows and enter information to activate drop down menus that enable you to create the desired rule. Remember that you can use the * and \ characters as described above.

5 Click **Next**.

6 If you want to exclude users from the group, search for and add those users on the Exclude Users from Group page.

7 Click **Next**.

8 Review the group configuration on the Review page, and then click **Save** to save and implement your rules and configuration.

Results

Just-in-time users are added based on the rules that you created.

Using Wildcard based matching for Just in Time Users

vRealize Automation supports wildcard based matching rules for configuring just in time users.

Enable Wildcard based matching

Wildcard based matching is not enabled by default. To enable wildcard based matching you must run the appropriate REST API command as follows.

```
PUT:- https://{{VRA_HOSTNAME}}/SAAS/t/VSPHERE.LOCAL/jersey/manager/api/system/config/
isDynamicGroupWildcardEnabled
Content-Type: application/vnd.vmware.horizon.manager.systemconfigparameter+json
Accept: application/vnd.vmware.horizon.manager.systemconfigparameter+json
Authorization: HZN <token> (edited)
{
   "name": "isDynamicGroupWildcardEnabled",
   "values": {
       "values": [
           "true"
       ]
```

The HZN token to be supplied to the API that enables wildcard configuration must be for the administrator user in the vsphere.local tenant.

Mapping Attributes in the SAML Assertion to vRealize Automation User Attributes

The attribute name in the SAML assertion must completely match the attribute name defined on the vRealize Automation User Attributes page. The SAML attribute containing the user’s first name should be named “firstName”, and the last name should be named “lastName”, etc. If the identity provider sends additional user attributes that not defined on User Attributes page, then the administrator must add those attributes to the page. For example, if the identity provider sends user group membership information in the SAML attribute called “groups” or “memberof”, you must add vRealize Automation User Attributes “groups” or “memberof”. Make sure to use exact capitalization for the attribute names.

Note To positively identify a string such as Group_Name in the multi-value attribute defining user group membership, build a wildcard as *Group_Name*.

For Match and Doesn't Match conditions, you can use an * as a wildcard to include the character pattern match in the rule. For example, entering `<userinput>*Smi*</userinput>` selects Smith, Smiley, Smirnoff and other similar variants, including those with smi in the middle of a name. If you want to find all exact matches to a pattern, add a backslash (\) before the * when entering the pattern. For example, `<userinput>*Adam\* </userinput>` finds all names that match exactly to the pattern Adam*. You can use * anywhere in the phrase, followed and preceded by any character, including * &amp; *.

Create a Business Group

Business groups are used to associate a set of services and resources to a set of users. These groups often correspond to a line of business, department, or other organizational unit. You create a business group so that you can configure reservations and entitle users to provision service catalog items for the business group members.

To add multiple users to a business group role, you can add multiple individual users, or you can add multiple users at the same time by adding an identity store group or a custom group to a role. For example, you can create a custom group Sales Support Team and add that group to the support role. You can also use existing identity store user groups. The users and groups you choose must be valid in the identity store.

To support vCloud Director integration, the same business group members in the vRealize Automation business group must also be members of the vCloud Director organization.

After a tenant administrator creates the business group, the business group manager has permission to modify the manager email address and the members. The tenant administrator can modify all the options.

This procedure assumes that IaaS is installed and configured.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.
- If you want to add machines created by business group members to a particular Active Directory organizational unit, configure the Active Directory policy. See Create an Active Directory Policy. You can and apply the policy when you create the business group, or add it later.
- If you want to provide a default machine prefix for the group that is prepended to provisioned machine names, request a prefix from a fabric administrator. See Configure Machine Prefixes. Machine prefixes are not applicable to XaaS requests.

Procedure

1 Select **Administration > Users and Groups > Business Groups**.

2 Click the **New** icon (➕).

3 Configure the business group details.

Option	Description
Name	Enter the name for the business group.
Description	Enter the description.
Send capacity alert emails to	Enter one or more email addresses of users that must receive capacity alert notifications. Email alias addresses are not supported, each email addresses must be for a specific user. Separate multiple entries with a comma. For example, `JoeAdmin@mycompany.com,WeiMgr@mycompany.com`.
Active Directory Policy	Select the default Active Directory policy for the business group.

4 Add custom properties.

5 Click **Next** to move to the Members page.

6 Enter a user name or custom user group name and press Enter.

You can add one or more individuals or custom user groups to the business group. You can specify the users now or you can create empty business groups to populate later.

Option	Description
Group manager role	Can create entitlements and assign approval policies for the group.
Support role	Can request and manage service catalog items on behalf of the other members of the business group.
Shared access role	Can use and run actions on the resources that other business group members deploy.
User role	Can request service catalog items to which they are entitled.

7 Click **Next** to move to the Infrastructure page.

8 Configure default infrastructure options.

Option	Description
Default machine prefix	Select a preconfigured machine prefix for the business group. This prefix is used by machine blueprints. If the blueprint uses the default prefix and you do not provide it here, a machine prefix is created based on the business group name. The best practice is to provide a default prefix. You can still configure blueprints with specific prefixes or allow service catalog users to override it when they request a blueprint. XaaS blueprints do not use default machine prefixes. If you configure a prefix here and entitle an XaaS blueprint to this business group, it does not affect the provisioning of an XaaS machine.
Active Directory container	Enter an Active Directory container. This option applies only to WIM provisioning. Other provisioning methods require extra configuration to join provisioned machines to an AD container.

9 Click **Finish**.

Results

Tenant administrators can allocate resources to your business group by creating a reservation. Business group managers can create entitlements for members of the business group.

What to do next

- Create a reservation for your business group based on where the business group provisions machines. See Choosing a Reservation Scenario.
- If the catalog items are published and the services exist, you can create an entitlement for the business group members. See Entitle Users to Services, Catalog Items, and Actions.

Troubleshooting Slow Performance When Displaying Group Members

The business group or custom group members are slow to display when viewing a group's details.

Problem

When you view user information in environments with a large number of users, the user names are slow to load in the user interface.

Cause

The extended time required to load the names occurs in environments with a large Active Directory environment.

Solution

◆ To reduce the retrieval workload, use Active Directory groups or custom groups whenever possible rather than adding hundreds of individual members by name.

Troubleshooting Unexpected Entries for Filtering

The business group list used for making filter selections shows unexpected or duplicate entries.

Problem

You made changes to business groups under **Administration > Users and Groups > Business Groups**. On the Deployments page, when trying to filter the deployments by business group, the list of available business groups to filter by doesn't reflect your changes or shows unexpected results, such as duplicate business groups.

Cause

The system polls for changes only once every 30 minutes.

Solution

Wait up to 30 minutes, and refresh the business group filter selection list by refreshing the browser.

Create Additional Tenants

As a system administrator, you can create additional vRealize Automation tenants so that users can access the appropriate applications and resources that they need to complete their work assignments.

A tenant is a group of users with specific privileges who work within a software instance. Typically, a default vRealize Automation tenant is created during system installation and initial configuration. After that, administrators can create additional tenants so that users can log in and complete their work assignments. Administrators can create as many tenants as needed for system operation. When creating tenants, administrators must specify basic configuration such as name, login URL, local users, and administrators. After configuring basic tenant information, the tenant administrator must log in and set up an appropriate Active Directory connection using the Directories Management functionality on the Administrative tab of the vRealize Automation console. In addition, tenant administrators can apply custom branding to tenants.

Prerequisites

Log in to the vRealize Automation console as a **system administrator**.

Procedure

1 (Optional) Specify Tenant Information

   The first step to configuring a tenant is to name the new tenant and add it to vRealize Automation and create the tenant-specific access URL.

2 (Optional) Configure Local Users

   The vRealize Automation system administrator must configure local users for each applicable tenant.

3 (Optional) Appoint Administrators

   You can appoint one or more tenant administrators and IaaS administrators from the identity stores you configured for a tenant.

Specify Tenant Information

The first step to configuring a tenant is to name the new tenant and add it to vRealize Automation and create the tenant-specific access URL.

Prerequisites

Log in to the vRealize Automation console as a **system administrator**.

Procedure

1 Select **Administration > Tenants**.

2. Click the **New** icon (✚).
3. Enter a name in the **Name** text box.
4. (Optional) Enter a description in the **Description** text box.
5. Enter a unique identifier for the tenant in the **URL Name** text box.

   This URL token is used to append a tenant-specific identifier to the vRealize Automation console URL.

   For example, enter `mytenant` to create the URL https://*vrealize-appliance-hostname.domain.name*/vcac/org/*mytenant*.

   Note The tenant URL must use lowercase characters only in vRealize Automation 7.0 and 7.1.

6. (Optional) Enter an email address in the **Contact Email** text box.
7. Click **Submit and Next**.

Configure Local Users

The vRealize Automation system administrator must configure local users for each applicable tenant.

After an administrator creates the general information for a tenant, the Local users tab becomes active, and the administrator can designate users who can access the tenant. When tenant configuration is complete, local tenant users can log in to their respective tenants to complete work assignments.

Note After you add a user, you cannot change its configuration. If you need to change anything about the user configuration, you must delete the user and recreate it.

Procedure

1. Click the **Add** button on the Local users tab.
2. Enter the users first and last names into the **First name** and **Last name** fields on the User Details dialog.
3. Enter the user email address into the **Email** field.
4. Enter the user ID and password for the user in the **User name** and **Password** fields.
5. Click the **Add** button.
6. Repeat these steps as applicable for all local users of the tenant.

Results

The specified local users are created for the tenant.

Appoint Administrators

You can appoint one or more tenant administrators and IaaS administrators from the identity stores you configured for a tenant.

Tenant administrators are responsible for configuring tenant-specific branding, as well as managing identity stores, users, groups, entitlements, and shared blueprints within the context of their tenant. IaaS Administrators are responsible for configuring infrastructure source endpoints in IaaS, appointing fabric administrators, and monitoring IaaS logs.

Prerequisites

- Before you appoint IaaS administrators, you must install IaaS. For information about installing IaaS as part of a distributed deployment, see Install the IaaS Components in a Distributed Configuration. For information about installing IaaS as part of a minimal deployment, see Installing IaaS Components.

Procedure

1 Enter the name of a user or group in the **Tenant Administrators** search box and press Enter.

  For faster results, enter the entire user or group name, for example myAdmins@mycompany.domain. Repeat this step to appoint additional tenant administrators.

2 If you have installed IaaS, enter the name of a user or group in the **IaaS Administrators** search box and press Enter.

  For faster results, enter the entire user or group name, for example IaaSAdmins@mycompany.domain. Repeat this step to appoint additional infrastructure administrators.

3 Click **Add**.

Delete a Tenant

A system administrator can delete any unwanted tenants from vRealize Automation.

If you delete a tenant, that tenant will be removed from the vRealize Automation interface immediately, but it may take several hours for the tenant to be completely removed from your deployment. If you delete a tenant and want to create another tenant with the same URL, allow several hours for complete deletion before creating the new tenant.

Prerequisites

Log in to the vRealize Automation console as a **system administrator**.

Procedure

1 Select **Administration > Tenants**.

2 Select the tenant that you want to delete.

  Do not click the actual name to select the tenant. Doing so will open the tenant for editing.

3 Click **Delete**.

Results

The tenant is deleted from your vRealize Automation deployment.

Configuring Security Settings for Multi-tenancy

You can control the availability of NSX security objects across tenants in a multi-tenant environment.

When you create an NSX security object, its default availability can be either global, meaning available in all tenants for which the associated endpoint has a reservation, or hidden to all users except the administrator.

The availability of security objects across tenants depends on whether the associated endpoint has a reservation or reservation policy in the tenant.

The means by which you control the availability of new security objects across tenants and the behavior you see in existing security objects, relative to cross-tenancy, after upgrading to this vRealize Automation release are summarized in related topic Controlling Tenant Access for Security Objects in vRealize Automation.

Configuring Custom Branding

vRealize Automation enables you to apply custom branding to tenant login and application pages.

Custom branding can include text and background colors, business logos, company name, privacy policies, copyright statements and other relevant information that you want to appear on tenant login or application pages.

Custom Branding for Tenant Login Page

Use the Login Screen Branding page to apply custom branding to your vRealize Automation tenant login pages.

You can use default vRealize Automation branding on your tenant login pages, or you can configure custom branding using the Login Screen Branding page. Note that custom branding applies in the same manner to all of your tenant applications.

This page enables you to configure branding on all tenant login pages.

The Login Screen Branding page displays the currently implemented tenant login branding in the Preview pane.

Note After saving new tenant login page branding, there may be a delay of up to five minutes before it becomes visible on all login pages.

Prerequisites

To use a custom logo or other image with your branding, you must have the appropriate files available.

Procedure

1 Log in to vRealize Automation as a system or tenant administrator.

2 Click the **Administration** tab.

3 Select **Branding > Login Screen Branding**

4 To add a logo image, click **Upload** beneath the Logo field, then navigate to the appropriate folder and select a logo image file.

5 To add an additional image, click **Upload** beneath the Image (optional) field, then navigate to the appropriate folder and select an additional image file.

6 To customize the background colors, enter the appropriate hex codes in the **Background color**, **Masthead color**, **Login button background color** and **Login button foreground color** fields.

  Search the internet for a list of hex color codes if needed.

7 Click **Save** to apply your settings.

Results

Tenant users see the custom branding on their login pages.

Custom Branding for Tenant Applications

Use the Application Branding page to apply custom branding to vRealize Automation tenant applications.

You can use default vRealize Automation branding on your user applications, or you can configure custom branding using the Application Branding page. This page enables you to configure branding on the header and footer of application pages. Note that custom branding applies in the same manner to all of your user applications.

The Application Branding page displays the currently implemented header or footer branding at the bottom of the page.

Prerequisites

If you want to use a custom logo with your branding, you must have the logo image file available.

Procedure

1 Log in to vRealize Automation as a system or tenant administrator.

2 Click the **Administration** tab.

3 Select **Branding > Application Branding**

4. Click the **Header** tab if it is not already active.
5. If you want to use the default vRealize Automation branding, click the **Use Default** check box.
6. To implement custom branding, make the appropriate selections in the fields on the **Header** and **Footer** tabs.

   a. Click the **Browse** button in the **Header Logo** field, then navigate to the appropriate folder and select an logo image file.

   b. Type the appropriate company name in the **Company name** field.

      The specified name appears when a user mouses over the logo.

   c. Type the appropriate name into the **Product name** field.

      The name you enter here appears in the application header adjacent to the logo.

   d. Enter the appropriate hex color code for the application perimeter background color in the **Background hex color** field.

      Search the internet for a list of hex color codes if needed.

   e. Enter the appropriate hex code for the text color in the **Text hex color** field.

      Search the internet for a list of hex text color codes if needed.

   f. Click **Next** to activate the Footer tab.

   g. Type the desired statement into the **Copyright notice** field.

   h. Type the link to you company privacy policy statement in the **Privacy policy link** field.

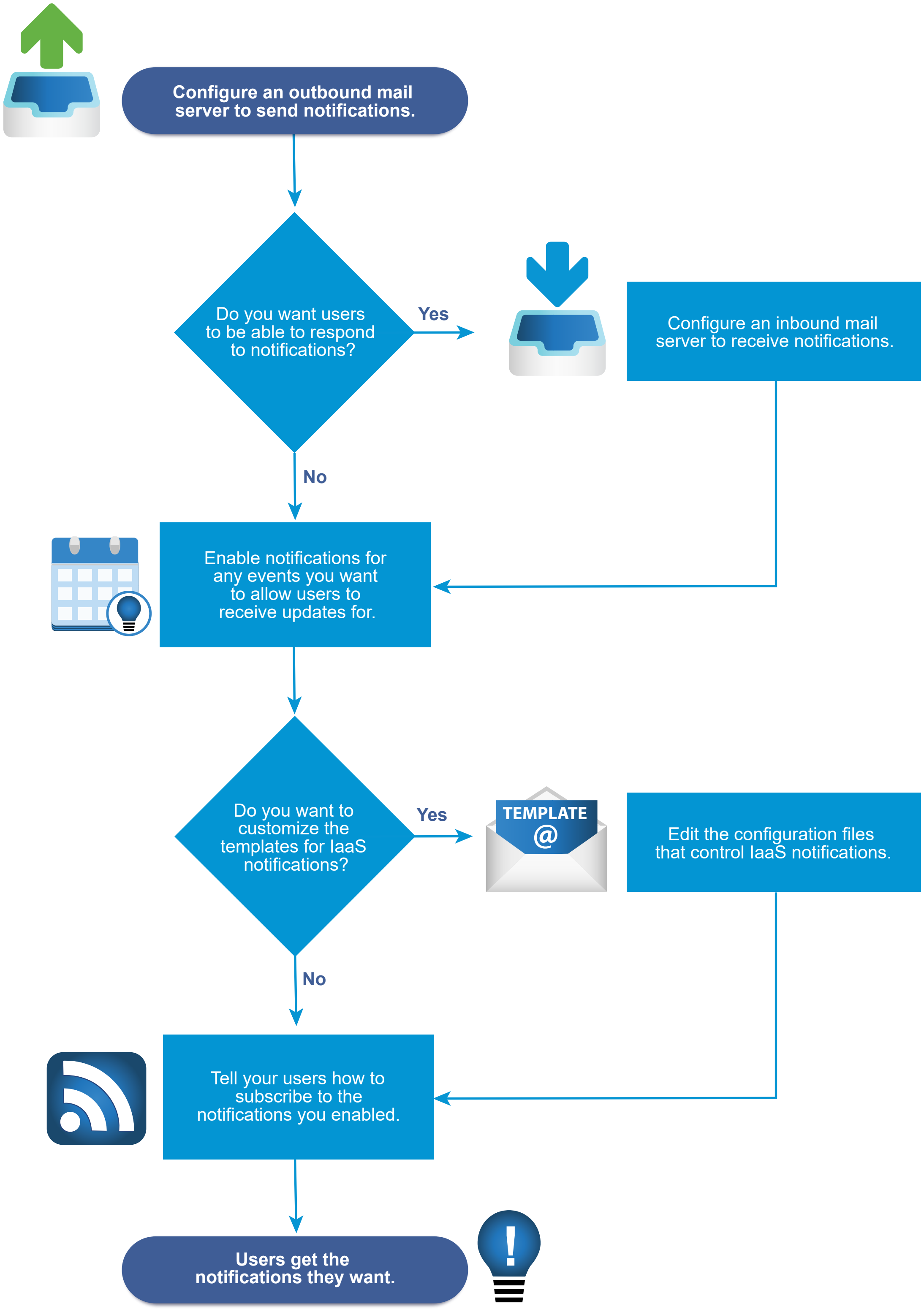
   i. Type the desired company contact information in the **Contact link** field.
7. Click **Update** to implement your branding configuration.

Results

Tenant users see the custom branding on their application pages.

Checklist for Configuring Notifications

You can configure vRealize Automation to send users notifications when specific events occur. Users can choose which notifications to subscribe to, but they can only select from events you enable as notification triggers.


Configure an outbound mail server to send notifications.
Do you want users to be able to respond to notifications?
Yes
Configure an inbound mail server to receive notifications.
No
Enable notifications for any events you want to allow users to receive updates for.
Do you want to customize the templates for IaaS notifications?
Yes
TEMPLATE
@
Edit the configuration files that control IaaS notifications.
No
Tell your users how to subscribe to the notifications you enabled.
Users get the notifications they want.

The Configuring Notifications Checklist provides a high-level overview of the sequence of steps required to configure notifications and provides links to decision points or detailed instructions for each step.

Table 4-10. Checklist for Configuring Notifications

Task	Required Role	Details
❑ Configure an outbound email server to send notifications.	■ System administrators configure default global servers. ■ Tenant administrators configure servers for their tenants.	To configure a server for your tenant for the first time, see Add a Tenant-Specific Outbound Email Server. If you need to override a default global server, see Override a System Default Outbound Email Server. To configure global default servers for all tenants, see Create a Global Outbound Email Server.
❑ (Optional) Configure an inbound email server so that users can complete tasks by responding to notifications.	■ System administrators configure default global servers. ■ Tenant administrators configure servers for their tenants.	To configure a server for your tenant for the first time, see Add a Tenant-Specific Inbound Email Server. If you need to override a default global server, see Override a System Default Inbound Email Server. To configure a global default server for all tenants, see Create a Global Inbound Email Server.
❑ (Optional) Specify when to send an email notification prior to a machine expiration date.	System administrator	See Customize the Date for Email Notification for Machine Expiration.
❑ Select the vRealize Automation events to trigger user notifications. Users can only subscribe to notifications for events you enable as notification triggers.	Tenant administrator	See Configure Notifications.
❑ (Optional) Configure the templates for notifications sent to machine owners concerning events that involve their machines, such as lease expiration.	Anyone with access to the directory `\Templates` under the vRealize Automation server install directory (typically *%SystemDrive%*`\Program Files x86\VMware\vCAC\Server`) can configure the templates for these email notifications.	See Configuring Templates for Automatic IaaS Emails.
❑ You users are automatically subscribed to the configured notifications. If necessary, you can provide your users with instructions about how to subscribe to the notifications that you enabled. They can choose to subscribe to only the notifications that are relevant to their roles.	All users	See Subscribe to Notifications.

Configuring Global Email Servers for Notifications

Tenant administrators can add email servers as part of configuring notifications for their own tenants. As a system administrator, you can set up global inbound and outbound email servers that appear to all tenants as the system defaults. If tenant administrators do not override these settings before enabling notifications, vRealize Automation uses the globally configured email servers.

Create a Global Inbound Email Server

System administrators create a global inbound email server to handle inbound email notifications, such as approval responses. You can create only one inbound server, which appears as the default for all tenants. If tenant administrators do not override these settings before enabling notifications, vRealize Automation uses the globally configured email server.

Prerequisites

Log in to the vRealize Automation console as a **system administrator**.

Procedure

1 Select **Administration > Email Servers**.

2 Click the **Add** icon (✚).

3 Select **Email – Inbound**.

4 Click **OK**.

5 Enter a name in the **Name** text box.

6 (Optional) Enter a description in the **Description** text box.

7 (Optional) Select the **SSL** check box to use SSL for security.

8 Choose a server protocol.

9 Type the name of the server in the **Server Name** text box.

10 Type the server port number in the **Server Port** text box.

11 Type the folder name for emails in the **Folder Name** text box.

This option is required only if you choose IMAP server protocol.

12 Enter a user name in the **User Name** text box.

13 Enter a password in the **Password** text box.

14 Type the email address that vRealize Automation users can reply to in the **Email Address** text box.

15 (Optional) Select **Delete From Server** to delete from the server all processed emails that are retrieved by the notification service.

16 Choose whether vRealize Automation can accept self-signed certificates from the email server.

17. Click **Test Connection**.
18. Click **Add**.

Create a Global Outbound Email Server

System administrators create a global outbound email server to handle outbound email notifications. You can create only one outbound server, which appears as the default for all tenants. If tenant administrators do not override these settings before enabling notifications, vRealize Automation uses the globally configured email server.

Prerequisites

Log in to the vRealize Automation console as a **system administrator**.

Procedure

1. Select **Administration > Email Servers**.
2. Click the **Add** icon (✚).
3. Select **Email – Outbound**.
4. Click **OK**.
5. Enter a name in the **Name** text box.
6. (Optional) Enter a description in the **Description** text box.
7. Type the name of the server in the **Server Name** text box.
8. Choose an encryption method.
   - Click **Use SSL**.
   - Click **Use TLS**.
   - Click **None** to send unencrypted communications.
9. Type the server port number in the **Server Port** text box.
10. (Optional) Select the **Required** check box if the server requires authentication.
    a. Type a user name in the **User Name** text box.
    b. Type a password in the **Password** text box.
11. Type the email address that vRealize Automation emails should appear to originate from in the **Sender Address** text box.

    This email address corresponds to the user name and password you supplied.
12. Choose whether vRealize Automation can accept self-signed certificates from the email server.
13. Click **Test Connection**.
14. Click **Add**.

Add a Tenant-Specific Outbound Email Server

Tenant administrators can add an outbound email server to send notifications for completing work items, such as approvals.

Each tenant can have only one outbound email server. If your system administrator has already configured a global outbound email server, see Override a System Default Outbound Email Server.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.
- If the email server requires authentication, the specified user must be in an identity store and the business group.

Procedure

1. Select **Administration > Notifications > Email Servers**.
2. Click the **Add** icon (✚).
3. Select **Email – Outbound**.
4. Click **OK**.
5. Enter a name in the **Name** text box.
6. (Optional) Enter a description in the **Description** text box.
7. Type the name of the server in the **Server Name** text box.
8. Choose an encryption method.
   - Click **Use SSL**.
   - Click **Use TLS**.
   - Click **None** to send unencrypted communications.
9. Type the server port number in the **Server Port** text box.
10. (Optional) Select the **Required** check box if the server requires authentication.

    a. Type a user name in the **User Name** text box.

    b. Type a password in the **Password** text box.
11. Type the email address that vRealize Automation emails should appear to originate from in the **Sender Address** text box.

    This email address corresponds to the user name and password you supplied.
12. Choose whether vRealize Automation can accept self-signed certificates from the email server.

    This option is available only if you enabled encryption.
    - Click **Yes** to accept self-signed certificates.

   - Click **No** to reject self-signed certificates.

13. Click **Test Connection**.

14. Click **Add**.

Add a Tenant-Specific Inbound Email Server

Tenant administrators can add an inbound email server so that users can respond to notifications for completing work items, such as approvals.

Each tenant can have only one inbound email server. If your system administrator already configured a global inbound email server, see Override a System Default Inbound Email Server.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.
- Verify that the specified user is in an identity store and in the business group.

Procedure

1. Select **Administration** > **Notifications** > **Email Servers**.

2. Click the **Add** icon (✚).

3. Select **Email - Inbound** and click **OK**.

4. Configure the following inbound email server options.

Option	Action
Name	Enter a name for the inbound email server.
Description	Enter a description of the inbound email server.
Security	Select the **Use SSL** check box.
Protocol	Choose a server protocol.
Server Name	Enter the server name.
Server Port	Enter the server port number.

5. Type the folder name for emails in the **Folder Name** text box.

   This option is required only if you choose IMAP server protocol.

6. Enter a user name in the **User Name** text box.

7. Enter a password in the **Password** text box.

8. Type the email address that vRealize Automation users can reply to in the **Email Address** text box.

9. (Optional) Select **Delete From Server** to delete from the server all processed emails that are retrieved by the notification service.

10 Choose whether vRealize Automation can accept self-signed certificates from the email server.

   This option is available only if you enabled encryption.

   - Click **Yes** to accept self-signed certificates.
   - Click **No** to reject self-signed certificates.

11 Click **Test Connection**.

12 Click **Add**.

Override a System Default Outbound Email Server

If the system administrator configured a system default outbound email server, the tenant administrator can override this global setting.

Prerequisites

Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Notifications > Email Servers**.

2 Select the Outbound email server.

3 Click **Override Global**.

4 Enter a name in the **Name** text box.

5 (Optional) Enter a description in the **Description** text box.

6 Type the name of the server in the **Server Name** text box.

7 Choose an encryption method.

   - Click **Use SSL**.
   - Click **Use TLS**.
   - Click **None** to send unencrypted communications.

8 Type the server port number in the **Server Port** text box.

9 (Optional) Select the **Required** check box if the server requires authentication.

   a Type a user name in the **User Name** text box.

   b Type a password in the **Password** text box.

10 Type the email address that vRealize Automation emails should appear to originate from in the **Sender Address** text box.

   This email address corresponds to the user name and password you supplied.

11 Choose whether vRealize Automation can accept self-signed certificates from the email server.

   This option is available only if you enabled encryption.

   - Click **Yes** to accept self-signed certificates.
   - Click **No** to reject self-signed certificates.

12 Click **Test Connection**.

13 Click **Add**.

Override a System Default Inbound Email Server

If the system administrator has configured a system default inbound email server, tenant administrators can override this global setting.

Prerequisites

Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Notifications > Email Servers**.

2 Select the Inbound email server in the Email Servers table.

3 Click **Override Global**.

4 Enter the following inbound email server options.

Option	Action
Name	Enter the name of the inbound email server.
Description	Enter a description of the inbound email server.
Security	Select the **SSL** check box to use SSL for security.
Protocol	Choose a server protocol.
Server Name	Enter the server name.
Server Port	Enter the server port number.

5 Type the folder name for emails in the **Folder Name** text box.

   This option is required only if you choose IMAP server protocol.

6 Enter a user name in the **User Name** text box.

7 Enter a password in the **Password** text box.

8 Type the email address that vRealize Automation users can reply to in the **Email Address** text box.

9 (Optional) Select **Delete From Server** to delete from the server all processed emails that are retrieved by the notification service.

10 Choose whether vRealize Automation can accept self-signed certificates from the email server.

   This option is available only if you enabled encryption.

   - Click **Yes** to accept self-signed certificates.
   - Click **No** to reject self-signed certificates.

11 Click **Test Connection**.

12 Click **Add**.

Revert to System Default Email Servers

Tenant administrators who override system default servers can revert the settings back to the global settings.

Prerequisites

Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Notifications > Email Servers**.

2 Select the email server to revert.

3 Click **Revert to Global**.

4 Click **Yes**.

Configure Notifications

Each user determines whether to receive notifications, but tenant administrators determine which events trigger notifications.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.
- Verify that a tenant administrator or system administrator configured an outbound email server. See Add a Tenant-Specific Outbound Email Server.

Procedure

1 Select **Administration > Notifications > Scenarios**.

2 Select one or more notifications.

3 Click **Activate**.

Results

Users who subscribe to notifications in their preference settings now receive the notifications.

Customize the Date for Email Notification for Machine Expiration

You can specify when to send an email notification prior to a machine expiration date.

You can change the setting that defines the number of days before a machine's expiration date that vRealize Automation sends an expiration notification email. The email notifies users of a machine's expiration date. By default, the setting is 7 days prior to machine expiration.

Procedure

1 Log in to the vRealize Automation server by using credentials with administrative access.

2 Navigate to and open the `/etc/vcac/setenv-user` file.

3 Add the following line to the file to specify the number of days prior to machine expiration, where 3 in this example specifies 3 days prior to machine expiration.

```
VCAC_OPTS="$VCAC_OPTS -Dlease.enforcement.prearchive.notification.days=3"
```

4 Restart the vCAC services on the virtual appliance by running the following command:

```
service vcac-server restart
```

What to do next

If you are working in a high availability load balancer environment, repeat this procedure for all the virtual appliances in the HA environment.

Configuring Templates for Automatic IaaS Emails

You can configure notification emails to be sent to machine owners about various vRealize Automation events that involve their machines.

The events that trigger notifications can include the expiration or approaching expiration of archive periods and virtual machine leases.

For information about configuring and enabling or disabling vRealize Automation email notifications, see the following blog article and Knowledge Base articles:

- Email Customization in vRealize Automation
- Customizing email templates in vRealize Automation (2088805)
- Examples for customizing email templates in vRealize Automation (2102019)

Subscribe to Notifications

If your administrators have configured notifications, you are automatically subscribed. Notification events can include the successful completion of a catalog request or a required approval.

If you must manually subscribe, you can enable your notifications.

Prerequisites

Log in to vRealize Automation.

Procedure

1 Click **Preferences**.

2 Select the **Enabled** check box for the Email protocol in the Notifications table.

3 Click **Apply**.

4 Click **Close**.

Create a Custom RDP File to Support RDP Connections for Provisioned Machines

System administrators create a custom remote desktop protocol file that IaaS architects use in blueprints to configure RDP settings. You create the RDP file and provide architects with the full pathname for the file so they can include it in blueprints, then a catalog administrator entitles users to the RDP action.

Note If you are using Internet Explorer with Enhanced Security Configuration enabled, you cannot download `.rdp` files.

Prerequisites

Log in to the IaaS Manager Service as an administrator.

Procedure

1 Set your current directory to `<`*`vRA_installation_dir`*`>\Rdp`.

2 Copy the file `Default.rdp` and rename it to `Console.rdp` in the same directory.

3 Open the `Console.rdp` file in an editor.

4 Add RDP settings to the file.

For example, **`connect to console:i:1`**.

5 If you are working in a distributed environment, log in as a user with administrative privileges to the IaaS Host Machine where the Model Manager Website component is installed.

6 Copy the `Console.rdp` file to the directory *`vRA_installation_dir`*`\Server\Website\Rdp`.

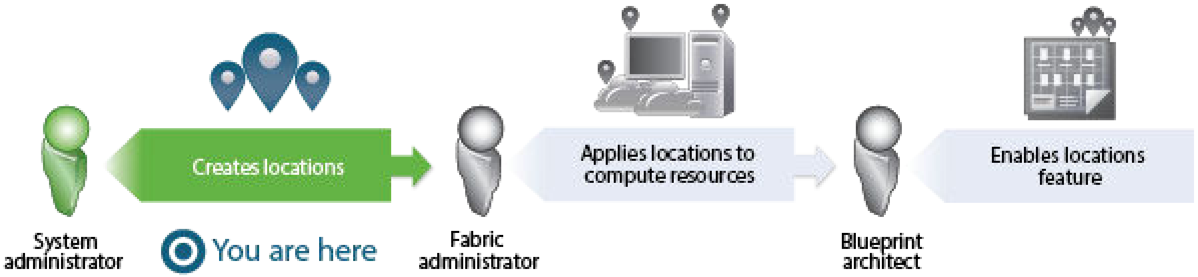
7 Add the `VirtualMachine.Rdp.File` custom property to the blueprint.

Your IaaS architects can add the RDP custom properties to Windows machine blueprints, and then catalog administrators can entitle users to the Connect Using RDP action. See Add RDP Connection Support to Your Windows Machine Blueprints.

Scenario: Add Datacenter Locations for Cross Region Deployments

As a system administrator, you want to define locations for your Boston and London datacenters so your fabric administrators can apply the appropriate locations to compute resources in each datacenter. When your blueprint architects create blueprints, they can enable the locations feature so users can choose to provision machines in Boston or London when they fill out their catalog item request forms.

You have a datacenter in London, and a datacenter in Boston, and you do not want users in Boston provisioning machines on your London infrastructure or vice versa. To ensure that Boston users provision on your Boston infrastructure, and London users provision on your London infrastructure, you want to allow users to select an appropriate location for provisioning when they request machines.



You cannot filter datacenter locations in the xml file based on the tenant or business group. When working in a multi-tenant environment, you can use property definitions to filter based on the tenant or business group. For information about using property definitions, see blog post How to use dynamic property definitions.

Procedure

1 Log in to your IaaS Web Server host using administrator credentials.

  This is the machine on which you installed the IaaS Website component.

2 Edit the file `WebSite\XmlData\DataCenterLocations.xml` in the Windows server install directory (typically *%SystemDrive%*`\Program Files x86\VMware\vCAC\Server`).

3 Edit the CustomDataType section of the file to create Data Name entries for each location.

```
<CustomDataType>
     <Data Name="London" Description="London datacenter" />
        <Data Name="Boston" Description="Boston datacenter" />
</CustomDataType
```

4 Save and close the file.

5 Restart the manager service.

6 If you have more than one IaaS Web Server host, repeat this procedure on each redundant instance.

Results

Your fabric administrator can apply the appropriate location to compute resources located in each datacenter. See Scenario: Apply a Location to a Compute Resource for Cross Region Deployments .

What to do next

You can add the `Vrm.DataCenter.Location` property to a blueprint, or enable the **Display Location on Request** option in the blueprint, to require that the user supply a datacenter location when they request machine provisioning.

Configuring vRealize Orchestrator

vRealize Orchestrator is an automation and management engine that extends vRealize Automation to support XaaS and other extensibility. You can configure and use the vRealize Orchestrator server that is preconfigured in the vRealize Automation appliance, or you can deploy vRealize Orchestrator as an external server instance and associate that external instance with vRealize Automation.

vRealize Orchestrator allows administrators and architects to develop complex automation tasks by using the workflow designer, and then access and run the workflows from vRealize Automation.

vRealize Orchestrator can access and control external technologies and applications by using vRealize Orchestrator plug-ins.

Configuring vRealize Automation to use vRealize Orchestrator makes it possible to publish vRealize Orchestrator workflows in the vRealize Orchestrator service catalog as part of the XaaS blueprint management.

If you want to run workflows to extend the management of IaaS machines, you must configure vRealize Orchestrator as an endpoint.

Configuration Privileges

System and tenant administrators can configure vRealize Automation to use an external or the embedded vRealize Orchestrator server.

In addition, system administrators can also determine the workflow folders that are available to each tenant.

Tenant administrators can configure the vRealize Orchestrator plug-ins as endpoints.

Role	vRealize Orchestrator-Related Configuration Privileges
System administrators	■ Configure the vRealize Orchestrator server for all tenants. ■ Define the default vRealize Orchestrator workflow folders per tenant.
Tenant administrators	■ Configure the vRealize Orchestrator server for their own tenant. ■ Add vRealize Orchestrator plug-ins as endpoints.

Configure the Embedded vRealize Orchestrator Server

The vRealize Automation Appliance includes a preconfigured instance of vRealize Orchestrator.

Prerequisites

Deploy the vRealize Automation Appliance.

Procedure

1 Log in to the vRealize Automation console as a **system administrator** or **tenant administrator**.

2 Select **Administration > VRO Configuration > Server Configuration**.

3 Click **Use the default Orchestrator server**.

Results

Connections to the embedded vRealize Orchestrator server are now configured. The **vCAC** workflows folder and the related utility actions are automatically imported. The **vCAC > ASD** workflows folder contains workflows for configuring endpoints and creating resource mappings.

Log in to the vRealize Orchestrator Control Center

To edit the configuration of the default vRealize Orchestrator instance embedded in vRealize Automation, you must log in to the vRealize Orchestrator Control Center.

The configuration services of the embedded vRealize Orchestrator instance starts automatically.

Note You can verify that the configuration starts automatically by running the `chkconfig vco–configurator` command from the vRealize Orchestrator Appliance command-line console. If the service reports `off`, run the `chkconfig vco–configurator on` command and reboot the appliance.

Procedure

1 Connect to the vRealize Automation URL in a Web browser.

2 Click **vRealize Orchestrator Control Center**.

You are redirected to https://*vra-va-hostname.domain.name_or_load_balancer_address*:8283/vco-controlcenter.

3 Enter the root credentials of your vRealize Automation environment.

Log in to the vRealize Orchestrator Client

To perform general administration tasks or to edit and create workflows in the default vRealize Orchestrator instance, you must log in to the vRealize Orchestrator client.

The vRealize Orchestrator client interface is designed for developers with administrative rights who want to develop workflows, actions, and other custom elements.

Procedure

1. Connect to the vRealize Automation URL in a Web browser.
2. To log in to the HTML5-based vRealize Orchestrator client.

   a. Click **vRealize Orchestrator Client**.

   b. Enter the vRealize Orchestrator client user name and password, and click **Sign in**.

      The credentials are the default tenant administrator user name and password.

3. To log in to the vRealize Orchestrator Legacy Client.

   a. Click **vRealize Orchestrator Legacy Client**.

      The client file is downloaded.

   b. Click the download and following the prompts.

   c. In the **Security Warning** window, select an option to handle the certificate warning.

      The vRealize Orchestrator client communicates with the vRealize Orchestrator server by using an SSL certificate. A trusted CA does not sign the certificate during installation. You receive a security warning each time you connect to the vRealize Orchestrator server.

Option	Description
Continue	Continue using the current SSL certificate. The warning message appears again when you reconnect to the same vRealize Orchestrator server, or when you try to synchronize a workflow with a remote vRealize Orchestrator server.
Cancel	Close the window and stop the login process.

   d. Click **Run**.

   e. On the vRealize Orchestrator login page, enter the IP or the domain name of the vRealize Automation appliance in the **Host name** text box, and **443** as the default port number.

      For example, enter *vrealize_automation_appliance_ip*:443.

   f. Enter the vRealize Orchestrator client user name and password, and click **Login**.

      The credentials are the default tenant administrator user name and password.

What to do next

Use the vRealize Orchestrator client to develop and run workflows, and export your content to other vRealize Orchestrator environments by using packages. See *Using the VMware vRealize Orchestrator Client* and *Developing with VMware vRealize Orchestrator*.

Configure an External vRealize Orchestrator Server

You can set up vRealize Automation to use an external vRealize Orchestrator server.

System administrators can configure the default vRealize Orchestrator server globally for all tenants. Tenant administrators can configure the vRealize Orchestrator server only for their tenants.

Connections to external vRealize Orchestrator server instances require the user account to have view and run permissions in vRealize Orchestrator.

- Single Sign-On authentication. The user information is passed to vRealize Orchestrator with the XaaS request and the user is granted view and run permissions for the requested workflow.
- Basic authentication. The provided user account must be a member of a vRealize Orchestrator group with view and run permissions or the member of the vcoadmins group.

Prerequisites

- Install and configure an external vRealize Orchestrator Appliance. See *Installing and Configuring vRealize Orchestrator* in the vRealize Orchestrator product documentation.
- Log in to the vRealize Automation console as a **system administrator** or **tenant administrator**.

Procedure

1. Select **Administration > vRO Configuration > Server Configuration**.
2. Click **Use an external Orchestrator server**.
3. Enter a name and, optionally, a description.
4. Enter the IP or the DNS name of the machine on which the vRealize Orchestrator server runs in the **Host** text box.

   Note If the external vRealize Orchestrator is configured to work in cluster mode, enter the IP address or host name of the load balancer virtual server that distributes the client requests across the vRealize Orchestrator servers in the cluster.

5. Enter the port number to communicate with the external vRealize Orchestrator server in the **Port** text box.

   8281 is the default port for vRealize Orchestrator.

6 Select the authentication type.

Option	Description
Single Sign-On	Connects to the vRealize Orchestrator server by using vCenter Single Sign-On. This option is applicable only if you configured the vRealize Orchestrator and vRealize Automation to use a common vCenter Single Sign-On instance.
Basic	Connects to the vRealize Orchestrator server with the user name and password that you enter in the **User name** and **Password** text boxes. The account that you provide must be a member of the vRealize Orchestrator vcoadmins group or a member of a group with view and run permissions.

7 Click **Test Connection**.

8 Click **OK**.

9 Import the `xaas.package` package.

   a Log in to the vRealize Automation Appliance as **root**.

   b Locate the `xaas.package` package in the `/usr/lib/vcac/content/o11n/` folder.

   c Import the `xaas.package` package to the external Client.

Results

You configured the connection to the external vRealize Orchestrator server, and imported the **vCAC** workflows folder and the related utility actions. The **vCAC > ASD** workflows folder contains workflows for configuring endpoints and creating resource mappings.

What to do next

Log in to the vRealize Orchestrator Client.

Configuring Resources

You can configure resources such as endpoints, reservations, and network profiles to support vRealize Automation blueprint definition and machine provisioning.

Checklist for Configuring IaaS Resources

IaaS administrators and fabric administrators configure IaaS resources to integrate existing infrastructure with vRealize Automation and to allocate infrastructure resources to vRealize Automation business groups.

You can use the Configuring IaaS Resources Checklist to see a high-level overview of the sequence of steps required to configure IaaS resources.

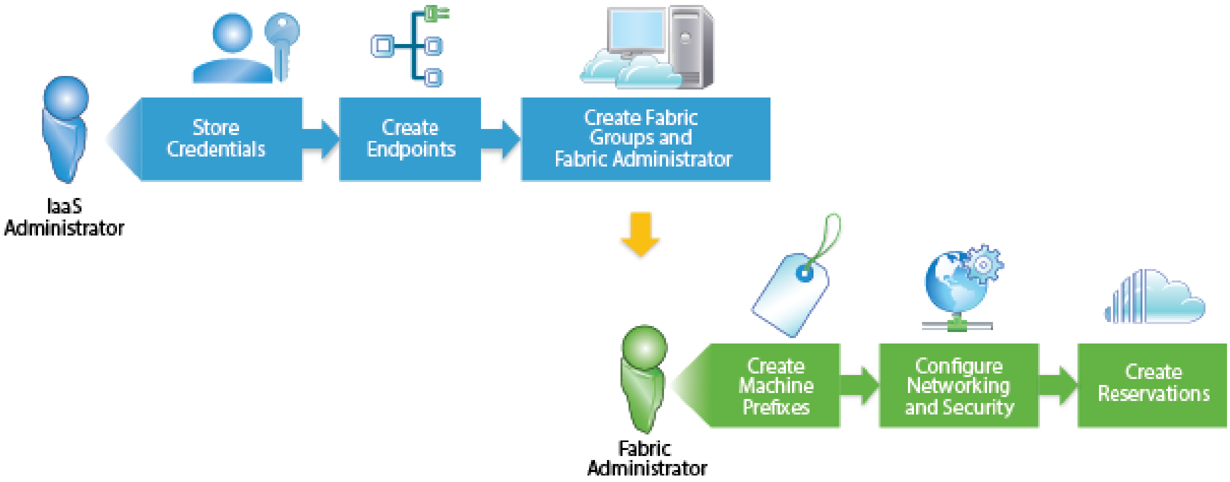


Table 4-11. Checklist for Configuring IaaS Resources

Task	vRealize Automation Role	Details
❑ Create endpoints for your infrastructure to bring resources under vRealize Automation management.	IaaS administrator	Choosing an Endpoint Scenario.
❑ Create a fabric group to organize infrastructure resources into groups and assign one or more administrators to manage those resources as your vRealize Automation fabric administrators.	IaaS administrator	Create a Fabric Group.
❑ Configure machine prefixes used to create names for machines provisioned through vRealize Automation.	Fabric administrator	Configure Machine Prefixes.
❑ (Optional) Create network profiles to configure network settings for provisioned machines.	Fabric administrator	Creating a Network Profile in vRealize Automation.
❑ Allocate infrastructure resources to business groups by creating reservations and, optionally, reservation and storage reservation profiles.	■ IaaS administrator if also configured as a Fabric administrator ■ Fabric administrator	Configuring Reservations and Reservation Policies.

Configuring Endpoints

You create and configure the endpoints that allow vRealize Automation to communicate with your infrastructure.

Endpoints definitions are categorized based on type:

- Cloud

  The cloud category contains the vCloud Air, vCloud Director, Amazon EC2, and OpenStack endpoint types

- IPAM

This category is only visible if you have registered a third-party IPAM endpoint type such as Infoblox IPAM in a vRealize Orchestrator workflow.

- Management

  This category contains the vRealize Operations Manager endpoint only.

- Network and Security

  This category contains the Proxy and NSX endpoint types.

  A Proxy endpoint can be associated to an Amazon, vCloud Air, or vCloud Director endpoint.

  An NSX endpoint can be associated to a vSphere endpoint.

- Orchestration

  This category contains the vRealize Orchestrator endpoint only.

- Storage

  This category contains the NetApp ONTAP endpoint.

- Virtual

  The virtual category contains the vSphere, Hyper-V (SCVMM), and KVM (RHEV) endpoint types.

You can configure additional endpoint types in vRealize Orchestrator and use them with supported endpoint types in vRealize Automation. You can also import, and export, endpoints programmatically.

For information about working with endpoints after upgrade or migration, see Considerations When Working With Upgraded or Migrated Endpoints.

Choosing an Endpoint Scenario

Choose an endpoint scenario based on the target endpoint type.

For information about available endpoint settings, see Endpoint Settings Reference.

Table 4-12. Choosing an Endpoint Scenario

Endpoint	More Information
vSphere	See Create a vSphere Endpoint in vRealize Automation and Associate to NSX .
NSX	See Create an NSX for vSphere Endpoint and Associate to a vSphere Endpoint in vRealize Automation or Create an NSX-T Endpoint and Associate to a vSphere Endpoint in vRealize Automation.
vCloud Air (Subscription or OnDemand)	See Create a vCloud Air Endpoint .
vCloud Director	See Create a vCloud Director Endpoint.
vRealize Orchestrator	See Create a vRealize Orchestrator Endpoint.

Table 4-12. Choosing an Endpoint Scenario (continued)

Endpoint	More Information
vRealize Operations	See Create a vRealize Operations Manager Endpoint.
Third party IPAM provider	See Create a Third-Party IPAM Provider Endpoint.
Microsoft Azure	See Create a Microsoft Azure Endpoint.
Puppet	See Create a Puppet Endpoint.
Amazon	See Create an Amazon Endpoint and Add an Amazon Instance Type.
OpenStack	See Create an OpenStack Endpoint.
Proxy	Create a Proxy Endpoint and Associate to a Cloud Endpoint
Hyper-V (SCVMM)	See Create a Hyper-V (SCVMM) Endpoint.
KVM (RHEV)	See Endpoint Settings Reference.
NetApp ONTAP	See Space-Efficient Storage for Virtual Provisioning and Endpoint Settings Reference.
Hyper-V (Standalone), XenServer or Xen Pool Master	See Create a Hyper-V, XenServer, or Xen Pool Endpoint.
Import endpoints	See Import or Export Endpoints Programmatically.

Endpoint Settings Reference

Use endpoint settings to define location and access credentials for data collection and service catalog deployment.

General Tab

Most vRealize Automation endpoints contain the following options. Settings that are unique to a particular endpoint type are noted.

Table 4-13. General Tab Settings

Setting	Description
Name	Enter the endpoint name.
Description	Enter the endpoint description.

Table 4-13. **General** Tab Settings (continued)

Setting	Description
Address	Enter the endpoint address using the endpoint-specific address format. ■ For a KVM (RHEV) or NetApp ONTAP endpoint, the address must be of one of the following formats: ■ `https://`*`FQDN`* ■ `https://`*`IP_address`* For example: **`https://mycompany-kvmrhev1.mycompany.local`** or **`netapp-1.mycompany.local`**. ■ For an OpenStack endpoint, the address must be of the format `https:// FQDN/powervc/openstack/ service`. For example: **`https://openstack.mycompany.com/powervc/openstack/admin`**. ■ For an OpenStack endpoint, the address must be of one of the following formats: ■ `https://`*`FQDN`*`:500` ■ `https://`*`IP_address`*`:500` ■ For a vSphere endpoint, the address must be of the format `https://`*`host`*`/`*`sdk`* . ■ For an NSX endpoint, the address must be of the format `https://`*`host`*. ■ For a vRealize Orchestrator endpoint, the address must be of the https protocol and include the fully qualified name or IP address of the vRealize Orchestrator server and thevRealize Orchestrator port number, for example `https://vrealize-automation-appliance-hostname:443/vco`. ■ For a vRealize Operations endpoint, the address must be of the format `https://`*`host`*`/`*`suite-api`*.
Integrated credentials	If you use your vSphere integrated credentials, you do not need to enter a user name and password. This setting applies to vSphere endpoints only.
User name	Enter the administrator-level user name that you stored for the endpoint in the endpoint-specific format as suggested in the user interface.
Password	Enter the administrator-level password that you stored for the endpoint.
OpenStack project	Enter an OpenStack tenant name. This setting applies to OpenStack endpoints only.
Organization	If you are an organization administrator, you can enter a vCloud Director organization name. This setting applies tovCloud Director only.
Access key ID	Enter the Amazon AWS key ID. This setting applies to Amazon endpoints only.
Secret access key	Enter your Amazon AWS secret access key. This setting applies to Amazon endpoints only.

Table 4-13. **General** Tab Settings (continued)

Setting	Description
Port	Enter the port value to connect to on the proxy endpoint address. This setting applies to Proxy endpoints only.
Priority	Enter a priority value as an integer greater than or equal to 1. The lower value specifies a higher priority. The priority value is associated to the embedded `VMware.VCenterOrchestrator.Priority` custom property. This setting applies to vRealize Orchestrator endpoints only.

Properties Tab

All endpoint types use a properties tab to capture custom properties or property groups and settings. For examples of custom properties for specific endpoint types, see Custom Properties Grouped by Function.

Association Tab

You can create an association to an NSX endpoint or a Proxy endpoint, depending on the endpoint you are associating from. You can associate a vSphere endpoint with an NSX endpoint to assign NSX settings to the vSphere endpoint. You can also associate a vCloud Air, vCloud Director, or Amazon endpoint with a proxy endpoint to assign proxy settings to the vCloud Air, vCloud Director, or Amazon endpoint.

Test Connection

You can use a test connection action to validate the credentials, host endpoint address, and certificate for a vSphere, NSX, or vRealize Operations Manager endpoint. See Considerations When Using Test Connection.

Create a vSphere Endpoint in vRealize Automation and Associate to NSX

You can create vSphere endpoints in vRealize Automation that communicate with vCenter to discover compute resources, collect data, and provision machines. You can also associate NSX settings to the vSphere endpoint by associating to an NSX for vSphere endpoint, one or more NSX-T endpoints, or both NSX endpoint types.

Associating a vSphere endpoint to NSX for vSphere and NSX-T endpoints allows you to configure NSX for vSphere and NSX-T for different clusters in a single vCenter:

- An IaaS administrator can associate a vSphere endpoint with both an NSX for vSphere endpoint and an NSX-T endpoint.
- A fabric administrator can create an NSX for vSphere or NSX-T reservation depending on the compute resource.
- A blueprint architect can author blueprints that are either NSX for vSphere specific or NSX-T specific. Both types of blueprints can be deployed in the same vCenter environment.

You can create an association between vSphere and NSX endpoints. Associations include:

- One vSphere endpoint associated with a single NSX for vSphere endpoint.
- One vSphere endpoint associated with multiple NSX-T endpoints.

- One NSX-T endpoint associated with multiple vSphere endpoints.
- One NSX for vSphere endpoint associated with a single vSphere endpoint.
- One vSphere endpoint associated with one NSX for vSphere endpoint and with one NSX-T endpoint.

  When a vSphere endpoint is associated to both an NSX for vSphere endpoint and an NSX-T endpoint, the cluster is managed by NSX for vSphere or by NSX-T. The NSX manager is determined by vRealize Automation when endpoints are data-collected and relationship is established. You can see the type of NSX platform that manages a specific cluster by inspecting the **NSX type** column on the **Compute Resources** page.

For information about creating NSX endpoints that are associated with a vSphere endpoint, see Create an NSX for vSphere Endpoint and Associate to a vSphere Endpoint in vRealize Automation or Create an NSX-T Endpoint and Associate to a vSphere Endpoint in vRealize Automation.

For information about validating the endpoint connection and certificate trust, see Considerations When Using Test Connection.

If you upgraded or migrated a vSphere endpoint that was using an NSX manager, a new NSX endpoint is created that contains an association between the source vSphere endpoint and a new NSX endpoint.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- You must install a vSphere proxy agent to manage your vSphere endpoint. The agent name and endpoint name must match. For information about installing the agent, see Installing and Configuring the Proxy Agent for vSphere.
- If you plan to use a vSphere endpoint to deploy VMs from OVF templates, verify that your credentials include the vSphere privilege `VApp.Import` in the vCenter Server that is associated with the endpoint.

  The `VApp.Import` privilege allows you to deploy a vSphere machine by using settings imported from an OVF. Details about this vSphere privilege are available in the vSphere SDK documentation.

  If the OVF is hosted on a Web site, see Create a Proxy Endpoint for OVF Host Web Site.
- Configure the vSphere Agent.
- To configure additional NSX network and security settings for the vSphere endpoint, create an NSX for vSphere or NSX-T endpoint. You can associate to your NSX endpoint when you create or edit the vSphere endpoint.

Procedure

1 Select **Infrastructure > Endpoints > Endpoints**.

2 Select **New > Virtual > vSphere**.

3 Enter a name in the **Name** text box.

   The name must match the endpoint name provided to the vSphere proxy agent during installation. If the name does not match, data collection fails.

4 (Optional) Enter a description in the **Description** text box.

5 Enter the URL for the vCenter Server instance in the **Address** text box.

   The URL must be of the type: `https://`*`hostname/sdk`* or `https://`*`IP_address/sdk`*.

   For example, `https://vsphereA/sdk`.

6 Enter your vSphere administrator-level user name and password or use your vSphere integrated credentials.

   Provide credentials that have permission to modify custom attributes.

   The user name format is *domain\username*.

   To use the vSphere proxy agent's service account to connect to the vCenter Server, select **Use Integrated Credentials**.

   If you use your vSphere integrated credentials, you do not need to enter a user name and password.

7 (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.

8 (Optional) To configure your NSX network and security settings for the endpoint, click **Associations** and associate to an existing NSX for vSphere or NSX-T endpoint.

   You must have at least one NSX endpoint to create an association.

9 (Optional) To validate the credentials, host endpoint address, and certificate trust, click **Test Connection**. The action also checks that the manager service and agent are running so that endpoint can be data-collected. The **OK** action tests for these same conditions.

   The **Test Connection** action returns information about any of the following conditions:

   - Certificate error

     If the certificate is not found, trusted, or has expired, you are prompted to accept a certificate thumb print. If you do not accept the thumb print, you can still save the endpoint but machine provisioning might fail.

   - Agent error

     The associated vSphere agent is not found. The agent must be running for the test to succeed.

   - Host error

     The specified endpoint address is not reachable or the associated manager service is not running. The manager service must be running for the test to succeed.

   - Credentials error

The specified user name and password combination is invalid for the endpoint at the specified address.

- Timeout

  The test action could not complete in the allowed two-minute time period.

If the **Test Connection** action fails, you can still save the endpoint but machine provisioning might fail.

If there is a trusted certificate issue, for example the certificate has expired, you are prompted to accept a certificate thumb print.

10 To save the endpoint, click **OK**.

The **OK** action tests for the same conditions as the **Test Connection** action. If it finds any of the preceding conditions, it returns a message. If it can save, it leaves the error on the screen for you to review.

Results

vRealize Automation collects data from your endpoint and discovers your compute resources.

Note Do not rename vSphere data centers after the initial data collection or provisioning might fail.

For more information, see Viewing Compute Resources and Running Data Collection.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Create an NSX for vSphere Endpoint and Associate to a vSphere Endpoint in vRealize Automation

You can create an NSX for vSphere endpoint and associate it to an existing vSphere endpoint in vRealize Automation.

You can associate an NSX for vSphere endpoint with a vSphere endpoint.

You can create an association between vSphere and NSX endpoints. Associations include:

- One vSphere endpoint associated with a single NSX for vSphere endpoint.
- One vSphere endpoint associated with multiple NSX-T endpoints.
- One NSX-T endpoint associated with multiple vSphere endpoints.
- One NSX for vSphere endpoint associated with a single vSphere endpoint.
- One vSphere endpoint associated with one NSX for vSphere endpoint and with one NSX-T endpoint.

When a vSphere endpoint is associated to both an NSX for vSphere endpoint and an NSX-T endpoint, the cluster is managed by NSX for vSphere or by NSX-T. The NSX manager is determined by vRealize Automation when endpoints are data-collected and relationship is established. You can see the type of NSX platform that manages a specific cluster by inspecting the **NSX type** column on the **Compute Resources** page.

For information about validating the endpoint connection and certificate trust, see Considerations When Using Test Connection.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- You must install a vSphere proxy agent to manage your vSphere endpoint, and you must use the same exact name for your endpoint and agent. For information about installing the agent, see Installing and Configuring the Proxy Agent for vSphere.
- Configure your NSX for vSphere network settings. See Configuring Network and Security Component Settings in vRealize Automation.
- Create a vSphere Endpoint in vRealize Automation and Associate to NSX .

In preparation for using NSX network, security, and load balancing capabilities in vRealize Automation, when using NSX Manager credentials you must use the NSX Manager administrator account.

Procedure

1. Select **Infrastructure > Endpoints > Endpoints**.
2. Select **New > Network and Security > NSX**.
3. Enter a name in the **Name** text box.
4. (Optional) Enter a description in the **Description** text box.
5. Enter the URL for the NSX for vSphere instance in the **Address** text box.

   The URL must be of the type: `https://`*`hostname`* or `https://`*`IP_address`*.

   For example, `https://abx.nsx-manager.local/`.
6. Enter the NSX administrator-level user name and password that are stored for the NSX for vSphere endpoint.
7. (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.
8. To associate the NSX for vSphere network and security settings to an existing vSphere endpoint, click **Associations** and select an existing vSphere endpoint.

   You must create the vSphere endpoint before you can create the association.

   A vSphere endpoint can be associated with only one type of network and security platform - either NSX for vSphere or NSX-T.

You can only associate an NSX for vSphere endpoint to one vSphere endpoint. This association constraint means that you cannot provision a universal on-demand network and attach it to vSphere machines that are provisioned on different vCenters.

When the association is finished, the Description column on the page indicates the association type of NSX for vSphere.

9 (Optional) To validate the credentials, host endpoint address, and certificate trust, click **Test Connection**. The action also checks that the manager service and agent are running so that endpoint can be data-collected. The **OK** action tests for these same conditions.

The **Test Connection** action returns information about any of the following conditions:

- Certificate error

  If the certificate is not found, trusted, or has expired, you are prompted to accept a certificate thumb print. If you do not accept the thumb print, you can still save the endpoint but machine provisioning might fail.

- Agent error

  The associated vSphere agent is not found. The agent must be running for the test to succeed.

- Host error

  The specified endpoint address is not reachable or the associated manager service is not running. The manager service must be running for the test to succeed.

- Credentials error

  The specified user name and password combination is invalid for the endpoint at the specified address.

- Timeout

  The test action could not complete in the allowed two-minute time period.

If the **Test Connection** action fails, you can still save the endpoint but machine provisioning might fail.

If there is a trusted certificate issue, for example the certificate has expired, you are prompted to accept a certificate thumb print.

10 To save the endpoint, click **OK**.

The **OK** action tests for the same conditions as the **Test Connection** action. If it finds any of the preceding conditions, it returns a message. If it can save, it leaves the error on the screen for you to review.

Results

vRealize Automation collects data from your endpoint and discovers your compute resources.

For information about running data collection for existing endpoints after initial data collection, see Viewing Compute Resources and Running Data Collection.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Create an NSX-T Endpoint and Associate to a vSphere Endpoint in vRealize Automation

You can create an NSX-T endpoint and associate it to an existing vSphere endpoint in vRealize Automation.

vRealize Automation uses basic authentication to connect with the NSX-T endpoint.

To facilitate fault tolerance and high availability in deployments, each NSX-T data center endpoint represents a cluster of three NSX managers.

- vRealize Automation can point to one of the NSX managers. With this option, one NSX manager receives the API calls from vRealize Automation.
- vRealize Automation can point to the Virtual IP of the cluster. With this option, one NSX manager assumes control of the VIP. That manager receives the API calls from vRealize Automation. In case of failure, another node in the cluster assumes control of the VIP and receives the API calls from vRealize Automation.

  For more information about VIP configuration, see *Configure a Virtual IP (VIP) Address for a Cluster* in the *NSX-T Data Center Installation Guide* at VMware NSX-T Data Center Documentation.
- vRealize Automation can point to a load balancer VIP to load-balance the calls to the three NSX managers. Using this option, all three NSX managers receive API calls from vRealize Automation.

  You can configure the VIP on a third-party load balancer or on an NSX-T load balancer.

  For large scale environments, consider using this option to split the vRealize Automation API calls among the three NSX managers.

Use this information as you specify the NSX-T endpoint in step 5.

You can associate an NSX-T endpoint with one or more vSphere endpoints.

You can create an association between vSphere and NSX endpoints. Associations include:

- One vSphere endpoint associated with a single NSX for vSphere endpoint.
- One vSphere endpoint associated with multiple NSX-T endpoints.
- One NSX-T endpoint associated with multiple vSphere endpoints.
- One NSX for vSphere endpoint associated with a single vSphere endpoint.
- One vSphere endpoint associated with one NSX for vSphere endpoint and with one NSX-T endpoint.

  When a vSphere endpoint is associated to both an NSX for vSphere endpoint and an NSX-T endpoint, the cluster is managed by NSX for vSphere or by NSX-T. The NSX manager is determined by vRealize Automation when endpoints are data-collected and relationship is established. You can see the type of NSX platform that manages a specific cluster by inspecting the **NSX type** column on the **Compute Resources** page.

When you deploy a blueprint that contains an NSX-T endpoint, the deployment assigns a tag to NSX-T components in the deployment. The tag name and the deployment name match.

For information about validating the endpoint connection and certificate trust, see Considerations When Using Test Connection.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- You must install a vSphere proxy agent to manage your vSphere endpoint, and you must use the same exact name for your endpoint and agent. For information about installing the agent, see Installing and Configuring the Proxy Agent for vSphere.
- Configure your NSX-T network settings. See Configuring Network and Security Component Settings in vRealize Automation.
- Create a vSphere Endpoint in vRealize Automation and Associate to NSX .

In preparation for using NSX network, security, and load balancing capabilities in vRealize Automation, when using NSX Manager credentials you must use the NSX Manager administrator account.

Procedure

1. Select **Infrastructure > Endpoints > Endpoints**.
2. Select **New > Network and Security > NSX-T**.
3. Enter a name in the **Name** text box.
4. (Optional) Enter a description in the **Description** text box.
5. Enter the URL for the NSX-T endpoint manager instance or VIP (see above) in the **Address** text box.

   The URL must be of the type: `https://`*`hostname`* or `https://`*`IP_address`*.

   For example, `https://abx-nsxt3-manager.local`.
6. Enter the NSX administrator-level user name and password that are stored for the NSX-T endpoint.
7. (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.
8. To associate the NSX-T network and security settings to an existing vSphere endpoint, click **Associations** and select an existing vSphere endpoint.

   You must create the vSphere endpoint before you can create the association.

   A vSphere endpoint can be associated with only one type of network and security platform - either NSX for vSphere or NSX-T.

   You can associate an NSX-T endpoint to more than one vSphere endpoint. One NSX-T instance can manage multiple ESX clusters on different vCenters.

When the association is finished, the Description column on the page indicates the association type of NSX-T.

9 (Optional) To validate the credentials, host endpoint address, and certificate trust, click **Test Connection**. The action also checks that the manager service and agent are running so that endpoint can be data-collected. The **OK** action tests for these same conditions.

   The **Test Connection** action returns information about any of the following conditions:

   - Certificate error

     If the certificate is not found, trusted, or has expired, you are prompted to accept a certificate thumb print. If you do not accept the thumb print, you can still save the endpoint but machine provisioning might fail.

   - Agent error

     The associated vSphere agent is not found. The agent must be running for the test to succeed.

   - Host error

     The specified endpoint address is not reachable or the associated manager service is not running. The manager service must be running for the test to succeed.

   - Credentials error

     The specified user name and password combination is invalid for the endpoint at the specified address.

   - Timeout

     The test action could not complete in the allowed two-minute time period.

   If the **Test Connection** action fails, you can still save the endpoint but machine provisioning might fail.

   If there is a trusted certificate issue, for example the certificate has expired, you are prompted to accept a certificate thumb print.

10 To save the endpoint, click **OK**.

   The **OK** action tests for the same conditions as the **Test Connection** action. If it finds any of the preceding conditions, it returns a message. If it can save, it leaves the error on the screen for you to review.

Results

vRealize Automation collects data from your endpoint and discovers your compute resources.

For information about running data collection for existing endpoints after initial data collection, see Viewing Compute Resources and Running Data Collection.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Create a vCloud Air Endpoint

You can create a vCloud Air endpoint for a an OnDemand or subscription service. You can optionally associate proxy settings to the vCloud Director endpoint by associating to a Proxy endpoint.

For information about vCloud Air Management Console, see vCloud Air documentation.

Note Reservations defined for vCloud Air endpoints and vCloud Director endpoints do not support the use of network profiles for provisioning machines.

For vCloud Air endpoints, the Organization name and the vDC name must be identical for a vCloud Air subscription instance.

For information about associating proxy settings to your endpoint, see Create a Proxy Endpoint and Associate to a Cloud Endpoint.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- Verify that you have **Virtual Infrastructure Administrator** authorization for your vCloud Air subscription service or OnDemand account.
- If you want to configure additional security and force connections to pass through a proxy server, create a Proxy endpoint. You can associate to the Proxy endpoint as you create the vCloud Director endpoint. See Create a Proxy Endpoint and Associate to a Cloud Endpoint.

Procedure

1. Select **Infrastructure > Endpoints > Endpoints**.
2. Select **New > Cloud > vCloud Air**.
3. Enter a name and, optionally, a description.
4. Accept the default vCloud Air endpoint address in the **Address** text box or enter a new one.

   The default vCloud Air endpoint address is https://vca.vmware.com, as specified in the `Default URL for vCloud Air endpoint` global property.
5. Enter your administrator-level user name and password.

   The credentials must be those of thevCloud Air subscription service or OnDemand account administrator.

   The user name format is *domain\username*.

   Provide credentials for an organization administrator with rights to connect by using VMware Remote Console.
6. (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.

7 (Optional) To configure additional security and force connections to pass through a proxy server, click **Associations** and associate to an existing Proxy endpoint.

   You must have at least one Proxy endpoint to create an association.

8 Click **OK**.

What to do next

Create a Fabric Group.

Create a vCloud Director Endpoint

You can create a vCloud Director endpoint to manage all of the vCloud Director virtual data centers (vDCs) in your environment, or you can create separate endpoints to manage each vCloud Director organization. You can optionally associate proxy settings to the vCloud Director endpoint by associating to a Proxy endpoint.

For information about Organization vDCs, see vCloud Director documentation.

Do not create a single endpoint and individual organization endpoints for the same vCloud Director instance.

vRealize Automation uses a proxy agent to manage vSphere resources.

Note Reservations defined for vCloud Air endpoints and vCloud Director endpoints do not support the use of network profiles for provisioning machines.

Lease information for vCloud Director machines must be specified in vRealize Automation and not in vCloud Director. If you specify lease information in vCloud Director, that lease information is not recognized or used in vRealize Automation. Enter lease information for vCloud Director machines in your vRealize Automation blueprint, not in vCloud Director.

For information about associating proxy settings to your endpoint, see Create a Proxy Endpoint and Associate to a Cloud Endpoint.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- If you want to configure additional security and force connections to pass through a proxy server, create a Proxy endpoint. You can associate to the Proxy endpoint as you create the vCloud Director endpoint. See Create a Proxy Endpoint and Associate to a Cloud Endpoint.

Procedure

1 Select **Infrastructure > Endpoints > Endpoints**.

2 Select **New > Cloud > vCloud Director**.

3 Enter a name and, optionally, a description.

4 Enter the URL of the vCloud Director server in the **Address** text box.

   The URL must be of the type *FQDN* or *IP_address*.

For example, https://mycompany.com.

5 Enter your administrator-level user name and password.

- To connect to the vCloud Director server and specify the organization for which the user has the administrator role, use organization administrator credentials. With these credentials, the endpoint can access only the associated organization vDCs. You can add endpoints for each additional organization in the vCloud Director instance to integrate with vRealize Automation.
- To allow access to all Organization vDCs in the vCloud Director instance, use system administrator credentials for a vCloud Director and leave the **Organization** text box empty.

6 If you are an organization administrator, you can enter a vCloud Director organization name in the **Organization** text box.

Option	Description
Discover all Organization vCDs	If you have implemented vCloud Director in a private cloud, you can leave the **Organization** text box blank to allow the application to discover all the available Organization vDCs.
Separate endpoints for each Organization vCD	Enter a vCloud Director organization name in the **Organization** text box.

The **Organization** name matches your vCloud Director Organization name, which might also appear as your virtual data center (vDC) name. If you are using a Virtual Private Cloud, then this name is a unique identifier in the M123456789-12345 format. In a dedicated cloud, it is the given name of the target vDC.

If you are connecting directly to vCloud Director at the system level, for example leaving the Organization field blank, you need system administrator credentials. If you are entering an Organization in the endpoint, you need a user who has Organization Administrator credentials in that organization.

Provide credentials with rights to connect by using VMware Remote Console.

- To manage all organizations with a single endpoint, provide credentials for a system administrator.
- To manage each organization virtual datacenter (vDC) with a separate endpoint, create separate organization administrator credentials for each vDC.

Do not create a single system-level endpoint and individual organization endpoints for the same vCloud Director instance.

7 (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.

8 (Optional) To configure additional security and force connections to pass through a proxy server, click **Associations** and associate to an existing Proxy endpoint.

You must have at least one Proxy endpoint to create an association.

9 Click **OK**.

What to do next

Create a Fabric Group.

Create an Amazon Endpoint

You can create an endpoint to connect to an Amazon instance. You can optionally associate proxy settings to the Amazon endpoint by associating to a Proxy endpoint.

vRealize Automation provides several Amazon instance types for you to use when creating blueprints, but if you want to import your own instance types see Add an Amazon Instance Type.

For information about associating proxy settings to your endpoint, see Create a Proxy Endpoint and Associate to a Cloud Endpoint.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- If you want to configure additional security and force connections to pass through a proxy server, create a Proxy endpoint. You can associate to the Proxy endpoint as you create the Amazon endpoint. See Create a Proxy Endpoint and Associate to a Cloud Endpoint.

Procedure

1. Select **Infrastructure > Endpoints > Endpoints**.
2. Select **New > Cloud > Amazon EC2**.
3. Enter a name and, optionally, a description.

   Typically this name indicates the Amazon account that corresponds to this endpoint.
4. Enter the administrative-level access key ID for the Amazon endpoint.

   Only one endpoint can be associated with an Amazon access key ID.

   To obtain the access key needed to create the Amazon endpoint, you must either request a key from a user who has AWS Full Access Administrator credentials or be additionally configured with the AWS Full Access Administrator policy. See Amazon documentation for details.
5. Enter the secret access key for the Amazon endpoint.
6. (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.
7. (Optional) To configure additional security and force connections to pass through a proxy server, click **Associations** and associate to an existing Proxy endpoint.

   You must have at least one Proxy endpoint to create an association.
8. Click **OK**.

Results

After you create the endpoint, vRealize Automation begins collecting data from the Amazon Web Services regions.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Add an Amazon Instance Type
Several instance types are supplied with vRealize Automation for use with Amazon blueprints. An administrator can add and remove instance types.

The machine instance types managed by IaaS administrators are available to blueprint architects when they create or edit an Amazon blueprint. Amazon machine images and instance types are made available through the Amazon Web Services product.

Prerequisites

Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1. Click **Infrastructure > Administration > Instance Types**.
2. Click **New**.
3. Add a new instance type, specifying the following parameters.

   Information about the available Amazon instances types and the setting values that you can specify for these parameters is available from Amazon Web Services documentation in *EC2 Instance Types - Amazon Web Services (AWS)* at aws.amazon.com/ec2 and *Instance Types* at docs.aws.amazon.com.

   - Name
   - API name
   - Type Name
   - IO Performance Name
   - CPUs
   - Memory (GB)
   - Storage (GB)
   - Compute Units

4. Click the **Save** icon (✓).

Results

When IaaS architects create Amazon Web Services blueprints, they can use your custom instance types.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Create a Proxy Endpoint and Associate to a Cloud Endpoint

You can create a proxy endpoint and associate its proxy settings to a vCloud Air, vCloud Director, or Amazon endpoint.

If you upgraded or migrated a vCloud Air, vCloud Director, or Amazon endpoint that was using a proxy manager, a new vCloud Air, vCloud Director, or Amazon endpoint is created that contains an association between the vCloud Air, vCloud Director, or Amazon endpoint and a new Proxy endpoint.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- Create one of the following endpoint types:
  - Create a vCloud Air Endpoint
  - Create an Amazon Endpoint
  - Create a vCloud Director Endpoint

  You must have at least one vCloud Air, vCloud Director, or Amazon endpoint to create an association from the Proxy endpoint.

Procedure

1. Select **Infrastructure > Endpoints > Endpoints**.
2. Select **New > Network and Security > Proxy**.
3. Enter a name in the **Name** text box.
4. (Optional) Enter a description in the **Description** text box.
5. Enter the URL for the installed proxy agent in the **Address** text box.
6. Enter the port number to use for connecting to the proxy server in the **Port** text box.
7. Enter your administrator-level user name and password.
8. (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.
9. To associate the proxy settings to a vCloud Air, vCloud Director, or Amazon endpoint, click **Associations** and select one or more endpoints.

   You must have at least one vCloud Air, vCloud Director, or Amazon endpoint to create an association.

   You can associate the Proxy endpoint to more than one endpoint.
10. Click **OK**.

Results

vRealize Automation collects data from your endpoint and discovers your compute resources.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Create a Proxy Endpoint for OVF Host Web Site

You can create a proxy endpoint to use when importing OVF to a vSphere machine component in a blueprint or as a value set for an Image component profile when the OVF is hosted on a Web site.

For information about configuring for OVF deployment, see Create a vSphere Endpoint in vRealize Automation and Associate to NSX and Configuring a Blueprint to Provision from an OVF.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1 Select **Infrastructure > Endpoints > Endpoints**.

2 Select **New > Network and Security > Proxy**.

3 Enter a name in the **Name** text box.

4 (Optional) Enter a description in the **Description** text box.

5 Enter the URL for the Web site that is hosting the OVF in the **Address** text box.

6 Enter the port number to use for connecting to the Web site proxy server in the **Port** text box.

7 Enter your administrator-level user name and password.

8 (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.

9 Click **OK**.

Results

You can now use the endpoint to define the Web site at which to obtain OVF. For details, see Define Blueprint Settings for a vSphere Component By Using an OVF and Define an Image Value Set for a Component Profile By Using an OVF.

Create a vRealize Orchestrator Endpoint

You can create a vRealize Orchestrator endpoint to connect to a vRealize Orchestrator server.

You can configure multiple endpoints to connect to different vRealize Orchestrator servers, but you must configure a priority for each endpoint.

When executing vRealize Orchestrator workflows, vRealize Automation tries the highest priority vRealize Orchestrator endpoint first. If that endpoint is not reachable, then it proceeds to try the next highest priority endpoint until a vRealize Orchestrator server is available to run the workflow.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1. Select **Infrastructure > Endpoints > Endpoints**.
2. Select **New > Orchestration > vRealize Orchestrator**.
3. Enter a name and, optionally, a description.
4. Enter a URL with the fully qualified name or IP address of the vRealize Orchestrator server and the vRealize Orchestrator port number.

   The transport protocol must be HTTPS. If no port is specified, the default port 443 is used.

   To use the default vRealize Orchestrator instance embedded in the vRealize Automation appliance, type `https://`*`vrealize-automation-appliance-hostname`*`:443/vco`.
5. Provide your vRealize Orchestrator credentials in the **User name** and **Password** text boxes to connect to the vRealize Orchestrator endpoint.

   The credentials you use should have Execute permissions for any vRealize Orchestrator workflows to call from IaaS.

   To use the default vRealize Orchestrator instance embedded in the vRealize Automation appliance, the user name is `administrator@vsphere.local` and the password is the administrator password that was specified when configuring SSO.
6. Enter an integer greater than or equal to 1 in **Priority** text box.

   A lower value specifies a higher priority.
7. (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.
8. Click **OK**.

Configuring vRealize Orchestrator Endpoints for Networking
If you are using vRealize Automation workflows to call vRealize Orchestrator workflows, you must configure the vRealize Orchestrator instance or server as an endpoint.

For information about adding a vRealize Orchestrator endpoint, see Create a vRealize Orchestrator Endpoint.

You can associate a vRealize Orchestrator endpoint with a machine blueprint to make sure that all of the vRealize Orchestrator workflows for machines provisioned from that blueprint are run using that endpoint.

vRealize Automation by default includes an embedded vRealize Orchestrator instance. It is recommended that you use the embedded instance as your vRealize Orchestrator endpoint for running vRealize Automation workflows in a production or test environment or when creating a proof of concept.

It is also recommended that you use this vRealize Orchestrator endpoint for running vRealize Automation workflows in a production environment.

The vRealize Orchestrator plug-in is automatically installed with vRealize Orchestrator 7.1 and later. There is no separate vRealize Orchestrator plug-in to install.

Create a vRealize Operations Manager Endpoint

You can create a vRealize Operations Manager endpoint to connect to a vRealize Operations Manager host suite API.

For information about validating the vRealize Operations Manager connection and certificate trust, see Considerations When Using Test Connection.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1. Select **Infrastructure > Endpoints > Endpoints**.
2. Select **New > Management > vRealize Operations Manager**.
3. Enter a name and, optionally, a description.
4. Enter the URL for the vRealize Operations Manager server in the **Address** text box.

   The URL must be of the format: `https://`*`hostname/suite-api`*`.`
5. Enter your vRealize Operations Manager user name and password credentials.
6. (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.
7. (Optional) To validate the credentials, host endpoint address, and certificate trust, click **Test Connection**. The action also checks that the manager service and agent are running so that endpoint can be data-collected. The **OK** action tests for these same conditions.

   The **Test Connection** action returns information about any of the following conditions:

   - Certificate error

     If the certificate is not found, trusted, or has expired, you are prompted to accept a certificate thumb print. If you do not accept the thumb print, you can still save the endpoint but machine provisioning might fail.

   - Agent error

     The associated vSphere agent is not found. The agent must be running for the test to succeed.

   - Host error

The specified endpoint address is not reachable or the associated manager service is not running. The manager service must be running for the test to succeed.

- Credentials error

  The specified user name and password combination is invalid for the endpoint at the specified address.

- Timeout

  The test action could not complete in the allowed two-minute time period.

If the **Test Connection** action fails, you can still save the endpoint but machine provisioning might fail.

If there is a trusted certificate issue, for example the certificate has expired, you are prompted to accept a certificate thumb print.

8 Click **OK**.

Create a Third-Party IPAM Provider Endpoint

If you registered and configured a third-party IPAM endpoint type in vRealize Orchestrator, you can create an endpoint for that IPAM solution provider in vRealize Automation.

If you imported a vRealize Orchestrator package for providing an external IPAM solution and registered the IPAM endpoint type in vRealize Orchestrator, you can select that IPAM endpoint type when you create a vRealize Automation endpoint.

Note This example is based on use of the Infoblox IPAM plug-in, which is available for download at the VMware Solution Exchange. You can also use this procedure if you created your own IPAM provider package using the VMware-supplied IPAM Solution SDK. The procedure for importing and configuring your own third-party IPAM solution package is the same as described in the prerequisites.

The first IPAM endpoint for vRealize Automation is created when you register the endpoint type for the IPAM solution provider plug-in in vRealize Orchestrator.

Prerequisites

- Obtain and Import a Third-Party IPAM Provider Package in vRealize Orchestrator.
- Run Workflow to Register Third-Party IPAM Endpoint Type in vRealize Orchestrator.
- Log in to vRealize Automation as an **IaaS administrator**.

For this example, create an Infoblox IPAM endpoint using an endpoint type that you registered in vRealize Orchestrator for your third-party IPAM provider plug-in or package.

Procedure

1 Select **Infrastructure > Endpoints > Endpoints**.

2 Select **New > IPAM > *IPAM endpoint type***.

   Select a registered external IPAM provider endpoint type such as Infoblox. External IPAM provider endpoints are only available if you imported a third-party vRealize Orchestrator package, and run the package workflows to register the endpoint type.

   For Infoblox IPAM, only primary IPAM endpoint types are listed. You can specify secondary IPAM endpoint types by using custom properties.

   For this example, select a registered external IPAM endpoint type, for example **Infoblox NIOS**.

3 Enter a name and, optionally, a description.

4 Enter the location of the registered IPAM endpoint in the **Address** text box using the provider-specific URL format, for example https:/*host_name/name*.

   For example, you might create several IPAM endpoints, such as `https://nsx62-scale-infoblox` and `https://nsx62-scale-infoblox2`, when you registered the IPAM endpoint type in vRealize Orchestrator. Enter a primary registered endpoint type. To also specify one or more secondary IPAM endpoints, you can use custom properties to emulate the extensible attributes that are specific to the IPAM solution provider.

5 To access the IPAM solution provider account, enter the user name and password required .

   The IPAM solution provider account credentials are required to create, configure, and edit the endpoint when working in vRealize Automation. vRealize Automation uses the IPAM endpoint credentials to communicate with the specified endpoint type, for example Infoblox, to allocate IP addresses and perform other operations. This behavior is similar to how vRealize Automation uses vSphere endpoint credentials.

6 (Optional) Click **Properties** and add endpoint properties that are meaningful to the specific IPAM solution provider.

   Each IPAM solution provider, for example Infoblox and Bluecat, use unique extensible attributes that you can emulate by using vRealize Automation custom properties. For example, Infoblox uses extensible attributes to differentiate primary and secondary endpoints.

7 Click **OK**.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Create a Microsoft Azure Endpoint

You can create a Microsoft Azure endpoint to facilitate a credentialed connection between vRealize Automation and an Azure deployment.

An endpoint establishes a connection to a resource, in this case an Azure instance, that you can use to create virtual machine blueprints. You must have an Azure endpoint to use as the basis of blueprints for provisioning Azure virtual machines. If you use multiple Azure subscriptions, you need endpoints for each subscription ID.

As an alternative, you can create an Azure connection directly from vRealize Orchestrator using the Add an Azure Connection command located under **Library > Azure > Configuration** in the vRealize Orchestrator workflow tree. For most scenarios, creating a connection through the endpoint configuration as described herein is the preferred option.

Azure endpoints are supported by vRealize Orchestrator and XaaS functionality. You can create, delete, or edit an Azure endpoint. If you change an existing endpoint and do not run any updates on the Azure portal through the updated connection for several hours, problems may occur. You must restart the vRealize Orchestrator service using the `service vco-service restart` command. Failure to restart the service may result in errors.

Prerequisites

- Configure a Microsoft Azure instance and obtain a valid Microsoft Azure subscription from which you can use the subscription ID. See Microsoft Azure Endpoint Configuration for more information about configuring Azure and obtaining a subscription ID.
- Verify that your vRealize Automation deployment has at least one tenant and one business group.
- Create an Active Directory application as described in https://azure.microsoft.com/en-us/documentation/articles/resource-group-create-service-principal-portal.
- Make note of the following Azure related information, as you will need it during endpoint and blueprint configuration.
  - subscription ID
  - tenant ID
  - storage account name
  - resource group name
  - location
  - virtual network name
  - client application ID
  - client application secret key
  - virtual machine image URN
- The vRealize Automation Azure implementation supports a subset of the Microsoft Azure supported regions. See Azure Supported Regions.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1. Select **Administration > vRO Configuration > Endpoints**.
2. Click the **New** icon (✚).
3. On the Plug-in tab, click the **Plug-in** drop-down menu and select **Azure**.

4 Click **Next**.

5 Enter a name and, optionally, a description.

6 Click **Next**.

7 Populate the text boxes on the Details tab as appropriate for the endpoint.

Parameter	Description
Connection settings	
Connection name	Unique name for the new endpoint connection. This name appears in the vRealize Orchestrator interface to help you identify a particular connection.
Azure subscription id	The identifier for your Azure subscription. The ID defines the storage accounts, virtual machines and other Azure resources to which you have access.
Azure Environment	The geographic region for the deployed Azure resource. vRealize Automation supports all current Azure regions based on the subscription ID.
Resource manager settings	
Azure service URI	The URI through which you gain access to your Azure instance. The default value of `https://management.azure.com/` is appropriate for many typical implementations. This box is auto-populated when you select an environment.
Tenant Id	The Azure tenant ID that you want the endpoint to use.
Client Id	The Azure client identifier that you want the endpoint to use. This is assigned when you create an Active Directory application.
Client secret	The key used with an Azure client ID. This key is assigned when you create an Active Directory application.
Azure storage URI	The URI through which you gain access to your Azure storage instance. This box is auto-populated when you select an environment.
Proxy Settings	
Proxy host	If your company uses a proxy Web server, enter the host name of that server.
Proxy port	If your company uses a proxy Web server, enter the port number of that server.

8 (Optional) Click Properties and add supplied custom properties, property groups, or your own custom property definitions.

9 Click **Finish**.

What to do next

Create appropriate resource groups, storage accounts, and network security groups in Azure. You should also create load balancers if appropriate for your implementation.

Action	Options
Create an Azure resource group	■ Create the resource group using the Azure portal. See the Azure documentation for specific instructions. ■ Use the appropriate vRealize Orchestrator workflow found under the `Library/Azure/Resource/Create resource group`. ■ In vRealize Automation, create and publish an XaaS blueprint that contains the vRealize Orchestrator workflow. You can request the resource group after attaching it to the service and entitlements. **Note** The Resource Group resource type is not supported or managed by vRealize Automation.
Create an Azure storage account	■ Use Azure to create a storage account. See the Azure documentation for specific instructions. ■ Use the appropriate vRealize Orchestrator workflow found under `Library/Azure/Storage/Create storage account`. ■ In vRealize Automation, create and publish an XaaS blueprint that contains the vRealize Orchestrator workflow. You can request the storage account after attaching it to the service and entitlements.
Create an Azure network security group	■ Use Azure to create a security group. See the Azure documentation for specific instructions. ■ Use the appropriate vRealize Orchestrator workflow found under the `Library/Azure/Network/Create Network security group` . ■ In vRealize Automation, create and publish an XaaS blueprint that contains the vRealize Orchestrator workflow. You can request the security group after attaching it to the service and entitlements.

Microsoft Azure Endpoint Configuration

You must gather some information and perform some configuration in order to create a Microsoft Azure endpoint in vRealize Automation.

Procedure

1 Locate and record your Microsoft Azure subscritption and tenant IDs.

- Subscription ID - Click the Subscriptions icon on the left toolbar in your Azure portal to view the subscription ID.
- Tenant ID - Click the Help icon and select Show Diagnostics in your Azure portal. Search for tenant and record the ID when you have located it.

2 You can create a new storage account and a resource group to get started. Altenatively, you can create these in blueprints later.

- Storage Account - Use the following procedure to configure an account.

  1 In your Azure portal, locate the Storage Accounts icon on the sidebar. Make sure the correct subscription is selected and click **Add**. You can also, search for storage account in the Azure search field.

  2 Enter the required information for the storage account. You will need your subscription ID.

  3 Select whether to use an existing resource group or create a new one. Make note of your resoruce group name, as you will need it later.

Note Save the location of your storage account as you will need it later.

3 Create a virtual network. Alternatively, if you have a suitable existing network, you can select that one.

If you are creating a network, you must select Use an Existing Resource Group and specify the group that you created in the preceding step. Also, select the same location that you specified previously. Microsoft Azure will not deploy virtual machines or other objects if the location doesn't match between all applicable components that the object will consume.

a Locate the Virtual Network icon on the left panel and click it or search for virtual network. Make sure to select the correct subscription and click **Add**.

b Enter a unique name for your new virtual network and record it for later.

c Enter the appropriate IP address for your virtual network in the **Address space** field.

d Ensure that the correct subscription is selected and click **Add**.

e Enter the remaining basic configuration information.

f You can modify the other options as necessary, but for most configurations, you can leave the defaults.

g Click **Create**.

4 Set up an Azure Active Directory application so that vRealize Automation can authenticate.

a Locate the Active Directory icon on the Azure left menu and click it.

b Click **App Registrations** and select **Add**.

c Type a name for your application that complies with Azure name validation.

d Leave Web app/API as the Application Type.

e The Sign-on URL can be anything that is appropriate for your usage.

f Click **Create**.

5 Create a secret key to authenticate the application in vRealize Automation.

   a Click the name of your application in Azure.

      Make note of your Application ID for later use.

   b Click **All Settings** in the next pane and select Keys from the settings list.

   c Enter a description for the new key and choose a duration.

   d Click **Save** and make sure to copy the key value to a safe location as you will be unable to retrieve it later.

   e On the left menu, select **API Permissions** for the application and click **Add a Permission** to create a new permission.

   f Select Azure Service Management on the Select an API page.

   g Click **Delegated Permissions**.

   h Under Select permissions select user_impersonation and then click **Add Permissions**.

6 Authorize your Active Directory application to connect to your Azure subscription so that you can deploy and manage virtual machines.

   a In the left menu, click the Subscriptions icon, and select your new subscription.

      You may need to click on the text of the name to get the panel to slide over.

   b Select the Access control (IAM) option to see the permissions to your subscription.

   c Click **Add** under the Add a Role Assignment heading.

   d Choose Contributor from the Role drop down.

   e Leave the default selection in the Assign Access to drop down.

   f Type the name of your application in the Select box.

   g Click **Save.**

   h Add additional roles so that your new application has Owner, Contributor, and Reader roles.

   i Click the **Save**.

What to do next

You must install the Microsoft Azure command line interface tools. These tools are freely available for both Windows and Mac operating systems. See the Microsoft documentation for more information about downloading and installing these tools.

When you have the command line interface installed, you must authenticate to your new subscription.

1 Open a terminal window and type your Microsoft Azure login. You will receive a URL and a shortcode that will allow you to authenticate.

2 In a browser, enter the code that you received from the application on your device.

3 Enter your Auth Code and click **Continue**.

4 Select your Azure account and login.

   If you have multiple subscriptions, ensure that the correct one is selected using the `azure account set <subscription-name>` command.

5 Before you proceed, you must register the Microsoft.Compute provider to your new Azure subscription using the `azure provider register microsoft.compute` command.

   If the command times out and generates an error the first time your run it, run it again.

When you have completed configuration, you can use the `azure vm image list` command to retrieve available virtual machine image names. You can choose the desired image and record the URN provided for it and later use it in blueprints.

Create a Puppet Endpoint

You can create a Puppet endpoint to support addition of Puppet configuration management components to vSphere virtual machines. These components enable you to use a Puppet Master to enforce configuration management on virtual machines.

An endpoint establishes a connection to an external resource, in this case a Puppet Master instance. The endpoint enables you to place Puppet configuration management components on vSphere virtual machine blueprints. Provisioned virtual machines based on these blueprints contain a Puppet agent that facilitates control by the associated Puppet Master.

For more information about the Puppet plug-in and a demo of its configuration, see https://www.youtube.com/watch?v=P-VglzE9o-o.

Prerequisites

- Install and configure Puppet Enterprise as appropriate for your environment.
- Download and install the Puppet plug-in version 3.0 on your vRealize Orchestrator deployment. You can download the plug-in from https://marketplace.vmware.com/vsx/solutions/puppet-plugin-for-vrealize-automation?ref=search. See https://docs.puppet.com/pe/latest/vro_intro.html for information about installing and using the plug-in.

Procedure

1 Select **Administration > vRO Configuration > Endpoints**.

2 Click the **New** icon (✚).

3 On the Plug-in tab, click the **Plug-in** drop-down menu and select **Puppet Plug-in**.

4 Click **Next**.

5 Enter a name and, optionally, a description.

6 Click **Next**.

7 Populate the text boxes on the **Details** tab as appropriate for the endpoint.

Parameter	Description
Display name for this Puppet Master	The name of the Puppet Master associated with the endpoint connection . This name appears in the vRealize Orchestrator interface to help you identify a particular connection.
Hostname or IP address	The FQDN or IP address of the Puppet Master used by this endpoint.
SSH Port	The port defined for use with secure communication for this Puppet Master.
SSH RBAC and Username	The role based access control username required to connect with the Puppet Master.
SSH and RBAC Password	The role based access control username required for secure configuration with the Puppet Master.
Use sudo for shell commands on this master?	Select this option if you want administrators to be able to use Sudo commands on Linux servers for security options on virtual machines based on this endpoint..

8 Click **OK**.

Results

You can now add Puppet configuration management components to vSphere blueprints so that you can deploy vSphere virtual machines that contain Puppet agents.

Create an Ansible Endpoint

You can create an Ansible endpoint to support the addition of Ansible configuration management components to vSphere virtual machines. These components enable you to use an Ansible tower to enforce configuration management on virtual machines.

Prerequisites

- Install and configure an Ansible Tower as appropriate for your environment.
- Download and install the Ansible plug-in on youvRealize Orchestrator deployment. The plug-in is available from https://marketplace.vmware.com/vsx/solutions/sovlabs-ansible-tower-plug-in-for-vra-cm-framework-1?ref=search.

Procedure

1 Select **Administration > vRO Configuration > Endpoints**.

2 Click the **New** icon.

3 On the Plug-in tab, click the **Plug-in** drop-down menu and select Ansible Plug-in.

4 Click **Next**.

5 Enter a name and, optionally, a description on the Endpoint tab.

6 Click **Next**.

7 Populate the text boxes on the Details tab pages as appropriate for the endpoint.

Details Tab Page	Description
Ansible Tower Endpoint Configuration	Add endpoint configuration information. ■ Ansible Tower Endpoint Configuration: Enter the name and IP address or host name in the appropriate text boxes. ■ Ansible Tower Credential Configuration: Enter the login credentials for the Ansible tower associated with this endpoint. ■ Import SSL Certificate: Select whether you want the Ansible tower certificate to be accepted by vRealize Orchestrator silently.
Ansible Tower Host Access	If applicable enter the SSH credentials for the Ansible Tower machine so that a deployed machine can connect to it to configure a custom dynamic inventory script.
Organization and Inventory Setup	Configure your organization name and inventory. Add dynamic inventory configuration values.
Filters and Groups	Configure key value pair property filters and Ansible dynamic groups.
Prompt on Launch Overrides (Optional)	Configure Ansible Job options as well as machine, template and inventory options.
vRA Property Translation	If applicable, enter the desired replacement string for use by Ansible during processing of custom properties after provisioning.

8 Click **Finish**.

Create a Hyper-V (SCVMM) Endpoint

You can create endpoints to allow vRealize Automation to communicate with your SCVMM environment and discover compute resources, collect data, and provision machines.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- You must install and configure a DEM agent to manage your Hyper-V (SCVMM) endpoint. For information, see SCVMM Requirements.

For related information, see Preparing Your SCVMM Environment.

Procedure

1 Select **Infrastructure > Endpoints > Endpoints**.

2 Select **New > Virtual > Hyper-V (SCVMM)**.

3 Enter a name in the **Name** text box.

4 (Optional) Enter a description in the **Description** text box.

5 Enter the URL for the endpoint in the **Address** text box.

   The URL must be of the type: *`FQDN`* or *`IP_address`*.

   For example: `mycompany-scvmm1.mycompany.local`.

6 Enter the administrative-level user name and password that you stored for this endpoint.

   If you did not already store the credentials, you can do so now.

7 (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.

8 Click **OK**.

Results

vRealize Automation collects data from your endpoint and discovers your compute resources.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Create an OpenStack Endpoint

You create an endpoint to allow vRealize Automation to communicate with your OpenStack instance.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- Verify that your vRealize Automation DEMs are installed on a machine that meets the Openstack or PowerVC requirements. See OpenStack Requirements.
- Verify that your flavor of OpenStack is currently supported. See *vRealize Automation Support Matrix*.

After you upgrade or migrate from an earlier vRealize Automation installation, if data collection fails for OpenStack endpoints you can add the `VMware.Endpoint.Openstack.IdentityProvider.Domain.Name` custom property to each Keystone V3 OpenStack endpoint to specify a valid domain name and enable data collection.

Procedure

1 Select **Infrastructure > Endpoints > Endpoints**.

2 Select **New > Cloud > OpenStack**.

3 Enter a name and, optionally, a description.

4 Enter the URL for the endpoint in the **Address** text box.

Option	Description
PowerVC	The URL must be of the format `http://`*`myPowerVC`*`.com:5000` or `http://`*`FQDN`*`:5000`.
Openstack	The URL must be of the format *`FQDN`*`:5000` or *`IP_address`*`:5000`. Do not include the `/v2.0` suffix in the endpoint address.

5 Enter your administrative-level user name and password.

The credentials you provide must have the administrator role in the OpenStack tenant associated with the endpoint.

6 Enter an OpenStack tenant name in the **OpenStack project** text box.

If you set up multiple endpoints with different OpenStack tenants, create reservation policies for each tenant. This ensures that machines are provisioned to the appropriate tenant resources.

7 Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.

If Keystone V3 is in effect, add the `VMware.Endpoint.Openstack.IdentityProvider.Domain.Name` custom property to designate a specific domain.

8 Click **OK**.

Results

vRealize Automation collects data from your endpoint and discovers your compute resources.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Create a Hyper-V, XenServer, or Xen Pool Endpoint

You can create endpoints to allow vRealize Automation to communicate with the Hyper-V, XenServer, or Xen pool main environment and discover compute resources, collect data, and provision machines.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- A system administrator must install a proxy agent with stored credentials that correspond to your endpoint. See Installing the Proxy Agent for Hyper-V or XenServer.

Procedure

1 Select **Infrastructure > Endpoints > Agents**.

2 Enter the fully qualified DNS name of your Hyper-V server, Xen server, or Xen main pool in the **Compute resource** text box.

   Note For a Xen pool endpoint, you must enter the name of the main pool. To avoid duplicate entries in the vRealize Automation compute resource table, specify an address that matches the configured Xen pool main address. For example, if the Xen pool main address uses the host name, enter the host name and not the FQDN. If the Xen pool main address uses FQDN, then enter the FQDN.

3 Select the proxy agent that your system administrator installed for this endpoint from the **Proxy agent name** drop-down menu.

4 (Optional) Enter a description in the **Description** text box.

5 Click **OK**.

Results

vRealize Automation collects data from your endpoint and discovers your compute resources.

What to do next

Add the compute resources from your endpoint to a fabric group. See Create a Fabric Group.

Considerations When Using Test Connection

You can use the test connection action to validate the credentials, host endpoint address, and certificate for a vSphere, NSX for vSphere, NSX-T, and vRealize Operations Manager endpoint.

The action also checks that the manager service and agent are running so that the endpoint can be data-collected.

The **Test Connection** action returns information about any of the following conditions:

- Certificate error

  If the certificate is not found, trusted, or has expired, you are prompted to accept a certificate thumb print. If you do not accept the thumb print, you can still save the endpoint but machine provisioning might fail.

- Agent error

  The associated vSphere agent is not found. The agent must be running for the test to succeed.

- Host error

  The specified endpoint address is not reachable or the associated manager service is not running. The manager service must be running for the test to succeed.

- Credentials error

  The specified user name and password combination is invalid for the endpoint at the specified address.

- Timeout

  The test action could not complete in the allowed two-minute time period.

If you receive errors when running **Test Connection** on upgraded or migrated endpoints, see Considerations When Working With Upgraded or Migrated Endpoints for steps needed to establish certificate trust.

Import or Export Endpoints Programmatically

To programmatically import and export endpoints in vRealize Automation 7.3 or later you must use either new vRealize Automation endpoint-configuration-service REST APIs or use vRealize CloudClient.

The vRealize CloudClient documentation contains all applicable command line formatting, samples, and usage information.

You can download the vRealize CloudClient application and documentation from the vRealize CloudClient product page of the https://developercenter.vmware.com/tool/cloudclient.

Viewing Compute Resources and Running Data Collection

You can view the machine and compute resource that is associated with a specific endpoint. You can also manually start data collection.

Prerequisites

Verify that at least one endpoint exists.

Procedure

1. Select **Infrastructure > Endpoints > Endpoints**.

   Users who do not have IaaS administrator privileges, can select **Infrastructure > Compute Resources > Compute Resources** to view resources and run data collection from the compute resource.

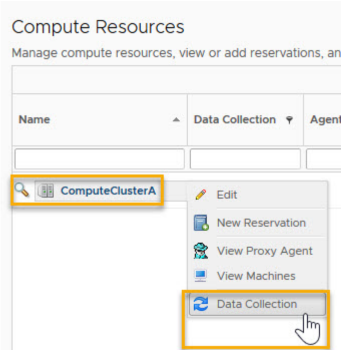
2. Select **Infrastructure > Endpoints > Endpoints**.

3. Select an existing endpoint row and click **Actions**.

   Select any one of the following available actions.

   - Click **View Compute Resources** to open the **Infrastructure > Compute Resource** page. You can use this page to view and edit compute resource settings. You can also run data collection for a selected compute resource from the **Compute Resources** page.
   - Click **View Machines** to open the **Infrastructure > Managed Machines** page.
   - Click **Data Collection** to open the Data Collection page and start data collection for the endpoint. You can refresh the page to display the current status of the request.

You can run data collection from an endpoint's associated compute resource. For example, to data-collect an existing NSX-T endpoint, use **Infrastructure > Compute Resources > Compute Resources** to view resources and then click **Data Collection** to open the **Data Collection** page for the compute resource. Find the desired endpoint in the list and click **Request Now**.



Considerations When Working With Upgraded or Migrated Endpoints

After you upgrade or migrate from a pre-vRealize Automation 7.3 release, the following considerations are important to understand and act on.

This information applies to endpoints that were upgraded or migrated to this vRealize Automation release.

- When you upgrade or migrate from a pre-vRealize Automation 7.3 release, each vCloud Air, vCloud Director, and Amazon endpoint that contains proxy settings is associated to a new proxy endpoint that contains its proxy settings.

  After upgrade or migration, the new Proxy endpoint name is Proxy_*YYYYY* where *YYYYY* is a hash of the proxy's URL, port, and credentials. If you used the same proxy settings (for example the same URL, port, and credentials) for a different endpoint (for example, a vCloud Air or Amazon endpoint), after upgrade or migration there is only one Proxy endpoint and an association between the vCloud Air and Amazon endpoint and the new Proxy endpoint. A proxy endpoint can be associated to more than one Amazon, vCloud Air or vCloud Director endpoint.

- When you upgrade or migrate vSphere endpoints that contain NSX manager settings, each vSphere endpoint is associated to a new NSX endpoint that contains its NSX manager settings.

  After upgrade or migration, the NSX endpoint name is NSX_*XXXXX* where *XXXXX* is the name of the parent vSphere endpoint in the pre-vRealize Automation 7.3 release.

- When vRealize Automation upgrade or migration is finished, an infrastructure administrator can change the new NSX and Proxy endpoint names.

- The default security setting for upgraded or migrated endpoints is not to accept untrusted certificates.

- After upgrading or migrating from an earlier vRealize Automation installation, if you were using untrusted certificates you must perform the following steps for all vSphere and NSX endpoints to enable certificate validation. Otherwise, the endpoint operations fail with certificate errors. For more information, see VMware Knowledge Base articles *Endpoint communication is broken after upgrade to vRA 7.3 (2150230)* at http://kb.vmware.com/kb/2150230 and *How to download and install vCenter Server root certificates to avoid Web Browser certificate warnings (2108294)* at http://kb.vmware.com/kb/2108294.

  a After upgrade or migration, log in to the vRealize Automation vSphere agent machine and restart your vSphere agents by using the **Services** tab.

  Migration might not restart all agents, so manually restart them if needed.

  b Wait for at least one ping report to finish. It takes a minute or two for a ping report to finish.

  c When the vSphere agents have started data collection, log in to vRealize Automation as an IaaS administrator.

  d Click **Infrastructure > Endpoints > Endpoints**.

  e Edit a vSphere endpoint and click **Test Connection**.

  f If a certificate prompt appears, click **OK** to accept the certificate.

  If a certificate prompt does not appear, the certificate might currently be correctly stored in a trusted root authority of the Windows machine hosting service for the endpoint, for example as a proxy agent machine or DEM machine.

  g To apply the certificate acceptance and save the endpoint, click **OK**.

  h Repeat this procedure for each vSphere endpoint.

  i Repeat this procedure for each NSX endpoint.

  j Navigate to **Infrastructure > Compute Resources**, right click on your **vCenter Compute** resource, and run **Data Collection**.

  If the **Test Connection** action is successful but some data collection or provisioning operations fail, you can install the same certificate on all the agent machines that serve the endpoint and on all DEM machines. Alternatively, you can uninstall the certificate from existing machines and repeat the preceding procedure for the failing endpoint.

- The vRealize Automation REST APIs that were used to programmatically create, edit, and delete endpoints in vRealize Automation 7.2 and earlier are no longer supported in vRealize Automation 7.3 and later. To programmatically create, edit, and delete endpoints in vRealize Automation 7.3 or later you must use either new vRealize Automation endpoint-configuration-service REST APIs or use vRealize CloudClient.

- After you upgrade or migrate from an earlier vRealize Automation installation, if data collection fails for OpenStack endpoints you can add the `VMware.Endpoint.Openstack.IdentityProvider.Domain.Name` custom property to each Keystone V3 OpenStack endpoint to specify a valid domain name and enable data collection.

- When you upgrade a third-party IPAM endpoint, such as Infoblox IPAM, the vRealize Orchestrator package that contains the `RegisterIPAMEndpoint` workflow is upgraded. You might need to rerun the workflow in vRealize Orchestrator when the vRealize Automation upgrade is finished.
- To make a credentials change to multiple endpoints, you can either edit the endpoints individually or use vRealize CloudClient to perform a bulk update.
- Some endpoint types, such as vCloud Air and vCloud Director, cannot be upgraded or migrated directly from vRealize Automation 6.2.x to vRealize Automation 7.3 or greater.
- After a successful upgrade or migration to vRealize Automation 7.3, if the **Infrastructure > Endpoints** page does not show any endpoints or only shows some endpoint types and endpoints, see Knowledge Base Article 2150252 for a suggested workaround.

Considerations When Deleting Endpoints

You can delete certain endpoint types under certain conditions.

- You can delete endpoints that have not been data-collected.
- You can delete an OpenStack, Amazon and VRO endpoint if it has been data collected, but has no reservations. Other endpoint types cannot be deleted if they have been data collected.
- You can delete a third-party IPAM endpoint if it has no association to a network profile.
- When deleting a vSphere endpoint, the confirmation prompt lists the following dependencies:
  - The endpoint has been data-collected.
  - The endpoint is referenced in a reservation that maps to a compute resource. You cannot delete an endpoint is referenced in a reservation. Reservations require a compute resource.
  - The endpoint contains a template that is referenced in an existing blueprint.

    The blueprint is not deleted when you delete the endpoint.
  - The endpoint is used by virtual machines that are in use.
- You can delete endpoints programmatically by using either the new CREATE, EDIT, and DELETE vRealize Automation endpoint-configuration-service REST APIs introduced in vRealize Automation 7.3 or by using vRealize CloudClient. You cannot delete endpoints by using the pre-vRealize Automation 7.3 endpoint-configuration-service REST APIs.

Troubleshooting Attached vSphere Endpoint Cannot be Found

When data collection fails for a vSphere endpoint, it can be due to a mismatch between the proxy name and the endpoint name.

Problem

Data collection fails for a vSphere endpoint. The log messages return an error similar to the following:

```
This exception was caught: The attached endpoint
             'vCenter' cannot be found.
```

Cause

The endpoint name you configure in vRealize Automation must match the endpoint name provided to the vSphere proxy agent during installation. Data collection fails for a vSphere endpoint if there is a mismatch between the endpoint name and the proxy agent name. Until an endpoint with a matching name is configured, the log messages return an error similar to the following:

```
This exception was caught: The attached endpoint
             'expected endpoint name' cannot be found.
```

Solution

1. Select **Infrastructure > Monitoring > Log**.
2. Look for an `Attached Endpoint Cannot be Found` error message.

   For example,

   ```
   This exception was caught: The attached endpoint
                'expected endpoint name' cannot be found.
   ```
3. Edit your vSphere endpoint to match the expected endpoint name shown in the log message.
   a. Select **Infrastructure > Endpoints > Endpoints**.
   b. Click the name of the endpoint to edit.
   c. Enter the expected endpoint name in the **Name** text box.
   d. Click **OK**.

Solution

The proxy agent can commute with the endpoint and data collection is successful.

Create a Fabric Group

You can organize infrastructure resources into fabric groups and assign one or more fabric administrators to manage the resources in the fabric group.

Fabric groups are required for virtual and cloud endpoints. You can grant the fabric administrator role to multiple users by either adding multiple users one at a time or by choosing an identity store group or custom group as your fabric administrator.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.
- Create at least one endpoint. See Choosing an Endpoint Scenario.

Procedure

1. Select **Infrastructure > Endpoints > Fabric Groups**.
2. Click the **New** icon (✚).
3. Enter a name in the **Name** text box.
4. (Optional) Enter a description in the **Description** text box.
5. Enter a user name or user email address in the **Fabric administrators** text box, click the Search icon, and select the provided user email address.

   Repeat this step to add multiple users.
6. Select one or more **Compute resources** to include in your fabric group.

   Only resources that exist in the clusters you select for your fabric group are discovered during data collection. For example, only templates that exist in the clusters you select are discovered and available for cloning on reservations you create for business groups.
7. Click **OK**.

Results

Fabric administrators can now configure machine prefixes. See Configure Machine Prefixes.

Users who are currently logged in to the vRealize Automation must log out and log back in to the vRealize Automation before they can navigate to the pages to which they have been granted access.

Configure Machine Prefixes

You can create machine prefixes that are used to create names for machines provisioned through vRealize Automation. A machine prefix is required when defining a machine component in the blueprint design canvas.

A prefix is a base name to be followed by a counter of a specified number of digits. When the digits are all used, vRealize Automation rolls back to the first number.

Machine prefixes must conform to the following limitations:

- Contain only the case-insensitive ASCII letters a through z, the digits 0 through 9, and the hyphen (-).
- Not begin with a hyphen.
- No other symbols, punctuation characters, or blank spaces can be used.
- No longer than 15 characters, including the digits, to conform to the Windows limit of 15 characters in host names.

Longer host names are truncated when a machine is provisioned, and updated the next time data collection is run. However, for WIM provisioning names are not truncated and provisioning fails when the specified name is longer than 15 characters.

- vRealize Automation does not support multiple virtual machines of the same name in a single instance. If you choose a naming convention that causes an overlap in machine names, vRealize Automation does not provision a machine with the redundant name. If possible, vRealize Automation skips the name that is already in use and generates a new machine name using the specified machine prefix. If a unique name cannot be generated, provisioning fails.

Prerequisites

Log in to vRealize Automation as a **fabric administrator**.

Procedure

1 Click **Infrastructure > Administration > Machine Prefixes**.

2 Click the **New** icon (✚).

3 Enter the machine prefix in the **Name** text box.

4 Specify if the machine prefix is displayed in all tenants or only in the current tenant in the **Visibility** column.

5 Enter the number of machine prefix digits in the **Number of Digits** text box.

6 Enter the counter start number in the **Next Number** text box.

7 Click the **Save** icon (✓).

Results

Tenant administrators can create business groups so that users can access vRealize Automation to request machines.

Creating a Network Profile in vRealize Automation

A network profile contains IP information such as gateway, subnet, and address range. vRealize Automation uses vSphere DHCP or a specified IPAM provider to assign IP addresses to the machines it provisions based on network profile settings.

You can create a network profile to define a type of available network. You can create external network profiles and templates for on-demand network address translation (NAT) and routed or private network profiles. The profiles can build NSX logical switches and appropriate routing settings for a network path.

Network profiles are used to configure network settings when machines are provisioned. Network profiles also specify the configuration of NSX Edge devices that are created when you provision machines.

Available Network Types

The following network types are available as you define a network profile:

- Existing network
- On-demand routed network
- On-demand NAT network
- On-demand private network (NSX for vSphere only)

Table 4-14. Available Network Types for a vRealize Automation Network Profile

Network Type	Description
External	Existing network configured on the vSphere server. They are the external part of the NAT and routed networks types. An external network profile can define a range of static IP addresses available on the external network. You can use IP ranges obtained from the supplied VMware IPAM endpoint or from a third-party IPAM service provider endpoint that you have registered and configured in vRealize Orchestrator, such as Infoblox IPAM. An IP range is created from an IP block during allocation. An external network profile with a static IP range is a prerequisite for NAT and routed networks. See Creating an External Network Profile For an Existing Network.
NAT	On-demand network created during provisioning. NAT networks that use one set of IP addresses for external communication and another set for internal communications. With one-to-one NAT networks, every virtual machine is assigned an external IP address from the external network profile and an internal IP address from the NAT network profile. With one-to-many NAT networks, all machines share a single IP address from the external network profile for external communication. You can use IP ranges obtained from the supplied VMware IPAM endpoint or from a third-party IPAM service provider endpoint that you have registered and configured in vRealize Orchestrator, such as Infoblox IPAM. An IP range is created from an IP block during allocation. A NAT network profile defines local and external networks that use a translation table for mutual communication. See Creating a NAT Network Profile For an On-Demand Network.

Table 4-14. Available Network Types for a vRealize Automation Network Profile (continued)

Network Type	Description
Routed	On-demand network created during provisioning. Routed networks contain a routable IP space divided across subnets that are linked together using Distributed Logical Router (DLR). Every new routed network has the next available subnet assigned to it and is associated with other routed networks that use the same network profile. The virtual machines that are provisioned with routed networks that have the same routed network profile can communicate with each other and the external network. You can use IP ranges obtained from the supplied VMware IPAM endpoint or from a third-party IPAM service provider endpoint that you have registered and configured in vRealize Orchestrator, such as Infoblox IPAM. An IP range is created from an IP block during allocation. A routed network profile defines a routable space and available subnets. See Creating a Routed Network Profile For an On-Demand Network.
Private (NSX for vSphere only)	On-demand network created during provisioning. This option is only available for NSX for vSphere. This option is not available for NSX-T. Private networks include the following considerations: ■ Private networks have no inbound or outbound connectivity. An edge is not provisioned for private networks. ■ You can create a private network profile with or without static IP addresses or ranges. DHCP and third-party IPAM are not supported for private networks. See Create a Private Network Profile for an On-Demand Network in vRealize Automation.

For NSX information about networking, see VMware NSX Data Center for vSphere Documentation and VMware NSX-T Data Center Documentation.

For related information about configuring networking and security for NSX-T in vRealize Automation, see VMware blog Application Networking and Security with vRealize Automation and NSX-T.

Using Supplied or Third-party IPAM

Network profiles also support third-party IP Address Management (IPAM) providers, such as Infoblox. When you configure a network profile for IPAM, your provisioned machines can obtain their IP address data, and related information such as DNS and gateway, from the configured IPAM solution. You can use an external IPAM package for a third-party provider, such as Infoblox, to define an IPAM endpoint for use with a network profile.

Note If you are using a third-party IPAM provider and want to specify on which network to deploy your machine, use a separate network profile for each VLAN to avoid the known issue described in Knowledge Base Article 2148656.

If you do not use a third-party IPAM provider, but instead use the vRealize Automation-supplied IPAM endpoint, you can specify the ranges of IP addresses that network profiles can use. Each IP address in the specified ranges that are allocated to a machine is reclaimed for reassignment when the machine is destroyed. You can create a network profile to define a range of static IP addresses that can be assigned to machines. When provisioning virtual machines by cloning or by using kickstart/autoYaST provisioning, the requesting machine owner can assign static IP addresses from a predetermined range.

Specifying a Network Profile in a Reservation or Blueprint

You specify a network profile when you create reservations and blueprints. In a reservation, you can assign a network profile to a network path and specify any one of those paths for a machine component in a blueprint. You can assign a network profile to a specific network path on a reservation. For some machine component types, such as vSphere, you can assign a network profile when you create or edit blueprints.

You can use an existing network profile and an on-demand network profile as you define network adapters and load balancers for a vSphere machine.

If you specify a network profile in a reservation and a blueprint, the blueprint values take precedence.

Making Changes After Blueprint Deployment

While you cannot change the network profile of a deployed virtual machine, you can change the network to which the VM is connected. If the network is associated to a different network profile, vRealize Automation assigns an IP address from that network profile to the VM. The VM continues to use the old IP address until you update the IP address on the guest operating system. If you use the Reconfigure action on the deployed VM, you must update the IP address on the guest operating system.

Using Network Profiles to Control IP Address Ranges

You can use network profiles to assign static IP addresses from a predefined range to virtual machines that are provisioned by cloning, by using Linux kickstart or autoYaST, or to cloud machines that are provisioned in OpenStack by using kickstart.

By default, vRealize Automation uses Dynamic Host Configuration Protocol (DHCP) to assign IP addresses to provisioned machines.

You can create network profiles to define a range of static IP addresses that you can assign to machines. You can assign network profiles to specific network paths on a reservation. Machines that are provisioned by cloning or by kickstart or autoYaST and are attached to a network path with an associated network profile are provisioned with an assigned static IP address. For provisioning with a static IP address assignment, you must use a customization specification.

You can assign a network profile to a vSphere machine component in a blueprint by adding an existing, on-demand NAT, or on-demand routed network component to the design canvas and selecting a network profile to which to connect the vSphere machine component. You can also assign network profiles to blueprints by using the custom property `VirtualMachine.NetworkN.ProfileName`, where *N* is the network identifier.

You can optionally use the supplied vRealize Automation IPAM or a registered and configured third-part IPAM service provider endpoint in your network profile to obtain and configure IP addresses. For information about external IPAM requirements, see Checklist For Providing Third-Party IPAM Provider Support.

When you select a third-party IPAM service provider endpoint in a network profile, vRealize Automation retrieves IP ranges from the registered external IPAM provider endpoint, such as Infoblox. It then allocates IP values from that endpoint. The specified range subnet mask is used to allocate subnets from the IP block.

If you specify a network profile in a reservation and a blueprint, the blueprint values take precedence.

Understanding CSV File Format for Importing Network Profile IP Addresses

You can import IP address network ranges to a vRealize Automation network profile by using a properly formatted CSV file.

The CSV file entries must adhere to the following format.

CSV Field	Description
`ip_address`	An IP address in IPv4 format.
`machine_name`	Name of a managed machine in vRealize Automation. If the field is empty, the default is no name. If the field is empty, the `status` field value cannot be Allocated.
`status`	Allocated or Unallocated, case-sensitive. If the field is empty, the default value is Unallocated. If the status is Allocated, the `machine_name` field cannot be empty.
`NIC_offset`	A non-negative integer. The NIC offset indicates which virtual machine NIC the IP address is assigned to. If a virtual machine allocates more than one IP addresses for different NICs, there is an IP address entry for every NIC that contains the corresponding NIC offset. A setting of 0 specifies no offset.

The following example entry shows a machine IP address of 100.10.100.1, a name of mymachine01, a status of allocated, and no NIC offset.

```
100.10.100.1,mymachine01,Unallocated,0
```

Scenario: Import IP Addresses To a Network Profile From a CSV File

You can add IP addresses to a network profile range by importing a properly formatted CSV file. You can also change the addresses in the network profile range by editing the range in vRealize Automation or by importing a changed or different CSV file.

You can add or change the IP addresses in a network profile range by importing from a CSV file or by entering values manually. Alternatively, you can allow a third-party IPAM provider to supply IP addresses.

- Import an initial range of IP addresses into a vRealize Automation network profile.
- Apply the imported values to create our first named network range in the network profile.
- Delete one or more IP addresses from the network range vRealize Automation.
- Import a changed or different CSV file to examine how the network range values are changed.

You cannot use the **Import from CSV** option for network profiles that use a third-party IPAM endpoint because the IP addresses are managed by the third-party IPAM provider, not by vRealize Automation.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Create a CSV file containing IP addresses for import to a network range. See Create an External Network Profile by Using A Third-Party IPAM Provider and Understanding CSV File Format for Importing Network Profile IP Addresses.

Procedure

1. Select **Infrastructure > Reservations > Network Profiles**.
2. Click **New** and select a network profile type from the drop-down menu.

   For this example, select *External*.
3. Enter `My Network Profile with CSV` in the **Name** text box.
4. Enter `Testing network range IP addresses with CSV` in the **Description** text box.

   The CSV file import option applies to settings on the **Network Ranges** and **IP Addresses** tab pages.
5. (Optional) Select a configured IPAM endpoint if you have one available. If not, skip this step.
6. Enter an appropriate IP address value in the **Subnet mask** and **Gateway** text boxes.
7. Click the **DNS** tab.
8. Enter applicable information such as a DNS suffix and click the **Network Ranges** tab.

   The **Import from CVS** option is available when you click the **Network Ranges** tab.
9. To enter a new network range name and IP address range manually, click **New** or to import IP information from a properly formatted CSV file, click **Import from CSV**.

   - Click **New**.

     a. Enter a network range name.

     b. Enter a network range description.

     c. Enter the start IP address of the range.

     d. Enter the end IP address of the range.

   - Click **Import from CSV**.

     a. Browse to and select the CSV file or move the CSV file into the **Import from CSV** dialog box.

A row in the CSV file has the format *ip_address*, *machine_name*, *status*, *NIC offset*. For example:

```
100.10.100.1,mymachine01,Allocated,0
```

CSV Field	Description
`ip_address`	An IP address in IPv4 format.
`machine_name`	Name of a managed machine in vRealize Automation. If the field is empty, the default is no name. If the field is empty, the `status` field value cannot be Allocated.
`status`	Allocated or Unallocated, case-sensitive. If the field is empty, the default value is Unallocated. If the status is Allocated, the `machine_name` field cannot be empty.
`NIC_offset`	A non-negative integer. The NIC offset indicates which virtual machine NIC the IP address is assigned to. If a virtual machine allocates more than one IP addresses for different NICs, there is an IP address entry for every NIC that contains the corresponding NIC offset. A setting of 0 specifies no offset.

b Click **Apply**.

10 Click **OK**.

The IP addresses in the range appear in the defined IP addresses list.

The IP addresses appear when you click **Apply** or after you save and then edit the network profile.

11 To display the IP address data for the specified range address space, click the **IP Addresses** tab.

If you imported the IP address information from a CSV file, the range name is generated as *Imported from CSV*.

12 (Optional) To filter IP address entries, select an IP address from the **Network range** drop-down menu.

You can display information about the defined network ranges, the network ranges that are imported from a CSV file, or a named network range.

What to do next

If you import IP addresses from a CSV file again, the previous IP addresses are replaced with the information from the imported CSV file.

Creating an External Network Profile For an Existing Network

You can create external network profiles to specify network settings to configure existing networks for provisioning machines, including the configuration of NSX Edge devices to be used during provisioning.

You can use the supplied vRealize Automation IPAM provider endpoint or a third-party IPAM provider endpoint, such as Infoblox, that you have registered in vRealize Orchestrator.

Create an External Network Profile By Using the Supplied IPAM Endpoint

You can create an external network profile to define network properties and a range of static IP addresses for use when provisioning machines on an existing network.

You can define one or more network ranges of static IP addresses in the network profile for use in provisioning a machine. If you do not specify a range, you can use a network profile as a network reservation policy to select a reservation network path for a virtual machine network card (vNIC).

For information about creating an external network profile and using an external IPAM provider endpoint, see Create an External Network Profile by Using A Third-Party IPAM Provider.

Procedure

1. Specify External Network Profile Information By Using the Supplied IPAM Endpoint

   An external network profile identifies network properties and settings for an existing network. An external network profile is a requirement of NAT and routed network profiles.

2. Configure External Network Profile IP Ranges By Using the Supplied IPAM Endpoint

   You can define one or more network ranges of static IP addresses in the network profile for use in provisioning a machine. If you do not specify a range, you can use a network profile as a network reservation policy to select a reservation network path for a virtual machine network card (vNIC).

What to do next

You can assign a network profile to a network path in a reservation or a blueprint architect can specify the network profile in a blueprint. You can use the external network profile when you create an on-demand NAT or routed network profile.

Specify External Network Profile Information By Using the Supplied IPAM Endpoint

An external network profile identifies network properties and settings for an existing network. An external network profile is a requirement of NAT and routed network profiles.

For information about how you can create an external network profile by obtaining IPAM address information from a registered third-party IPAM endpoint such as Infoblox, see Checklist For Providing Third-Party IPAM Provider Support and Create an External Network Profile by Using A Third-Party IPAM Provider. Use the following procedure to create a network profile by using the VMware internal IPAM endpoint.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.

Procedure

1. Select **Infrastructure > Reservations > Network Profiles**.
2. Click **New** and select **External** from the drop-down menu.
3. Enter a name and, optionally, a description.

4 Accept the default **IPAM endpoint** value for the supplied **vRealize Automation IPAM** endpoint.

5 Enter an IP subnet mask in the **Subnet mask** text box.

  The subnet mask specifies the size of the entire routable address space that you want to define for your network profile.

  For example, enter 255.255.0.0.

6 Enter a routed gateway address, for example 10.10.110.1, in the **Gateway** text box.

  The gateway IP address defined in the network profile is assigned to the NIC during allocation. The gateway is required for NAT network profiles.

  For NSX-T, the DHCP server default gateway matches the NAT one-to-many default gateway. The IP pool default gateway matches the NAT one-to-many default gateway in vRealize Automation.

  If no value is assigned in the **Gateway** text box in the network profile, then you must use the `VirtualMachine.Network0.Gateway` custom property to assign a gateway.

7 Click the **DNS** tab.

8 Enter DNS and WINS values as needed.

  Use DNS values for the name registration and resolution. The values are optional for internal IPAM. The values are provided by the third-party IPAM provider for external IPAM.

  a (Optional) Enter a **Primary DNS** server value.

  b (Optional) Enter a **Secondary DNS** server value.

  c (Optional) Enter a **DNS suffixes** value.

  d (Optional) Enter a **DNS search suffixes** value.

  e (Optional) Enter a **Preferred WINS** server value.

  f (Optional) Enter an **Alternate WINS** server value.

What to do next

You can configure IP ranges for static IP addresses. See Configure External Network Profile IP Ranges By Using the Supplied IPAM Endpoint.

Configure External Network Profile IP Ranges By Using the Supplied IPAM Endpoint
You can define one or more network ranges of static IP addresses in the network profile for use in provisioning a machine. If you do not specify a range, you can use a network profile as a network reservation policy to select a reservation network path for a virtual machine network card (vNIC).

You can define IP range values manually, from an imported CSV file, or by using IP addresses supplied by an external IPAM provider. You can combine manually defined IP ranges and IP addresses imported via CSV. For example, you can define some ranges by using the user interface and others by importing from a CSV file.

If you import from a CSV file a second time, regardless of the CSV file name, the IP ranges imported from the previous CSV file import are erased and the new IP range information is added. Thus the previous import is overwritten when you import a second or more time. You can repeat the process of updating a CSV file and re-importing that CSV file into the network profile indefinitely.

If an external network profile does not have IP ranges defined, you can use it to specify which network is picked for a virtual network card (vNIC). If you are using the existing network profile in a routed or NAT network profile, it must have at least one static IP range.

Prerequisites

Specify External Network Profile Information By Using the Supplied IPAM Endpoint.

Procedure

1 Click the **Network Ranges** tab.

2 To enter a new network range name and IP address range manually, click **New** or to import IP information from a properly formatted CSV file, click **Import from CSV**.

   - Click **New**.

     a Enter a network range name.

     b Enter a network range description.

     c Enter the start IP address of the range.

     d Enter the end IP address of the range.

   - Click **Import from CSV**.

     a Browse to and select the CSV file or move the CSV file into the **Import from CSV** dialog box.

       A row in the CSV file has the format *ip_address*, *machine_name*, *status*, *NIC offset*. For example:

       ```
       100.10.100.1,mymachine01,Allocated,0
       ```

CSV Field	Description
ip_address	An IP address in IPv4 format.
machine_name	Name of a managed machine in vRealize Automation. If the field is empty, the default is no name. If the field is empty, the status field value cannot be Allocated.
status	Allocated or Unallocated, case-sensitive. If the field is empty, the default value is Unallocated. If the status is Allocated, the machine_name field cannot be empty.
NIC_offset	A non-negative integer. The NIC offset indicates which virtual machine NIC the IP address is assigned to. If a virtual machine allocates more than one IP addresses for different NICs, there is an IP address entry for every NIC that contains the corresponding NIC offset. A setting of 0 specifies no offset.

   b Click **Apply**.

3 Click **OK**.

   The IP addresses in the range appear in the defined IP addresses list.

   The IP addresses appear when you click **Apply** or after you save and then edit the network profile.

4 To display the IP address data for the specified range address space, click the **IP Addresses** tab.

   If you imported the IP address information from a CSV file, the range name is generated as *Imported from CSV*.

5 (Optional) To filter IP address entries, select an IP address from the **Network range** drop-down menu.

   You can display information about the defined network ranges, the network ranges that are imported from a CSV file, or a named network range.

6 (Optional) To filter IP addresses that match the IP status, select a status type from the **IP status** drop-down menu.

   For IP addresses that are in an expired or destroyed state, you can click **Reclaim** to make them available for allocation. You must save the profile for the reclamation to take effect. It may take a minute for the status column to update from `Expired` or `Destroyed` to `Allocated`.

7 To complete the network profile, click **OK**.

Results

You can assign a network profile to a network path in a reservation or a blueprint architect can specify the network profile in a blueprint. If you created an external network profile, you can use the external network profile when creating a NAT or routed network profile.

Create an External Network Profile by Using A Third-Party IPAM Provider

You can use a third-party IPAM provider solution that you have imported, configured, and registered in vRealize Orchestrator to obtain IP addresses from that third-party provider.

You can create an external network profile that uses a registered third-party IPAM solution provider endpoint to obtain gateway, subnet mask, and DHCP/WINS settings.

You can define one or more network ranges of static IP addresses in the network profile for use in provisioning a machine. If you do not specify a range, you can use a network profile as a network reservation policy to select a reservation network path for a virtual machine network card (vNIC).

For information about how to create an external network profile without using an IPAM provider or by using the supplied internal IPAM provider endpoint, see Create an External Network Profile By Using the Supplied IPAM Endpoint.

Procedure

1 Specify External Network Profile Information By Using a Third-Party IPAM Endpoint

   An external network profile identifies network properties and settings for an existing network. An external network profile is a requirement of NAT and routed network profiles. If you registered and configured an IPAM endpoint in vRealize Orchestrator, you can specify that IP address information be supplied by an IPAM provider.

2 Configure External Network Profile IP Ranges By Using a Third-Party IPAM Endpoint

   You can define one or more network ranges of static IP addresses in the network profile for use in provisioning a machine. If you do not specify a range, you can use a network profile as a network reservation policy to select a reservation network path for a virtual machine network card (vNIC).

What to do next

You can assign a network profile to a network path in a reservation or a blueprint architect can specify the network profile in a blueprint. You can use the external network profile when you create an on-demand NAT or routed network profile.

Specify External Network Profile Information By Using a Third-Party IPAM Endpoint
An external network profile identifies network properties and settings for an existing network. An external network profile is a requirement of NAT and routed network profiles. If you registered and configured an IPAM endpoint in vRealize Orchestrator, you can specify that IP address information be supplied by an IPAM provider.

Prerequisites

- Verify that you imported and configured an external IPAM provider plug-in in vRealize Orchestrator and registered the IPAM provider endpoint type in vRealize Orchestrator. In this example, the supported external IPAM solution provider is Infoblox. See Checklist For Providing Third-Party IPAM Provider Support.
- Create a Third-Party IPAM Provider Endpoint.
- Configure the vRealize Orchestrator Appliance with the registered IPAM Endpoint workflow as the standalone Orchestrator in the global tenant (administrator@vsphere.local).
- Log in to vRealize Automation as a **fabric administrator**.

Procedure

1 Select **Infrastructure > Reservations > Network Profiles**.

2 Click **New** and select **External** from the drop-down menu.

3 Enter a name and, optionally, a description.

4 If you have configured one or more third-party IPAM provider endpoints, select a third-party IPAM endpoint in the **IPAM endpoint** drop-down menu.

When you select a third-party IPAM provider endpoint that you have registered in vRealize Orchestrator, you obtain IP addresses from the specified IPAM service provider.

What to do next

You can now define network ranges for IP addresses to complete the network profile definition.

Configure External Network Profile IP Ranges By Using a Third-Party IPAM Endpoint

You can define one or more network ranges of static IP addresses in the network profile for use in provisioning a machine. If you do not specify a range, you can use a network profile as a network reservation policy to select a reservation network path for a virtual machine network card (vNIC).

You can define IP ranges by using the IP addresses that are supplied by a third-party IPAM provider.

vRealize Automation only saves external IPAM range IDs in the database, not range details. If you edit a network profile on this page or on a blueprint, vRealize Automation calls the IPAM service to get range details based on the selected range IDs.

Note There is a known issue with some third-party IPAM providers in which a query can time out when returning network ranges, resulting in an empty list. As a workaround, you can provide search criteria to avoid the timeout and obtain the network range information.

For example, depending on your IPAM provider, you may be able to add a property named `VLAN` to each network in the IPAM provider application and assign a value to that property, such as 4. You could then filter on the property and value, for example `VLAN=4`, in the **Select Network Range** text box on the vRealize Automation network profile page.

As an alternative, you can increase the timeout setting by using the following procedure:

1 On each of the vRealize Automation appliance cnodes, open the `/etc/vcac/webapps/o11n-gateway-service/WEB-INF/classes/META-INF/spring/root/o11n-gateway-service-context.xml` file.

2 Change the timeout value from 30 seconds to a higher number.

3 Restart vcac-server by entering `service vcac-server restart`.

Prerequisites

Specify External Network Profile Information By Using a Third-Party IPAM Endpoint.

Procedure

1 To create a network range, or select an existing network range, click the **Network Ranges** tab.

2 Select an address space from the list of all addresses spaces that are available for the endpoint from the **Address space** drop-down menu.

3 Click **Add** and select one or more available network ranges for the specified address space.

Selecting a network range may result in an empty list when using a third-party IPAM provider. For details, see Knowledge Base article 2148656 at http://kb.vmware.com/kb/2148656.

4 Click **OK**.

The IP addresses in the range appear in the defined IP addresses list.

The IP addresses appear when you click **Apply** or after you save and then edit the network profile.

5 To complete the network profile, click **OK**.

What to do next

You can assign a network profile to a network path in a reservation or a blueprint architect can specify the network profile in a blueprint.

Creating a Routed Network Profile For an On-Demand Network

You can create an on-demand routed network profile that uses either the supplied vRealize Automation IPAM endpoint or a properly configured and registered third-party IPAM endpoint.

A routed network profile represent routeable IP space that is divided across multiple networks. Each new routed network allocates the next available subnet from the routable IP space. A routed network can access all other routed networks that use the same network profile. Each routed subnet can access all other subnets created by the same network profile.

For a third-party IPAM provider, the routable IP space is created and managed by the third-party IPAM provider. The network administrator uses a third-party IPAM provider to define a routable IP space and create an IP block for it. You can select one or more IP blocks retrieved from the third-party IPAM provider when you create or edit a routed network profile.

When a new instance of a routed network profile is allocated from the third-party IPAM provider, vRealize Automation calls the provider to reserve the next available subnet and creates a range, using IP blocks that are determined by the routed network profile and the subnet size. The resulting range is used to allocate IP addresses for machines that are assigned to the routed network in the same deployment.

Create a Routed Network Profile By Using the Supplied IPAM Endpoint

When using a routed network profile with the supplied IPAM endpoint, you can define a routable IP space and available subnets for an on-demand routed network.

Using the supplied vRealize Automation IPAM endpoint, you can assign ranges of static IP addresses and a base IP address to the routed network profile.

You can use IP ranges obtained from the supplied VMware IPAM endpoint or from a third-party IPAM service provider endpoint that you have registered and configured in vRealize Orchestrator, such as Infoblox IPAM. An IP range is created from an IP block during allocation.

Procedure

1. Specify Routed Network Profile Information with the vRealize Automation IPAM Endpoint

   The network profile information identifies the routed network properties, its underlying external network profile, and other values used in provisioning the network when using the supplied IPAM endpoint.

2. Configure Routed Network Profile IP Ranges with the vRealize Automation IPAM Endpoint

   You can define one or more ranges of static IP addresses for use in provisioning a network.

Specify Routed Network Profile Information with the vRealize Automation IPAM Endpoint
The network profile information identifies the routed network properties, its underlying external network profile, and other values used in provisioning the network when using the supplied IPAM endpoint.

If you want to create a routed network profile by using a third-party IPAM endpoint, see Specify Routed Network Profile Information with a Third-Party IPAM Endpoint.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Create an external network profile. See Create an External Network Profile By Using the Supplied IPAM Endpoint.

Procedure

1. Select **Infrastructure > Reservations > Network Profiles**.
2. Click **New** and select **Routed** from the drop-down menu.
3. Enter a name and, optionally, a description.
4. Accept the default **IPAM endpoint** value for the supplied **vRealize Automation IPAM** endpoint.
5. Select an existing external network profile from the **External Network Profile** drop-down menu.
6. Enter the subnet mask in the **Subnet mask** text box that is associated with the external network profile.

   The subnet mask specifies the size of the entire routable address space to define for the network profile.

   For example, enter 255.255.0.0.

7. Select a value in the **Range subnet mask** text box drop-down menu.

   For example, enter 255.255.255.0.

The range subnet mask defines how you want to partition the network space into individual address blocks. The blocks are allocated to every deployment instance of the network profile.

For each deployment that uses a routed network profile, you use a range. The number of available routed ranges is equal to the subnet mask divide by the range subnet mask, for example 255.255.0.0/255.255.255.0 = 256.

8 Enter the first available IP address in the **Base IP** text box.

   This option is not available for third-party endpoints.

   For example, enter 120.120.0.1.

9 Click the **DNS** tab.

10 Enter DNS and WINS values as needed.

   Use DNS values for the name registration and resolution. The values are optional for internal IPAM. The values are provided by the third-party IPAM provider for external IPAM.

   a (Optional) Enter a **Primary DNS** server value.

   b (Optional) Enter a **Secondary DNS** server value.

   c (Optional) Enter a **DNS suffixes** value.

   d (Optional) Enter a **DNS search suffixes** value.

   e (Optional) Enter a **Preferred WINS** server value.

   f (Optional) Enter an **Alternate WINS** server value.

What to do next

Configure Routed Network Profile IP Ranges with the vRealize Automation IPAM Endpoint.

Configure Routed Network Profile IP Ranges with the vRealize Automation IPAM Endpoint

You can define one or more ranges of static IP addresses for use in provisioning a network.

During provisioning, every new routed network allocates the next available range and uses it as its IP space.

Prerequisites

Specify Routed Network Profile Information with the vRealize Automation IPAM Endpoint.

Procedure

1 To create a network range, or select an existing network range, click the **Network Ranges** tab.

2 Click **Generate Ranges** to generate network ranges based on the subnet mask, range subnet mask, and base IP address information that you entered on the General tab.

   Starting with the base IP address, vRealize Automation generates ranges based on the range subnet mask.

For example, vRealize Automation generates ranges of 255 IP ranges if the subnet mask is 255.255.0.0 and the range subnet mask is 255.255.255.0 using the name Range1 through Range*n*.

3 Click **OK**.

Create a Routed Network Profile By Using a Third-Party IPAM Endpoint

When you use a routed network profile with a third-party IPAM endpoint, routable IP space is created and managed by the third-party IPAM provider.

When you use a third-party IPAM endpoint in your routed network profile, the provider creates new IP ranges for each instance of the on-demand network.

You can use IP ranges obtained from the supplied VMware IPAM endpoint or from a third-party IPAM service provider endpoint that you have registered and configured in vRealize Orchestrator, such as Infoblox IPAM. An IP range is created from an IP block during allocation.

Procedure

1 Specify Routed Network Profile Information with a Third-Party IPAM Endpoint

The network profile information identifies the routed network properties, its underlying external network profile, and other values used in provisioning the network when using a third-party IPAM endpoint.

2 Configure Routed Network Profile IP Ranges with a Third-Party IPAM Endpoint

You can manage one or more named ranges of static IPv4 network addresses for use in provisioning a network.

Specify Routed Network Profile Information with a Third-Party IPAM Endpoint

The network profile information identifies the routed network properties, its underlying external network profile, and other values used in provisioning the network when using a third-party IPAM endpoint.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Create an external network profile. See Create an External Network Profile By Using the Supplied IPAM Endpoint or Create an External Network Profile by Using A Third-Party IPAM Provider.
- Create and configure a third-party IPAM endpoint. See Create a Third-Party IPAM Provider Endpoint.

Procedure

1 Select **Infrastructure > Reservations > Network Profiles**.

2 Click **New** and select **Routed** from the drop-down menu.

3 Enter a name and, optionally, a description.

4 If you have configured one or more third-party IPAM provider endpoints, select a third-party IPAM endpoint in the **IPAM endpoint** drop-down menu.

   When you select a third-party IPAM provider endpoint that you have registered in vRealize Orchestrator, you obtain IP addresses from the specified IPAM service provider.

5 Select an existing external network profile from the **External Network Profile** drop-down menu.

   Only external network profiles that are configured to use the specified IPAM endpoint are listed and available to select.

6 To determine how many network subnets to create, select a value in the **Range subnet mask** text box drop-down menu.

   For example, enter 255.255.255.0.

   The range subnet mask defines how you want to partition that space into individual address blocks that are allocated to every deployment instance of that network profile. When choosing a value for the range subnet mask, consider the number of deployments that you expect to use the routed network.

   A range is used for each deployment that uses a routed network profile. The number of available routed ranges is equal to the subnet mask divide by the range subnet mask, for example 255.255.0.0/255.255.255.0 = 256.

7 To define an address space and manage one or more named ranges of static IPv4 network addresses, click the **IP Blocks** tab.

   The available IP blocks are the source for IP ranges that you create or allocate for on-demand routing.

What to do next

Configure Routed Network Profile IP Ranges with a Third-Party IPAM Endpoint.

Configure Routed Network Profile IP Ranges with a Third-Party IPAM Endpoint
You can manage one or more named ranges of static IPv4 network addresses for use in provisioning a network.

During provisioning, each new routed network allocates the next available range and uses that allocated range as its IP space. The IP blocks are obtained from the third-party IPAM provider. During provisioning, a routed network is allocated from the block with a subnet mask that matches the provided range subnet mask.

Prerequisites

Specify Routed Network Profile Information with a Third-Party IPAM Endpoint.

Procedure

1 To limit the available IP blocks that are available for provisioning, select an address space from the **Address space** drop-down menu.

   You cannot select an address space after you add IP blocks. A routed network profile cannot span more than one address space.

2 Add one or more IP blocks or IPAM provider ranges.

   The IP blocks are retrieved from the third-party IPAM provider.

   Selecting a network range may result in an empty list when using a third-party IPAM provider. For details, see Knowledge Base article 2148656 at http://kb.vmware.com/kb/2148656.

   a Click **Add**.

   b Click **Search**.

   c Enter the search syntax or select IP blocks from the drop-down menu.

   d Click **OK**.

3 Click **Apply**.

4 Click **OK**.

Creating a NAT Network Profile For an On-Demand Network

You can create an on-demand NAT network profile that uses either the supplied vRealize Automation IPAM endpoint or a properly configured and registered third-party IPAM endpoint.

Create a NAT Network Profile By Using the Supplied IPAM Endpoint

You can create an on-demand NSX NAT network profile relative to an external network profile. When using the supplied vRealize Automation IPAM endpoint, you can assign ranges of static IP and DHCP addresses to the NAT network profile.

NAT networks use one set of IP addresses for external communication and another set of IP addresses for internal communication. External IP addresses are allocated from an external network profile and internal NAT IP addresses are defined by a NAT network profile. When you provision a new NAT network, a new instance of the NAT network profile is created and used to allocate machine IP addresses.

You can use IP ranges obtained from the supplied VMware IPAM endpoint or from a third-party IPAM service provider endpoint that you have registered and configured in vRealize Orchestrator, such as Infoblox IPAM. An IP range is created from an IP block during allocation.

For a NAT one-to-many network, you can define NAT rules that can be configured when you add a NAT network component to the blueprint. You can change a NAT rule when you edit the NAT network in a deployment.

Procedure

1 Specify NAT Network Profile Information with the vRealize Automation IPAM Endpoint

  The network profile identifies the on-demand NAT network properties, underlying external network profile, NAT type, and other values used to provision the network by using the embedded vRealize Automation IPAM.

2 Configure NAT Network Profile IP Ranges with the vRealize Automation IPAM Endpoint

  You can define one or more ranges of static IP addresses for use in provisioning a network.

Specify NAT Network Profile Information with the vRealize Automation IPAM Endpoint
The network profile identifies the on-demand NAT network properties, underlying external network profile, NAT type, and other values used to provision the network by using the embedded vRealize Automation IPAM.

If you want to create a NAT network profile that uses a third-party IPAM endpoint, see Specify NAT Network Profile Information with a Third-Party IPAM Endpoint.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Create an external network profile. See Create an External Network Profile By Using the Supplied IPAM Endpoint.

Procedure

1 Select **Infrastructure > Reservations > Network Profiles**.

2 Click **New** and select **NAT** from the drop-down menu.

3 Enter a name and, optionally, a description.

4 Accept the default **IPAM endpoint** value for the supplied **vRealize Automation IPAM** endpoint.

5 Select an existing external network profile from the **External Network Profile** drop-down menu.

6 Select a one-to-one or one-to-many network address translation type from the **NAT type** drop-down menu.

Option	Description
One-to-One	Assign an external static IP address to each network adapter. Every machine can access the external network and is accessible from the external network. All external IP addresses that are assigned to an NSX edge uplink must be part of the same subnet. When using NAT one-to-one in vRealize Automation, the corresponding external network profile must contain only IP ranges that exist within a single subnet.
One-to-Many	One external IP address is shared among all machines on the network. An internal machine can have either DHCP or static IP addresses. Every machine can access the external network, but no machine is accessible from the external network. Selecting this option enables the **Enabled** check box in the DHCP group. For NSX for vSphere, the NAT one-to-many translation type allows you to define NAT rules when you add a NAT network component to a blueprint. NSX for vSphere supports NAT one-to-one and NAT one-to-many networks, but NSX-T only supports NAT one-to-many.

7 Enter an IP subnet mask in the **Subnet mask** text box.

The subnet mask specifies the size of the entire routable address space that you want to define for your network profile.

For example, enter 255.255.0.0.

8 Enter a routed gateway address, for example 10.10.110.1, in the **Gateway** text box.

The gateway IP address defined in the network profile is assigned to the NIC during allocation. The gateway is required for NAT network profiles.

For NSX-T, the DHCP server default gateway matches the NAT one-to-many default gateway. The IP pool default gateway matches the NAT one-to-many default gateway in vRealize Automation.

If no value is assigned in the **Gateway** text box in the network profile, then you must use the `VirtualMachine.Network0.Gateway` custom property to assign a gateway.

9 (Optional) In the DHCP group, select the **Enabled** check box and enter the **IP range start** and **IP range end** values.

You can select the check box only if you set the NAT type to one-to-many.

For NSX-T, the first IP in the IP pool range matches the DHCP server IP address defined the `<FirstIpInPool>/<subnetMaskOfNat>` setting. The IP pool in NSX-T starts with the second IP address.

10 (Optional) Set a DHCP lease time to define how long a machine can use an IP address.

11 Click the **DNS** tab.

12 Enter DNS and WINS values as needed.

   Use DNS values for the name registration and resolution. The values are optional for internal IPAM. The values are provided by the third-party IPAM provider for external IPAM.

   a (Optional) Enter a **Primary DNS** server value.

   b (Optional) Enter a **Secondary DNS** server value.

   c (Optional) Enter a **DNS suffixes** value.

   d (Optional) Enter a **DNS search suffixes** value.

   e (Optional) Enter a **Preferred WINS** server value.

   f (Optional) Enter an **Alternate WINS** server value.

What to do next

Configure NAT Network Profile IP Ranges with the vRealize Automation IPAM Endpoint.

Configure NAT Network Profile IP Ranges with the vRealize Automation IPAM Endpoint

You can define one or more ranges of static IP addresses for use in provisioning a network.

You cannot overlap the start and end network range IP addresses with the DHCP addresses. If you attempt to save a profile that contains address ranges that overlap, vRealize Automation displays a validation error.

Prerequisites

Specify NAT Network Profile Information with the vRealize Automation IPAM Endpoint.

Procedure

1 To create a network range, or select an existing network range, click the **Network Ranges** tab.

2 To enter a new network range name and IP address range manually, click **New** or to import IP information from a properly formatted CSV file, click **Import from CSV**.

   - Click **New**.

     a Enter a network range name.

     b Enter a network range description.

     c Enter the start IP address of the range.

     d Enter the end IP address of the range.

   - Click **Import from CSV**.

     a Browse to and select the CSV file or move the CSV file into the **Import from CSV** dialog box.

A row in the CSV file has the format *ip_address*, *machine_name*, *status*, *NIC offset*. For example:

```
100.10.100.1,mymachine01,Allocated,0
```

CSV Field	Description
`ip_address`	An IP address in IPv4 format.
`machine_name`	Name of a managed machine in vRealize Automation. If the field is empty, the default is no name. If the field is empty, the `status` field value cannot be Allocated.
`status`	Allocated or Unallocated, case-sensitive. If the field is empty, the default value is Unallocated. If the status is Allocated, the `machine_name` field cannot be empty.
`NIC_offset`	A non-negative integer. The NIC offset indicates which virtual machine NIC the IP address is assigned to. If a virtual machine allocates more than one IP addresses for different NICs, there is an IP address entry for every NIC that contains the corresponding NIC offset. A setting of 0 specifies no offset.

b Click **Apply**.

3 Click **OK**.

The IP addresses in the range appear in the defined IP addresses list.

The IP addresses appear when you click **Apply** or after you save and then edit the network profile.

4 To display the IP addresses for the named network range, click the **IP Addresses** tab.

5 (Optional) To filter IP address entries, select an IP address from the **Network range** drop-down menu.

You can display information about the defined network ranges, the network ranges that are imported from a CSV file, or a named network range.

6 (Optional) To filter IP addresses that match the IP status, select a status type from the **IP status** drop-down menu.

For IP addresses that are in an expired or destroyed state, you can click **Reclaim** to make them available for allocation. You must save the profile for the reclamation to take effect. It may take a minute for the status column to update from `Expired` or `Destroyed` to `Allocated`.

7 Click **OK**.

Create a NAT Network Profile By Using a Third-Party IPAM Endpoint in vRealize Automation

You can create an on-demand NSX NAT network profile relative to an external network profile in vRealize Automation. When using an NSX NAT network profile with a third-party IPAM provider, IP space is created and managed by the third-party IPAM provider.

When you use a third-party IPAM endpoint in your NAT network profile, the provider creates new IP ranges for each instance of the on-demand network. An internal set of IP addresses defined with one or more ranges is created in the third-party IPAM provider endpoint for every instance of the network. The IP ranges allocate IP addresses for the machines on the network in the same deployment. Because there cannot be duplicate IP addresses defined within a single address space, a new address space is created by the provider for each instance of the network. When a NAT network is destroyed, its ranges are destroyed in the IPAM provider endpoint and in the new address space.

You can use IP ranges obtained from the supplied VMware IPAM endpoint or from a third-party IPAM service provider endpoint that you have registered and configured in vRealize Orchestrator, such as Infoblox IPAM. An IP range is created from an IP block during allocation.

For a NAT one-to-many network, you can define NAT rules that can be configured when you add a NAT network component to the blueprint. You can change a NAT rule when you edit the NAT network in a deployment.

Procedure

1 Specify NAT Network Profile Information with a Third-Party IPAM Endpoint

  The network profile information identifies the NAT network properties, its underlying external network profile, and other values used in provisioning the network when using a third-party IPAM endpoint.

2 Configure NAT Network Profile IP Ranges with a Third-Party IPAM Endpoint

  You can define one or more IP address ranges for use in provisioning a network by using NAT.

Specify NAT Network Profile Information with a Third-Party IPAM Endpoint

The network profile information identifies the NAT network properties, its underlying external network profile, and other values used in provisioning the network when using a third-party IPAM endpoint.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Create an external network profile. See Create an External Network Profile By Using the Supplied IPAM Endpoint or Create an External Network Profile by Using A Third-Party IPAM Provider.
- Create and configure a third-party IPAM endpoint. See Create a Third-Party IPAM Provider Endpoint.

Procedure

1 Select **Infrastructure > Reservations > Network Profiles**.

2 Click **New** and select **NAT** from the drop-down menu.

3 Enter a name and, optionally, a description.

4 If you have configured one or more third-party IPAM provider endpoints, select a third-party IPAM endpoint in the **IPAM endpoint** drop-down menu.

When you select a third-party IPAM provider endpoint that you have registered in vRealize Orchestrator, you obtain IP addresses from the specified IPAM service provider.

5 Select an existing external network profile from the **External Network Profile** drop-down menu.

Only external network profiles that are configured to use the specified IPAM endpoint are listed and available to select.

6 Select a one-to-one or one-to-many network address translation type from the **NAT type** drop-down menu.

Option	Description
One-to-One	Assign an external static IP address to each network adapter. Every machine can access the external network and is accessible from the external network. All external IP addresses that are assigned to an NSX edge uplink must be part of the same subnet. When using NAT one-to-one in vRealize Automation, the corresponding external network profile must contain only IP ranges that exist within a single subnet.
One-to-Many	One external IP address is shared among all machines on the network. An internal machine can use only static IP addresses. Every machine can access the external network, but no machine is accessible from the external network. DHCP is not supported when using NAT with a third-party IPAM provider. For NSX for vSphere, the NAT one-to-many translation type allows you to define NAT rules when you add a NAT network component to a blueprint. NSX for vSphere supports NAT one-to-one and NAT one-to-many networks, but NSX-T only supports NAT one-to-many.

7 Enter an IP subnet mask in the **Subnet mask** text box.

The subnet mask specifies the size of the entire routable address space that you want to define for your network profile.

For example, enter 255.255.0.0.

8 Enter a routed gateway address, for example 10.10.110.1, in the **Gateway** text box.

The gateway IP address defined in the network profile is assigned to the NIC during allocation. The gateway is required for NAT network profiles.

For NSX-T, the DHCP server default gateway matches the NAT one-to-many default gateway. The IP pool default gateway matches the NAT one-to-many default gateway in vRealize Automation.

If no value is assigned in the **Gateway** text box in the network profile, then you must use the `VirtualMachine.Network0.Gateway` custom property to assign a gateway.

9 Click the **DNS** tab.

10 Enter DNS and WINS values as needed.

   Use DNS values for the name registration and resolution. The values are optional for internal IPAM. The values are provided by the third-party IPAM provider for external IPAM.

   a (Optional) Enter a **Primary DNS** server value.

   b (Optional) Enter a **Secondary DNS** server value.

   c (Optional) Enter a **DNS suffixes** value.

   d (Optional) Enter a **DNS search suffixes** value.

   e (Optional) Enter a **Preferred WINS** server value.

   f (Optional) Enter an **Alternate WINS** server value.

What to do next

Configure NAT Network Profile IP Ranges with a Third-Party IPAM Endpoint.

Configure NAT Network Profile IP Ranges with a Third-Party IPAM Endpoint
You can define one or more IP address ranges for use in provisioning a network by using NAT.

Prerequisites

Specify NAT Network Profile Information with a Third-Party IPAM Endpoint.

Procedure

1 To create a network range, or select an existing network range, click the **Network Ranges** tab.

2 Click **New** and define a network range.

   a Enter a network range name and description.

   b Enter the start and end IP address to define the range.

   c Click **Apply**.

3 Click **OK**.

   The IP addresses in the range appear in the defined IP addresses list.

   The IP addresses appear when you click **Apply** or after you save and then edit the network profile.

4 To display the IP addresses for the named network range, click the **IP Addresses** tab.

5 (Optional) To filter IP address entries, select an IP address from the **Network range** drop-down menu.

   You can display information about the defined network ranges, the network ranges that are imported from a CSV file, or a named network range.

6 (Optional) To filter IP addresses that match the IP status, select a status type from the **IP status** drop-down menu.

   For IP addresses that are in an expired or destroyed state, you can click **Reclaim** to make them available for allocation. You must save the profile for the reclamation to take effect. It may take a minute for the status column to update from `Expired` or `Destroyed` to `Allocated`.

7 Click **OK**.

Create a Private Network Profile for an On-Demand Network in vRealize Automation

You can create a private network for NSX for vSphere that uses the IPAM specification that is supplied with vRealize Automation.

You can create an on-demand private network profile for NSX for vSphere relative to an external network profile.

Private networks are not available for NSX-T.

Private networks are not available for third-party IPAM.

You can define one or more ranges of static IP addresses for use in provisioning a network.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.

Procedure

1 Select **Infrastructure > Reservations > Network Profiles**.

2 Click **New** and select **Private** from the drop-down menu.

3 Enter a name and, optionally, a description.

4 Accept the default **IPAM endpoint** value for the supplied **vRealize Automation IPAM** endpoint.

5 Select the tenant ID as prompted.

6 Enter an IP subnet mask in the **Subnet mask** text box.

   The subnet mask specifies the size of the entire routable address space that you want to define for your network profile.

   For example, enter 255.255.0.0.

7 Enter a routed gateway address, for example 10.10.110.1 in the **Gateway** text box.

   The gateway IP address defined in the network profile is assigned to the NIC during allocation. If no value is assigned in the **Gateway** text box in the network profile, then you must use the `VirtualMachine.Network0.Gateway` custom property to assign a gateway.

8 Click the **DNS** tab.

9 Enter DNS and WINS values as needed.

If you attempt to save a profile that contains address ranges that overlap, vRealize Automation displays a validation error.

10 To create a network range, or select an existing network range, click the **Network Ranges** tab.

11 To enter a new network range name and IP address range manually, click **New** or to import IP information from a properly formatted CSV file, click **Import from CSV**.

- Click **New**.

  a Enter a network range name.

  b Enter a network range description.

  c Enter the start IP address of the range.

  d Enter the end IP address of the range.

- Click **Import from CSV**.

  a Browse to and select the CSV file or move the CSV file into the **Import from CSV** dialog box.

  A row in the CSV file has the format *ip_address*, *machine_name*, *status*, *NIC offset*. For example:

```
100.10.100.1,mymachine01,Allocated,0
```

CSV Field	Description
`ip_address`	An IP address in IPv4 format.
`machine_name`	Name of a managed machine in vRealize Automation. If the field is empty, the default is no name. If the field is empty, the `status` field value cannot be Allocated.
`status`	Allocated or Unallocated, case-sensitive. If the field is empty, the default value is Unallocated. If the status is Allocated, the `machine_name` field cannot be empty.
`NIC_offset`	A non-negative integer. The NIC offset indicates which virtual machine NIC the IP address is assigned to. If a virtual machine allocates more than one IP addresses for different NICs, there is an IP address entry for every NIC that contains the corresponding NIC offset. A setting of 0 specifies no offset.

  b Click **Apply**.

12 Click **OK**.

The IP addresses in the range appear in the defined IP addresses list.

The IP addresses appear when you click **Apply** or after you save and then edit the network profile.

13 To display the IP addresses for the named network range, click the **IP Addresses** tab.

14 (Optional) To filter IP address entries, select an IP address from the **Network range** drop-down menu.

You can display information about the defined network ranges, the network ranges that are imported from a CSV file, or a named network range.

15 (Optional) To filter IP addresses that match the IP status, select a status type from the **IP status** drop-down menu.

For IP addresses that are in an expired or destroyed state, you can click **Reclaim** to make them available for allocation. You must save the profile for the reclamation to take effect. It may take a minute for the status column to update from `Expired` or `Destroyed` to `Allocated`.

16 Click **OK**.

Releasing IP Addresses By Destroying Provisioned Machines

When you destroy a deployment, its IP addresses are deleted. The allocated IPs, for example the IPS in a network profile range, are released and made available for subsequent provisioning.

When you destroy a machine that has a static IP address, its IP address is made available for other machines to use. Unused addresses might not be available immediately because the process to reclaim static IP addresses runs every 30 minutes.

If you are using a third-party IPAM provider, vRealize Automation deletes the associated IP addresses by using the vRealize Orchestrator workflow in the third-party IPAM provider plug-in or package.

Configuring Reservations and Reservation Policies

A vRealize Automation reservation can define policies, priorities, and quotas that determine machine placement for provisioning requests.

Reservation policies restrict machine provisioning to a subset of available reservations. Storage reservation policies allow blueprint architects to assign machine volumes to different datastores.

To provision successfully, the reservation must have sufficient available storage. The reservation's storage availability depends on:

- How much storage is available on the data store/cluster.
- How much of that storage is reserved for that data store/cluster.
- How much of that storage is already allocated in vRealize Automation

For example, even if the vCenter Server has storage available for the data store/cluster, if sufficient storage is not reserved in the reservation then provisioning fails with a "No reservation is available to allocate..." error. The allocated storage on a reservation depends on the number of VMs (regardless of their state) on that specific reservation. See the VMware Knowledge Base article *Machine XXX: No reservation is available to allocate within the group XXX. Total XX GB of storage was requested (2151030)* at http://kb.vmware.com/kb/2151030 for more information.

Reservations

You can create a vRealize Automation reservation to allocate provisioning resources in the fabric group to a specific business group.

For example, you can use reservations to specify that a share of the memory, CPU, networking, and storage resources of a single compute resource belongs to a particular business group or that certain machines be allocated to a specific business group.

You use a networking reservation policy to manage the network communications for blueprint deployments. When requesting machine provisioning, the reservation policy is used to group the reservations that can be considered for the deployment.

You cannot share reservations among multiple business groups.

Note Storage and memory that are assigned to a provisioned machine by a reservation are released when the machine to which they are assigned is deleted in vRealize Automation by the Destroy action. The storage and memory are not released if the machine is deleted in the vCenter Server.

You can create a reservation for the following machine types:

- vSphere
- vCloud Air
- vCloud Director
- Amazon EC2
- Microsoft Azure
- Hyper V (SCVMM)
- Hyper-V Stand-alone
- KVM (RHEV)
- OpenStack
- XenServer

You can configure security settings by specifying information in a reservation, blueprint, or guest agent script. If machines require a guest agent, add a security rule to the reservation or the blueprint.

Choosing a Reservation Scenario

You can create reservations to allocate resources to business groups. Depending on your scenario, the procedure to create a reservation differs.

Choose a reservation scenario based on the target endpoint type.

Each business group must have at least one reservation for its members to provision machines of that type. For example, a business group with an OpenStack reservation but not an Amazon reservation, cannot request a machine from Amazon. In this example, the business group must be allocated a reservation specifically for Amazon resources.

Table 4-15. Choosing a Reservation Scenario

Scenario	Procedure
Create a vSphere reservation.	Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer
Create a reservation to allocate resources for a vCloud Air endpoint.	Create a vCloud Air Reservation
Create a reservation to allocate resources for a vCloud Director endpoint.	Create a vCloud Director Reservation
Create a reservation to allocate resources on an Amazon resource (with or without using Amazon Virtual Private Cloud).	Create an Amazon EC2 Reservation
Create a reservation to allocate resources on an OpenStack resource.	Create an OpenStack Reservation
Create a reservation to allocate resources for Hyper-V.	Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer
Create a reservation to allocate resources for KVM.	Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer
Create a reservation to allocate resources on an OpenStack. resource.	Create an OpenStack Reservation
Create a reservation to allocate resources for SCVMM.	Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer
Create a reservation to allocate resources for XenServer.	Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer
Create a reservation to allocate resources for Microsoft Azure.	Create a Reservation for Microsoft Azure

Creating Cloud Category Reservations

A cloud category type reservation provides access to the provisioning services of a cloud service account for a particular vRealize Automation business group. Available cloud reservation types include Amazon, OpenStack, vCloud Air, and vCloud Director.

A reservation is a share of the memory, CPU, networking, and storage resources of one compute resource allocated to a particular vRealize Automation business group.

A business group can have multiple reservations on one endpoint or reservations on multiple endpoints.

The allocation model for a reservation depends on the allocation model in the associated datacenter. Available allocation models are Allocation Pool, Pay As You Go, and reservation pool. For information about allocation models, see thevCloud Director or vCloud Air documentation.

In addition to defining the share of fabric resources allocated to the business group, a reservation can define policies, priorities, and quotas that determine machine placement.

To provision successfully, the reservation must have sufficient available storage. The reservation's storage availability depends on:

- How much storage is available on the data store/cluster.

- How much of that storage is reserved for that data store/cluster.
- How much of that storage is already allocated in vRealize Automation

For example, even if the vCenter Server has storage available for the data store/cluster, if sufficient storage is not reserved in the reservation then provisioning fails with a "No reservation is available to allocate..." error. The allocated storage on a reservation depends on the number of VMs (regardless of their state) on that specific reservation. See the VMware Knowledge Base article *Machine XXX: No reservation is available to allocate within the group XXX. Total XX GB of storage was requested (2151030)* at http://kb.vmware.com/kb/2151030 for more information.
Understanding Selection Logic for Cloud Reservations
When a member of a business group creates a provisioning request for a cloud machine, vRealize Automation selects a machine from one of the reservations that are available to that business group. Cloud reservations include Amazon, OpenStack, vCloud Air, and vCloud Director.

The reservation for which a machine is provisioned must satisfy the following criteria:

- The reservation must be of the same platform type as the blueprint from which the machine was requested.
- The reservation must be enabled.
- The reservation must have capacity remaining in its machine quota or have an unlimited quota.

  The allocated machine quota includes only machines that are powered on. For example, if a reservation has a quota of 50, and 40 machines have been provisioned but only 20 of them are powered on, the reservation’s quota is 40 percent allocated, not 80 percent.
- The reservation must have the security groups specified in the machine request.
- The reservation must be associated with a region that has the machine image specified in the blueprint.
- The reservation must have sufficient unallocated memory and storage resources to provision the machine.

  In a Pay As You Go reservation, resources can be unlimited.
- For Amazon machines, the request specifies an availability zone and whether the machine is to be provisioned a subnet in a Virtual Private Cloud (VPC) or a in a non-VPC location. The reservation must match the network type (VPC or non-VPC).
- For vCloud Air or vCloud Director, if the request specifies an allocation model, the virtual datacenter associated with the reservation must have the same allocation model.
- For vCloud Director or vCloud Air, the specified organization must be enabled.
- Any blueprint templates must be available on the reservation. If the reservation policy maps to more than one resources, the templates should be public.
- If the cloud provider supports network selection and the blueprint has specific network settings, the reservation must have the same networks.

If the blueprint or reservation specifies a network profile for static IP address assignment, an IP address must be available to assign to the new machine.

- If the request specifies an allocation model, the allocation model in the reservation must match the allocation model in the request.
- If the blueprint specifies a reservation policy, the reservation must belong to that reservation policy.

  Reservation policies are a way to guarantee that the selected reservation satisfies any additional requirements for provisioning machines from a specific blueprint. For example, if a blueprint uses a specific machine image, you can use reservation policies to limit provisioning to reservations associated with the regions that have the required image.

If no reservation is available that meets all of the selection criteria, provisioning fails.

If multiple reservations meet all of the criteria, the reservation from which to provision a requested machine is determined by the following logic:

- A reservation with a lower priority value is selected before a reservation with a higher priority value.
- If multiple reservations have the same priority, the reservation with the lowest percentage of its machine quota allocated is selected.
- If multiple reservations have the same priority and quota usage, machines are distributed among reservations in round-robin fashion.

  Note While round-robin selection of network profiles is not supported, round-robin selection of networks (if any) is supported, which can be associated with different network profiles.

If multiple storage paths are available on a reservation with sufficient capacity to provision the machine volumes, storage paths are selected according to the following logic.

- A storage path with a lower priority value is selected before a storage path with a higher priority value.
- If the blueprint or request specifies a storage reservation policy, the storage path must belong to that storage reservation policy.

  If the custom property `VirtualMachine.Disk`*`N`*`.StorageReservationPolicyMode` is set to Not Exact, and no storage path with sufficient capacity is available in the storage reservation policy, then provisioning proceeds with a storage path outside the specified storage reservation policy. The default value of `VirtualMachine.Disk`*`N`*`.StorageReservationPolicyMode` is Exact.
- If multiple storage paths have the same priority, machines are distributed among storage paths by using round-robin scheduling.

Create an Amazon EC2 Reservation

You must allocate resources to machines by creating a reservation before members of a business group can request machine provisioning.

You can work with Amazon reservations for Amazon Virtual Private Cloud or Amazon non-VPC. Amazon Web Services users can create a Amazon Virtual Private Cloud to design a virtual network topology according to your specifications. If you plan to use Amazon VPC, you must assign an Amazon VPC to a vRealize Automation reservation.

See Using Amazon Virtual Private Cloud.

When you create an Amazon reservation or configure a machine component in the blueprint, you can choose from the list of security groups that are available to the specified Amazon region. Security groups are imported during data collection.

Note After you create a reservation, you cannot change the business group or compute resource associations.

For information about creating an Amazon VPC by using the AWS Management Console, see Amazon Web Services documentation.

Procedure

1. Specify Amazon Reservation Information

   Each reservation is configured for a specific business group to grant them access to request machines on a specified compute resource.

2. Specify Resource and Network Settings for Amazon Reservations

   Specify resource and network settings for provisioning machines from this vRealize Automation reservation.

3. Specify Custom Properties and Alerts for Amazon Reservations

   You can associate custom properties with a vRealize Automation reservation. You can also configure alerts to send email notifications when reservation resources are low.

Procedure

1. Specify Amazon Reservation Information.
2. Specify Resource and Network Settings for Amazon Reservations.
3. Specify Custom Properties and Alerts for Amazon Reservations.

Specify Amazon Reservation Information
Each reservation is configured for a specific business group to grant them access to request machines on a specified compute resource.

Note After you create a reservation, you cannot change the business group or compute resource associations.

You can control the display of reservations when adding, editing, or deleting by using the **Filter By Category** option on the Reservations page. Note that test agent reservations do not appear in the reservations list when filtering by category.

For information about configuring for Amazon VPC, see Using Amazon Virtual Private Cloud.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Verify that a tenant administrator created at least one business group.

  See Create a Business Group.
- Verify that a compute resource exists.
- Configure network settings.

  See Configuring Network and Security Component Settings in vRealize Automation.
- (Optional) Configure network profile information.

  See Creating a Network Profile in vRealize Automation.
- Verify that you have access to a desired Amazon network. For example, if you want to use VPC, verify that you have access to an Amazon Virtual Private Cloud (VPC) network.

  See Using Optional Amazon Features.
- Verify that any required key pairs exist. See Managing Key Pairs.

Procedure

1. Select **Infrastructure > Reservations > Reservations**.
2. Click the **New** icon (✚) and select the type of reservation to create.

   Select **Amazon EC2**.
3. Enter a name in the **Name** text box.
4. Select a tenant from the **Tenant** drop-down menu.
5. Select a business group from the **Business group** drop-down menu.

   Only users in this business group can provision machines by using this reservation.
6. (Optional) Select a reservation policy from the **Reservation policy** drop-down menu.

   This option requires that one or more reservation policies exist. You can edit the reservation later to specify a reservation policy.

   You use a reservation policy to restrict provisioning to specific reservations.
7. Enter a number in the **Priority** text box to set the priority for the reservation.

   The priority is used when a business group has more than one reservation. A reservation with priority 1 is used for provisioning over a reservation with priority 2.
8. (Optional) Deselect the **Enable this reservation** check box if you do not want this reservation active.

Results

Do not navigate away from this page. Your reservation is not complete.

Specify Resource and Network Settings for Amazon Reservations

Specify resource and network settings for provisioning machines from this vRealize Automation reservation.

When you create an Amazon reservation or configure a machine component in the blueprint, you can choose from the list of security groups that are available to the specified Amazon account region. Security groups are imported during data collection. A security group acts as a firewall to control access to a machine. Every region includes at least the default security group. Administrators can use the Amazon Web Services Management Console to create additional security groups, configure ports for Microsoft Remote Desktop Protocol or SSH, and set up a virtual private network for an Amazon VPN. For information about creating and using security groups in Amazon Web Services, see Amazon documentation.

For related information about security groups, see Using Amazon Security Groups .

For related information about load balancers, see Using Elastic Load Balancers for Amazon Web Services.

Prerequisites

Specify Amazon Reservation Information.

Procedure

1 Click the **Resouces** tab.

2 Select a compute resource on which to provision machines from the **Compute resource** drop-down menu.

   Available Amazon regions are listed.

3 (Optional) Enter a number in the **Machine quota** text box to set the maximum number of machines that can be provisioned on this reservation.

   Only machines that are powered on are counted towards the quota. Leave blank to make the reservation unlimited.

4 Select a method of assigning key pairs to compute instances from the **Key pair** drop-down menu.

Option	Description
Not Specified	Controls key pair behavior at the blueprint level rather than the reservation level.
Auto-Generated per Business Group	Every machine provisioned in the same business group has the same key pair, including machines provisioned on other reservations when the machine has the same compute resource and business group. Because key pairs generated this way are associated with a business group, the key pairs are deleted when the business group is deleted.

Option	Description
Auto-Generated per Machine	Each machine has a unique key pair. This is the most secure method because no key pairs are shared among machines.
Specific Key Pair	Every machine provisioned on this reservation has the same key pair. Browse for a key pair to use for this reservation.

5 If you selected **Specific key Pair** in the **Key pair** drop-down menu, select a key pair value from the **Specific key pair** drop-down menu.

6 If you are configured for Amazon Virtual Private Cloud, enable the **Assign to a subnet in a VPC** check mark box. Otherwise, leave the box unchecked.

If you select **Assign to a subnet in a VPC**, the following locations or subnets, security groups, and load balancers options appear in a popup menu rather than on this same page.

For a VPC reservation, specify the security groups and subnets for each VPC that is allowed in the reservation.

7 Select one or more available locations (non-VPC) or subnets (VPC) from the **Locations** or **Subnets** list.

Select each available location or subnet that you want to be available for provisioning.

8 Select one or more security groups that can be assigned to a machine during provisioning from the **Security groups** list.

Select each security group that can be assigned to a machine during provisioning. Each available region requires at least one specified security group.

9 Select one or more available load balancers from the **Load balancers** list.

If you are using the elastic load balancer feature, select one or more available load balancers that apply to the selected locations or subnets.

Results

You can save the reservation now by clicking **Save**. Or you can add custom properties to further control reservation specifications. You can also configure email alerts to send notifications when resources allocated to this reservation become low.
Specify Custom Properties and Alerts for Amazon Reservations
You can associate custom properties with a vRealize Automation reservation. You can also configure alerts to send email notifications when reservation resources are low.

Custom properties and email alerts are optional configurations for the reservation. If you do not want to associate custom properties or set alerts, click **Save** to finish creating the reservation.

You can add as many custom properties as apply to your needs.

If configured, alerts are generated daily, rather than when the specified thresholds are reached.

Important Notifications are only sent if email alerts are configured and notifications are enabled.

Prerequisites

Specify Resource and Network Settings for Amazon Reservations.

Procedure

1. Click the **Properties** tab.
2. Click **New**.
3. Enter a valid custom property name.
4. If applicable, enter a property value.
5. Click **Save**.
6. (Optional) Add any additional custom properties.
7. Click the **Alerts** tab.
8. Enable the **Capacity Alerts** check box to configure alerts to be sent.
9. Use the slider to set thresholds for the available resource allocation.
10. Enter the AD user or group names (not email addresses) to receive alert notifications in the **Recipients** text box.

    Enter a name on each line. Press Enter to separate multiple entries.
11. Select **Send alerts to group manager** to include group managers in the email alerts.

    The email alerts are sent to the users included in the business group **Send manager emails to** list.
12. Specify a reminder frequency (days).
13. Click **Save**.

Results

The reservation is saved and appears in the Reservations list.

What to do next

You can configure optional reservation policies or begin preparing for provisioning.

Users who are authorized to create blueprints can create them now.

Create an OpenStack Reservation
You must allocate resources to machines by creating a reservation before members of a business group can request machine provisioning.

Create an OpenStack reservation.

Procedure

1. Specify OpenStack Reservation Information

   Each reservation is configured for a specific business group to grant them access to request machines on a specified compute resource.

2. Specify Resources and Network Settings for OpenStack Reservations

   Specify resource and network settings available to machines that are provisioned from this vRealize Automation reservation.

3. Specify Custom Properties and Alerts for OpenStack Reservations

   You can associate custom properties with a vRealize Automation reservation. You can also configure alerts to send email notifications when reservation resources are low.

Procedure

1. Specify OpenStack Reservation Information.
2. Specify Resources and Network Settings for OpenStack Reservations.
3. Specify Custom Properties and Alerts for OpenStack Reservations.

Specify OpenStack Reservation Information
Each reservation is configured for a specific business group to grant them access to request machines on a specified compute resource.

Note After you create a reservation, you cannot change the business group or compute resource associations.

You can control the display of reservations when adding, editing, or deleting by using the **Filter By Category** option on the Reservations page. Note that test agent reservations do not appear in the reservations list when filtering by category.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Verify that a tenant administrator created at least one business group.

  See Create a Business Group.
- Verify that a compute resource exists.
- Verify that any optional security groups or floating IP addresses are configured.

  See Preparing Red Hat OpenStack Network and Security Features.
- Verify that any required key pairs exist. See Managing Key Pairs.
- Verify that a compute resource exists.
- Configure network settings.

See Configuring Network and Security Component Settings in vRealize Automation.

Procedure

1 Select **Infrastructure > Reservations > Reservations**.

2 Click the **New** icon (+) and select the type of reservation to create.

Select **OpenStack**.

3 Enter a name in the **Name** text box.

4 Select a tenant from the **Tenant** drop-down menu.

5 Select a business group from the **Business group** drop-down menu.

Only users in this business group can provision machines by using this reservation.

6 (Optional) Select a reservation policy from the **Reservation policy** drop-down menu.

This option requires that one or more reservation policies exist. You can edit the reservation later to specify a reservation policy.

You use a reservation policy to restrict provisioning to specific reservations.

7 Enter a number in the **Priority** text box to set the priority for the reservation.

The priority is used when a business group has more than one reservation. A reservation with priority 1 is used for provisioning over a reservation with priority 2.

8 (Optional) Deselect the **Enable this reservation** check box if you do not want this reservation active.

Results

Do not navigate away from this page. Your reservation is not complete.
Specify Resources and Network Settings for OpenStack Reservations
Specify resource and network settings available to machines that are provisioned from this vRealize Automation reservation.

Prerequisites

Specify OpenStack Reservation Information.

Procedure

1 Click the **Resouces** tab.

2 Select a compute resource on which to provision machines from the **Compute resource** drop-down menu.

Only templates located on the cluster you select are available for cloning with this reservation.

During provisioning, machines are placed on a host that is connected to the local storage. If the reservation uses local storage, all the machines that are provisioned by the reservation are created on the host that contains that local storage. However, if you use the `VirtualMachine.Admin.ForceHost` custom property, which forces a machine to be provisioned to a different host, provisioning fails. Provisioning also fails if the template from which the machine is cloned is on local storage, but attached to a machine on a different cluster. In this case, provisioning fails because it cannot access the template.

3 (Optional) Enter a number in the **Machine quota** text box to set the maximum number of machines that can be provisioned on this reservation.

   Only machines that are powered on are counted towards the quota. Leave blank to make the reservation unlimited.

4 Select a method of assigning key pairs to compute instances from the **Key pair** drop-down menu.

Option	Description
Not Specified	Controls key pair behavior at the blueprint level rather than the reservation level.
Auto-Generated per Business Group	Every machine provisioned in the same business group has the same key pair, including machines provisioned on other reservations when the machine has the same compute resource and business group. Because key pairs generated this way are associated with a business group, the key pairs are deleted when the business group is deleted.
Auto-Generated per Machine	Each machine has a unique key pair. This is the most secure method because no key pairs are shared among machines.
Specific Key Pair	Every machine provisioned on this reservation has the same key pair. Browse for a key pair to use for this reservation.

5 If you selected **Specific key Pair** in the **Key pair** drop-down menu, select a key pair value from the **Specific key pair** drop-down menu.

6 Select one or more security groups that can be assigned to a machine during provisioning from the **Security groups** list.

7 Click the **Network** tab.

8 Configure a network path for machines provisioned by using this reservation.

   a (Optional) If the option is available, select a storage endpoint from the **Endpoint** drop-down menu.

      The FlexClone option is visible in the endpoint column if a NetApp ONTAP endpoint exists and if the host is virtual. If there is a NetApp ONTAP endpoint, the reservation page displays the endpoint assigned to the storage path. When you add, update, or delete an endpoint for a storage path, the change is visible in all the applicable reservations.

      When you add, update, or delete an endpoint for a storage path, the change is visible in the reservation page.

   b Select one or more **Network Adapters** for the machines to be provisioned for this reservation.

   c (Optional) Select an available **Network Profile** for each selected network adapter.

   d (Optional) If the Advanced settings are available, select a **Transport zone** and one or more **Tier 0 logical routers** to be used when deploying a blueprint that contains load balancers.

      A transport zone defines which clusters the network adapters span. If you specify a transport zone in a reservation and in a blueprint, the transport zone values must match.

   You can select more than one network adapter in a reservation, but only one network is used when provisioning a machine.

Results

You can save the reservation now by clicking **Save**. Or you can add custom properties to further control reservation specifications. You can also configure email alerts to send notifications when resources allocated to this reservation become low.
Specify Custom Properties and Alerts for OpenStack Reservations
You can associate custom properties with a vRealize Automation reservation. You can also configure alerts to send email notifications when reservation resources are low.

Custom properties and email alerts are optional configurations for the reservation. If you do not want to associate custom properties or set alerts, click **Save** to finish creating the reservation.

You can add as many custom properties as apply to your needs.

Important Notifications are only sent if email alerts are configured and notifications are enabled.

If configured, alerts are generated daily, rather than when the specified thresholds are reached.

Prerequisites

Specify Resources and Network Settings for OpenStack Reservations.

Procedure

1 Click the **Properties** tab.

2 Click **New**.

3 Enter a valid custom property name.

4 If applicable, enter a property value.

5 Click **Save**.

6 (Optional) Add any additional custom properties.

7 Click the **Alerts** tab.

8 Enable the **Capacity Alerts** check box to configure alerts to be sent.

9 Use the slider to set thresholds for the available resource allocation.

10 Enter the AD user or group names (not email addresses) to receive alert notifications in the **Recipients** text box.

   Enter a name on each line. Press Enter to separate multiple entries.

11 Select **Send alerts to group manager** to include group managers in the email alerts.

   The email alerts are sent to the users included in the business group **Send manager emails to** list.

12 Specify a reminder frequency (days).

13 Click **Save**.

Results

The reservation is saved and appears in the Reservations list.

What to do next

You can configure optional reservation policies or begin preparing for provisioning.

Users who are authorized to create blueprints can create them now.

Create a vCloud Air Reservation

You must allocate resources to machines by creating a vRealize Automation reservation before members of a business group can request machine provisioning.

Each business group must have at least one reservation for its members to provision machines of that type.

Procedure

1 Specify vCloud Air Reservation Information

   You can create a reservation for each vCloud Air machine subscription or OnDemand resource. Each reservation is configured for a specific business group to grant them access to request machines.

2 Specify Resources and Network Settings for a vCloud Air Reservation

   Specify resource and network settings available to vCloud Air machines that are provisioned from this vRealize Automation reservation.

3 Specify Custom Properties and Alerts for a vCloud Air Reservation

   You can associate custom properties with a vRealize Automation reservation. You can also configure alerts to send email notifications when reservation resources are low.

Procedure

1. Specify vCloud Air Reservation Information.
2. Specify Resources and Network Settings for a vCloud Air Reservation.
3. Specify Custom Properties and Alerts for a vCloud Air Reservation.

Specify vCloud Air Reservation Information

You can create a reservation for each vCloud Air machine subscription or OnDemand resource. Each reservation is configured for a specific business group to grant them access to request machines.

You can control the display of reservations when adding, editing, or deleting by using the **Filter By Category** option on the Reservations page. Note that test agent reservations do not appear in the reservations list when filtering by category.

Note After you create a reservation, you cannot change the business group or compute resource associations.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Verify that a tenant administrator created at least one business group.

  See Create a Business Group.
- Verify that a compute resource exists.
- Configure network settings.

  See Configuring Network and Security Component Settings in vRealize Automation.
- (Optional) Configure network profile information.

  See Creating a Network Profile in vRealize Automation.

Procedure

1. Select **Infrastructure > Reservations > Reservations**.
2. Click the **New** icon (✚) and select the type of reservation to create.

   The available cloud reservation types are Amazon, OpenStack, vCloud Air, and vCloud Director.

Select **vCloud Air**.

3 Enter a name in the **Name** text box.

4 Select a tenant from the **Tenant** drop-down menu.

5 Select a business group from the **Business group** drop-down menu.

Only users in this business group can provision machines by using this reservation.

6 (Optional) Select a reservation policy from the **Reservation policy** drop-down menu.

This option requires that one or more reservation policies exist. You can edit the reservation later to specify a reservation policy.

You use a reservation policy to restrict provisioning to specific reservations.

7 Enter a number in the **Priority** text box to set the priority for the reservation.

The priority is used when a business group has more than one reservation. A reservation with priority 1 is used for provisioning over a reservation with priority 2.

8 (Optional) Deselect the **Enable this reservation** check box if you do not want this reservation active.

Results

Do not navigate away from this page. Your reservation is not complete.
Specify Resources and Network Settings for a vCloud Air Reservation
Specify resource and network settings available to vCloud Air machines that are provisioned from this vRealize Automation reservation.

The available resource allocation models for machines provisioned from a vCloud Director reservation are Allocation Pool, Pay As You Go, and Reservation Pool. For Pay As You Go, you do not need to specify storage or memory amounts but do need to specify a priority for the storage path. For details about these allocation models, see vCloud Air documentation.

You can specify a standard or disk-level storage profile. Multi-level disk storage is available vCloud Air endpoints.

For integrations that use Storage Distributed Resource Scheduler (SDRS) storage, you can select a storage cluster to allow SDRS to automatically handle storage placement and load balancing for machines provisioned from this reservation. The SDRS automation mode must be set to Automatic. Otherwise, select a data store within the cluster for standalone data store behavior. SDRS is not supported for FlexClone storage devices.

Note Reservations defined for vCloud Air endpoints and vCloud Director endpoints do not support the use of network profiles for provisioning machines.

Prerequisites

Specify vCloud Director Reservation Information.

Procedure

1 Click the **Resouces** tab.

2 Select a compute resource on which to provision machines from the **Compute resource** drop-down menu.

   Only templates located on the cluster you select are available for cloning with this reservation.

3 Select an allocation model.

4 (Optional) Enter a number in the **Machine quota** text box to set the maximum number of machines that can be provisioned on this reservation.

   Only machines that are powered on are counted towards the quota. Leave blank to make the reservation unlimited.

5 Specify the amount of memory, in GB, to be allocated to this reservation from the Memory table.

   The overall memory value for the reservation is derived from your compute resource selection.

6 Select one or more listed storage paths.

   The available storage path options are derived from your compute resource selection.

   a Enter a value in the **This Reservation Reserved** text box to specify how much storage to allocate to this reservation.

   b Enter a value in the **Priority** text box to specify the priority value for the storage path relative to other storage paths that pertain to this reservation.

      The priority is used for multiple storage paths. A storage path with priority 0 is used before a path with priority 1.

   c Click the **Disable** option if you do not want to enable the storage path for use by this reservation.

   d Repeat this step to configure clusters and data stores as needed.

7 Click the **Network** tab.

8 Configure a network path for machines provisioned by using this reservation.

   a (Optional) If the option is available, select a storage endpoint from the **Endpoint** drop-down menu.

      The FlexClone option is visible in the endpoint column if a NetApp ONTAP endpoint exists and if the host is virtual. If there is a NetApp ONTAP endpoint, the reservation page displays the endpoint assigned to the storage path. When you add, update, or delete an endpoint for a storage path, the change is visible in all the applicable reservations.

      When you add, update, or delete an endpoint for a storage path, the change is visible in the reservation page.

   b Select one or more **Network Adapters** for the machines to be provisioned for this reservation.

   c (Optional) Select an available **Network Profile** for each selected network adapter.

   d (Optional) If the Advanced settings are available, select a **Transport zone** and one or more **Tier 0 logical routers** to be used when deploying a blueprint that contains load balancers.

      A transport zone defines which clusters the network adapters span. If you specify a transport zone in a reservation and in a blueprint, the transport zone values must match.

   You can select more than one network adapter in a reservation, but only one network is used when provisioning a machine.

Results

You can save the reservation now by clicking **Save**. Or you can add custom properties to further control reservation specifications. You can also configure email alerts to send notifications when resources allocated to this reservation become low.
Specify Custom Properties and Alerts for a vCloud Air Reservation
You can associate custom properties with a vRealize Automation reservation. You can also configure alerts to send email notifications when reservation resources are low.

Custom properties and email alerts are optional configurations for the reservation. If you do not want to associate custom properties or set alerts, click **Save** to finish creating the reservation.

You can add as many custom properties as apply to your needs.

If configured, alerts are generated daily, rather than when the specified thresholds are reached.

Important Notifications are only sent if email alerts are configured and notifications are enabled.

Alerts are not available for Pay As You Go reservations that were created with no specified limits.

Prerequisites

Specify Resources and Network Settings for a vCloud Air Reservation

Procedure

1 Click the **Properties** tab.

2 Click **New**.

3 Enter a valid custom property name.

4 If applicable, enter a property value.

5 (Optional) Check the **Encrypted** check box to encrypt the property value.

6 (Optional) Check the **Prompt User** check box to require that the user enter a value.

   This option cannot be overridden when provisioning.

7 Click **Save**.

8 (Optional) Add any additional custom properties.

9 Click the **Alerts** tab.

10 Enable the **Capacity Alerts** check box to configure alerts to be sent.

11 Use the slider to set thresholds for the available resource allocation.

12 Enter the AD user or group names (not email addresses) to receive alert notifications in the **Recipients** text box.

   Enter a name on each line. Press Enter to separate multiple entries.

13 Select **Send alerts to group manager** to include group managers in the email alerts.

   The email alerts are sent to the users included in the business group **Send manager emails to** list.

14 Specify a reminder frequency (days).

15 Click **Save**.

Results

The reservation is saved and appears in the Reservations list.
Create a vCloud Director Reservation
You must allocate resources to machines by creating a vRealize Automation reservation before members of a business group can request machine provisioning.

Each business group must have at least one reservation for its members to provision machines of that type.

Procedure

1 Specify vCloud Director Reservation Information

   You can create a reservation for each vCloud Director organization virtual datacenter (VDC). Each reservation is configured for a specific business group to grant them access to request machines on a specified compute resource.

2 Specify Resources and Network Settings for a vCloud Director Reservation

   Specify resource and network settings available to vCloud Director machines that are provisioned from this vRealize Automation reservation.

3 Specify Custom Properties and Alerts for vCloud Director Reservations

   You can associate custom properties with a vRealize Automation reservation. You can also configure alerts to send email notifications when reservation resources are low.

Procedure

1 Specify vCloud Director Reservation Information.

2 Specify Resources and Network Settings for a vCloud Director Reservation.

3 Specify Custom Properties and Alerts for vCloud Director Reservations.

What to do next

You can configure optional reservation policies or begin preparing for provisioning.

Users who are authorized to create blueprints can create them now.

Specify vCloud Director Reservation Information
You can create a reservation for each vCloud Director organization virtual datacenter (VDC). Each reservation is configured for a specific business group to grant them access to request machines on a specified compute resource.

You can control the display of reservations when adding, editing, or deleting by using the **Filter By Category** option on the Reservations page. Note that test agent reservations do not appear in the reservations list when filtering by category.

Note After you create a reservation, you cannot change the business group or compute resource associations.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Verify that a tenant administrator created at least one business group.

  See Create a Business Group.
- Verify that a compute resource exists.
- Configure network settings.

  See Configuring Network and Security Component Settings in vRealize Automation.
- (Optional) Configure network profile information.

  See Creating a Network Profile in vRealize Automation.

Procedure

1 Select **Infrastructure > Reservations > Reservations**.

2 Click the **New** icon (✚) and select the type of reservation to create.

   The available cloud reservation types are Amazon, OpenStack, vCloud Air, and vCloud Director.

   Select **vCloud Director**.

3 Enter a name in the **Name** text box.

4 Select a tenant from the **Tenant** drop-down menu.

5 Select a business group from the **Business group** drop-down menu.

   Only users in this business group can provision machines by using this reservation.

6 (Optional) Select a reservation policy from the **Reservation policy** drop-down menu.

   This option requires that one or more reservation policies exist. You can edit the reservation later to specify a reservation policy.

   You use a reservation policy to restrict provisioning to specific reservations.

7 Enter a number in the **Priority** text box to set the priority for the reservation.

   The priority is used when a business group has more than one reservation. A reservation with priority 1 is used for provisioning over a reservation with priority 2.

8 (Optional) Deselect the **Enable this reservation** check box if you do not want this reservation active.

Results

Do not navigate away from this page. Your reservation is not complete.
Specify Resources and Network Settings for a vCloud Director Reservation
Specify resource and network settings available to vCloud Director machines that are provisioned from this vRealize Automation reservation.

The available resource allocation models for machines provisioned from a vCloud Director reservation are Allocation Pool, Pay As You Go, and Reservation Pool. For Pay As You Go, you do not need to specify storage or memory amounts but do need to specify a priority for the storage path. For details about these allocation models, see vCloud Director documentation.

You can specify a standard or disk-level storage profile. Multi-level disk storage is available for vCloud Director 5.6 and greater endpoints. Multi-level disk storage is not supported for vCloud Director 5.5 endpoints.

For integrations that use Storage Distributed Resource Scheduler (SDRS) storage, you can select a storage cluster to allow SDRS to automatically handle storage placement and load balancing for machines provisioned from this reservation. The SDRS automation mode must be set to Automatic. Otherwise, select a data store within the cluster for standalone data store behavior. SDRS is not supported for FlexClone storage devices.

Note Reservations defined for vCloud Air endpoints and vCloud Director endpoints do not support the use of network profiles for provisioning machines.

Prerequisites

Specify vCloud Director Reservation Information.

Procedure

1. Click the **Resouces** tab.
2. Select a compute resource on which to provision machines from the **Compute resource** drop-down menu.

   Only templates located on the cluster you select are available for cloning with this reservation.
3. Select an allocation model.
4. (Optional) Enter a number in the **Machine quota** text box to set the maximum number of machines that can be provisioned on this reservation.

   Only machines that are powered on are counted towards the quota. Leave blank to make the reservation unlimited.
5. Specify the amount of memory, in GB, to be allocated to this reservation from the Memory table.

   The overall memory value for the reservation is derived from your compute resource selection.
6. Select one or more listed storage paths.

   The available storage path options are derived from your compute resource selection.

   a. Enter a value in the **This Reservation Reserved** text box to specify how much storage to allocate to this reservation.

   b. Enter a value in the **Priority** text box to specify the priority value for the storage path relative to other storage paths that pertain to this reservation.

      The priority is used for multiple storage paths. A storage path with priority 0 is used before a path with priority 1.

   c. Click the **Disable** option if you do not want to enable the storage path for use by this reservation.

   d. Repeat this step to configure clusters and data stores as needed.
7. Click the **Network** tab.

8 Configure a network path for machines provisioned by using this reservation.

   a (Optional) If the option is available, select a storage endpoint from the **Endpoint** drop-down menu.

      The FlexClone option is visible in the endpoint column if a NetApp ONTAP endpoint exists and if the host is virtual. If there is a NetApp ONTAP endpoint, the reservation page displays the endpoint assigned to the storage path. When you add, update, or delete an endpoint for a storage path, the change is visible in all the applicable reservations.

      When you add, update, or delete an endpoint for a storage path, the change is visible in the reservation page.

   b Select one or more **Network Adapters** for the machines to be provisioned for this reservation.

   c (Optional) Select an available **Network Profile** for each selected network adapter.

   d (Optional) If the Advanced settings are available, select a **Transport zone** and one or more **Tier 0 logical routers** to be used when deploying a blueprint that contains load balancers.

      A transport zone defines which clusters the network adapters span. If you specify a transport zone in a reservation and in a blueprint, the transport zone values must match.

   You can select more than one network adapter in a reservation, but only one network is used when provisioning a machine.

Results

You can save the reservation now by clicking **Save**. Or you can add custom properties to further control reservation specifications. You can also configure email alerts to send notifications when resources allocated to this reservation become low.
Specify Custom Properties and Alerts for vCloud Director Reservations
You can associate custom properties with a vRealize Automation reservation. You can also configure alerts to send email notifications when reservation resources are low.

Custom properties and email alerts are optional configurations for the reservation. If you do not want to associate custom properties or set alerts, click **Save** to finish creating the reservation.

You can add as many custom properties as apply to your needs.

If configured, alerts are generated daily, rather than when the specified thresholds are reached.

Important Notifications are only sent if email alerts are configured and notifications are enabled.

Alerts are not available for Pay As You Go reservations that were created with no specified limits.

Prerequisites

Specify Resources and Network Settings for a vCloud Director Reservation.

Procedure

1 Click the **Properties** tab.

2 Click **New**.

3 Enter a valid custom property name.

4 If applicable, enter a property value.

5 (Optional) Check the **Encrypted** check box to encrypt the property value.

6 (Optional) Check the **Prompt User** check box to require that the user enter a value.

   This option cannot be overridden when provisioning.

7 Click **Save**.

8 (Optional) Add any additional custom properties.

9 Click the **Alerts** tab.

10 Enable the **Capacity Alerts** check box to configure alerts to be sent.

11 Use the slider to set thresholds for the available resource allocation.

12 Enter the AD user or group names (not email addresses) to receive alert notifications in the **Recipients** text box.

   Enter a name on each line. Press Enter to separate multiple entries.

13 Select **Send alerts to group manager** to include group managers in the email alerts.

   The email alerts are sent to the users included in the business group **Send manager emails to** list.

14 Specify a reminder frequency (days).

15 Click **Save**.

Results

The reservation is saved and appears in the Reservations list.
Create a Reservation for Microsoft Azure
Create an Azure reservation for a specific business group to grant users in that group the ability to request Azure virtual machines on a specified compute resource.

If your deployment supports single sign-on through a VPN tunnel, you can configure support for this functionality with Azure virtual machines using the settings on the Properties tab.

Note Ignore the Alerts tab when creating an Azure reservation as it does not apply. After you create a reservation, you cannot change the business group associations. Also, unlike other machine types, there is no direct link between an Azure reservation and a blueprint.

You can control the display of reservations when adding, editing, or deleting by using the **Filter By Category** option on the Reservations page. Note that test agent reservations do not appear in the reservations list when filtering by category.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Verify that a tenant administrator created at least one business group.

  See Create a Business Group.
- (Optional) Configure network profile information.

  See Creating a Network Profile in vRealize Automation.
- Verify that you have access to any required Azure resources.
- Verify that any required key pairs exist. See Managing Key Pairs.
- Obtain a valid Azure Subscription ID that matches the one used with the applicable Azure endpoint. If you use multiple Azure subscriptions, you must create a reservation for each subscription.
- f your deployment supports single sign-on through a VPN tunnel, you must configure the appropriate VPC connectivity before creating a reservation. See Configure Network-to-Azure VPC Connectivity.

Procedure

1 Configure Microsoft Azure Basic Reservation Information

   Specify basic information for a Microsoft Azure reservation.

2 Configure Azure Reservation Resource Information

   When setting up an Azure reservation, you can assign resource group and storage account information based on the Azure instance you are using. When you set up a reservation, the vRealize Automation provisioning logic attempts to allocate resources, such as resource groups and storage accounts, according to the resource information specified by the reservation while provisioning a virtual machine.

3 Configure Azure Properties

   You can add custom properties to an Azure reservation to support options such as VPN tunneling to support communication between multiple networks. This functionality also facilitates the addition of software components to blueprints.

4 Configure Azure Reservation Network Information

   You can configure virtual network and load balancer information for an Azure virtual machine in the reservation.

Procedure

1 Configure Microsoft Azure Basic Reservation Information.

2 Configure Azure Reservation Resource Information.

3 Configure Azure Properties.

4 Configure Azure Reservation Network Information.

Configure Microsoft Azure Basic Reservation Information

Specify basic information for a Microsoft Azure reservation.

All information on the Reservation Information page are required except the Reservation Policy. All information on subsequent Azure reservation pages is optional.

Procedure

1. Select **Infrastructure > Administration > Reservations**.
2. Click the **New** icon (✚) and select the type of reservation to create.

   Select **Azure**.
3. Enter a name in the **Name** text box.
4. Select a business group from the **Business group** drop-down menu.

   Only users in this business group can provision machines by using this reservation.
5. Ignore the **Reservation policy** text box, as it does not apply to Azure reservations.
6. Enter a number in the **Priority** text box to set the priority for the reservation.

   The priority is used when a business group has more than one reservation. A reservation with priority 1 is used for provisioning over a reservation with priority 2.
7. (Optional) Deselect the **Enable this reservation** check box if you do not want this reservation active.
8. Click **OK**.

Configure Azure Reservation Resource Information

When setting up an Azure reservation, you can assign resource group and storage account information based on the Azure instance you are using. When you set up a reservation, the vRealize Automation provisioning logic attempts to allocate resources, such as resource groups and storage accounts, according to the resource information specified by the reservation while provisioning a virtual machine.

You can configure Resource Group and Storage Account information for an Azure virtual machine in the reservation, but you can also choose to leave these fields blank in the reservation. If you leave the fields blank, the default resource group and storage account information related to the specified Azure subscription ID will be used for any related blueprints. You can also update this information when creating a blueprint or when you provision a virtual machine.

Prerequisites

Obtain the subscription ID for your Azure instance.

Procedure

1. Enter your Azure subscription ID in the **Subscription ID** text box.

2 Select the location for the reservation by clicking the **Location** drop-down.

   You can leave this field blank to create a location agnostic reservation, but if you do location information must be specified either when creating a blueprint or when provisioning an Azure virtual machine.

3 Click **New** in the Resource Groups table.

   a Enter the appropriate Resource Group name information from your Azure instance in the **Name** text box.

   Note The **Name** box cannot be empty.

   b Assign a numeric priority value in the **Priority** text box.

   This assignment determines priority when a Resource Group has more than one resource group, with lower numbers taking precedence.

   c Click **Save** to add the Resource Group to the reservation.

4 Click **New** in the Storage Accounts table.

   a Enter the appropriate Storage Account name information from your Azure instance in the **Name** text box.

   Note The **Name** box cannot be empty.

   b Assign a numeric priority value in the **Priority** text box.

   c Click **Save** to add the Storage Account to the reservation.

   This assignment determines priority when a reservation has more than one Storage Account, with lower numbers taking precedence.

5 Click **OK** to proceed to the next tab.

Configure Azure Properties
You can add custom properties to an Azure reservation to support options such as VPN tunneling to support communication between multiple networks. This functionality also facilitates the addition of software components to blueprints.

You must create custom properties that define the appropriate URLs to support VPN tunneling on your network. In addition, you must create properties that define the path to the Azure tunneling configuration scripts downloaded previously.

Use the private IP address of your Azure tunnel physical machine and port 1443, which you assigned for *vRealize_automation_appliance_fqdn* when you invoked the SSH tunnel.

The following table shows the names and values for the properties required to support VPN tunneling.

Name	Value
`Azure.Windows.ScriptPath`	Specifies the path to the downloaded script that configures tunneling for Windows-based systems. Update the path as appropriate for your deployment.
`Azure.Linux.ScriptPath`	Specifies the path to the downloaded script that configures tunneling for Linux-based systems. Update the path as appropriate for your deployment.
`agent.download.url`	Specifies the URL for the VPN agent on your deployment. The URL format is `https://` *`Private_IP`*`:1443/software-service//resources/noble-agent.jar`
`software.agent.service.url`	Enter the VPN software agent service URL for your deployment, The URL format is `https://` *`Private_IP`*`:1443/software-service/api`
`software.ebs.url`	Enter the event broker service URL for your deployment. The URL format is `https://` *`Private_IP`*`:1443/event-broker-service/api`

Prerequisites

- Download the VMware-supplied Azure scripts from the **Guest and Software Agent Installers** page on your vRealize Automation Appliance.

  These scripts install Azure extensions required to support VPN tunneling. There are two scripts: `script.ps1` and `script.sh`. The `.ps1` file is for Windows systems, and the `.sh` file is for Linux systems.

  a Run `https://`*`vrealize-automation-appliance-fqdn`*`/software` to open the VMware vRealize Automation Appliance page.

  b Click the **Guest and software agents** link under the To install vRealize Automation components (IaaS, Guest and Software Agents, Tools) heading.

  c Download the Azure script files under the Azure Machines heading. Save the script files to an appropriate location. You must point to this location when configuring Azure reservation custom properties.

Procedure

1 Click the **Properties** tab.

2 Click **New**.

3 Enter the appropriate Name and Value for the custom property in the Properties dialog box.

4 As you create each property, click **OK** on the dialog box to add that property.

5 When you finish adding all required properties, click **OK** to save your settings.

What to do next

After you create the custom properties to support VPN tunneling, you can create software components for your Azure blueprints. See Designing Software Components for more information.

When setting up a software component for Azure, select **Azure Virtual Machine** in the Container drop-down on the New Software page.

Configure Azure Reservation Network Information

You can configure virtual network and load balancer information for an Azure virtual machine in the reservation.

You can also choose to leave this page partly or completely blank and configure virtual network and load balancer information when you provision a virtual machine.

If you specify a network profile and do not specify a subnet, then the name of the first existing network range of the specified network profile is used as the subnet name. If a network profile is specified, you can choose to leave the vNet text box blank. In this case, the name of this first network range of the specified network profile is used as the subnet name, and the vNet name is resolved to the first Azure vNet that contains an applicable subnet.

Prerequisites

Obtain the appropriate virtual network and load balancer information from your Azure instance where applicable.

Procedure

1 Click **New** in the Networks table to configure the appropriate Azure virtual network to use with your virtual machine.

   a Paste the appropriate vNet name information from your Azure instance into the **vNet** text box.

   b Paste the appropriate Subnet name information from your Azure instance into the **Subnet** text box.

   The Subnet specification is optional. If you leave this box empty, the subnet of the specified vNet is used by default.

   c Type or paste the appropriate name in the **Network Profile** text box. You can use the network profile in the blueprint to associate a network interface card with a network.

   The network profile specification is optional. Use if it you want to create you blueprint based on the network profile is defined in vRealize Automation rather than have it coupled with Azure network constructs.

   d Assign a numerical priority value in the **Priority** text box if applicable.

   This assignment determines priority when a virtual network has more than one reservation, with lower numbers taking precedence.

   e Click **Save** to add the Resource Group to the reservation.

2 Click **New** in the Load Balancers table if you are deploying multiple machines and use a load balancer.

   a Paste the appropriate load balancer name from your Azure instance into the **Name** text box.

   b Paste the appropriate name from your Azure instance into the **Backend Address Pool** text box.

   c Assign a numerical priority value in the **Priority** text box if applicable.

   This assignment determines priority when a virtual network has more than one load balancer, with lower numbers taking precedence.

   d Click **Save** to add the load balancer to the reservation.

3 Click **New** in the Security Groups table if you are deploying multiple machines that must communicate through a firewall.

   a Paste the security group name from your Azure instance into the **Name** text box.

   b Assign a numerical priority value in the **Priority** text box if applicable.

   This assignment determines priority when a virtual network has more than one security group, with lower numbers taking precedence.

   c Click **Save** to add the security group to the reservation.

4 Click **OK**.

Scenario: Create an Amazon Reservation for a Proof of Concept Environment
Because you used an SSH tunnel to temporarily establish network-to-Amazon VPC connectivity for your proof of concept environment, you have to add custom properties to your Amazon reservations to ensure the Software bootstrap agent and guest agent run communications through the tunnel.

Network-to-Amazon VPC connectivity is only required if you want to use the guest agent to customize provisioned machines, or if you want to include Software components in your blueprints. For a production environment, you would configure this connectivity officially through Amazon Web Services, but because you are working in a proof of concept environment, you configured a temporary SSH tunnel instead.

Using your fabric administrator privileges, you create a reservation to allocate your Amazon Web Services resources and you include several custom properties to support the SSH tunneling. You also configure the reservation on the same region and VPC as your tunnel machine.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Configure an SSH tunnel to establish network-to-Amazon VPC connectivity. Make a note of the subnet, security group, and private IP address of your Amazon Web Services tunnel machine. See Configure Network-to-Amazon VPC Connectivity for a Proof of Concept Environment.

- Create a business group for members of your IT organization who need to architect blueprints in your proof of concept environment. See Create a Business Group.
- Verify that a tenant administrator created at least one business group.

  See Create a Business Group.

Procedure

1 Scenaro: Specify Amazon Web Services Reservation Information for a Proof of Concept Environment

  You want to reserve resources for your team of blueprint architects so they can test the functionality in your proof of concept environment, so you configure this reservation to allocate resources to your architects business group.

2 Scenario: Specify Amazon Web Services Network Settings for a Proof of Concept Environment

  You configure the reservation to use the same region and networking settings that your tunnel machine is using, and you restrict the number of machines that can be powered on for this reservation to manage resource usage.

3 Scenario: Specify Custom Properties to Run Agent Communications Through Your Tunnel

  When you configured network-to-Amazon VPC connectivity, you configured port forwarding to allow your Amazon Web Services tunnel machine to access vRealize Automation resources.

Procedure

1 Scenaro: Specify Amazon Web Services Reservation Information for a Proof of Concept Environment.

2 Scenario: Specify Amazon Web Services Network Settings for a Proof of Concept Environment.

3 Scenario: Specify Custom Properties to Run Agent Communications Through Your Tunnel.

Scenaro: Specify Amazon Web Services Reservation Information for a Proof of Concept Environment
You want to reserve resources for your team of blueprint architects so they can test the functionality in your proof of concept environment, so you configure this reservation to allocate resources to your architects business group.

Note After you create a reservation, you cannot change the business group or compute resource associations.

Procedure

1 Select **Infrastructure > Reservations > Reservations**.

2 Click the **New** icon (✚) and select the type of reservation to create.

   Select **Amazon**.

3 Enter `Amazon Tunnel POC` in the **Name** text box.

4 Select the business group you created for your blueprint architects from the **Business Group** drop-down menu.

5 Enter a **1** in the **Priority** text box to set this reservation as the highest priority.

Results

You configured the business group and the priority for the reservation, but you still need to allocate resources and configure the custom properties for the SSH tunnel.
Scenario: Specify Amazon Web Services Network Settings for a Proof of Concept Environment
You configure the reservation to use the same region and networking settings that your tunnel machine is using, and you restrict the number of machines that can be powered on for this reservation to manage resource usage.

Procedure

1 Click the **Resouces** tab.

2 Select a compute resource on which to provision machines from the **Compute resource** drop-down menu.

   Select the Amazon Web Services region where your tunnel machine is located.

3 (Optional) Enter a number in the **Machine quota** text box to set the maximum number of machines that can be provisioned on this reservation.

   Only machines that are powered on are counted towards the quota. Leave blank to make the reservation unlimited.

4 Select **Specify Key Pair** from the **Key pair** drop-down menu.

   Because this is a proof of concept environment, you choose to share a single key pair for all machines provisioned by using this reservation.

5 Select the key pair you want to share with your architect users from the **Key Pair** drop-down menu.

6 Enable the **Assign to a subnet in a VPC** checkbox.

7 Select the same subnet and security groups that your tunnel machine is using.

Results

You configured the reservation to use the same region and networking settings as your tunnel machine, but you still need to add custom properties to ensure the Software bootstrap agent and guest agent run communications through the tunnel.
Scenario: Specify Custom Properties to Run Agent Communications Through Your Tunnel

When you configured network-to-Amazon VPC connectivity, you configured port forwarding to allow your Amazon Web Services tunnel machine to access vRealize Automation resources.

You need to add tunnel custom properties on the reservation to configure the agents to access those ports.

Note If you are using a PAT or NAT system network between your organization's network and the vRealize Automation network, you can use these properties to access your private IP address and port.

Procedure

1 Click the **Properties** tab.

2 Click **New**.

3 Configure the tunnel custom properties.

Use the private IP address of your Amazon Web Services tunnel machine and port 1443, which you assigned for *vRealize_automation_appliance_fqdn* when you invoked the SSH tunnel.

Option	Value
`software.ebs.url`	`https://`*`Private_IP:1443`*`/event-broker-service/api`
`software.agent.service.url`	`https://`*`Private_IP:1443`*`/software-service/api`
`agent.download.url`	`https://`*`Private_IP:1443`*`/software-service/resources/nobel-agent.jar`

4 Click **Save**.

Results

You created a reservation to allocate Amazon Web Services resources to your architects business group. You configured the reservation to support the guest agent and the Software bootstrap agent. Your architects can create blueprints that leverage the guest agent to customize deployed machines or include Software components.

Creating Virtual Category Reservations

A virtual category type reservation provides access to the provisioning services of a virtual machine deployment for a particular vRealize Automation business group. Available virtual reservation types include vSphere, Hyper-V, KVM, SCVMM, and XenServer.

A reservation is a share of the memory, CPU, networking, and storage resources of one compute resource allocated to a particular vRealize Automation business group.

A business group can have multiple reservations on one endpoint or reservations on multiple endpoints.

To provision virtual machines, a business group must have at least one reservation on a virtual compute resource. Each reservation is for one business group only, but a business group can have multiple reservations on a single compute resource, or multiple reservations on compute resources of different types.

In addition to defining the share of fabric resources allocated to the business group, a reservation can define policies, priorities, and quotas that determine machine placement.

To provision successfully, the reservation must have sufficient available storage. The reservation's storage availability depends on:

- How much storage is available on the data store/cluster.
- How much of that storage is reserved for that data store/cluster.
- How much of that storage is already allocated in vRealize Automation

For example, even if the vCenter Server has storage available for the data store/cluster, if sufficient storage is not reserved in the reservation then provisioning fails with a "No reservation is available to allocate..." error. The allocated storage on a reservation depends on the number of VMs (regardless of their state) on that specific reservation. See the VMware Knowledge Base article *Machine XXX: No reservation is available to allocate within the group XXX. Total XX GB of storage was requested (2151030)* at http://kb.vmware.com/kb/2151030 for more information.

Understanding Selection Logic for Reservations

When a member of a business group create a provisioning request for a virtual machine, vRealize Automation selects a machine from one of the reservations that are available to that business group.

The reservation for which a machine is provisioned must satisfy the following criteria:

- The reservation must be of the same platform type as the blueprint from which the machine was requested.

  A generic virtual blueprint can be provisioned on any type of virtual reservation.
- The reservation must be enabled.
- The compute resource must be accessible and not in maintenance mode.
- The reservation must have capacity remaining in its machine quota or have an unlimited quota.

  The allocated machine quota includes only machines that are powered on. For example, if a reservation has a quota of 50, and 40 machines have been provisioned but only 20 of them are powered on, the reservation’s quota is 40 percent allocated, not 80 percent.
- The reservation must have sufficient unallocated memory and storage resources to provision the machine.

  When a virtual reservation’s machine quota, memory, or storage is fully allocated, no further virtual machines can be provisioned from it. Resources may be reserved beyond the physical capacity of a virtualization compute resource (overcommitted), but when the physical capacity of a compute resource is 100% allocated, no further machines can be provisioned on any reservations with that compute resource until the resources are reclaimed.
- If the blueprint has specific network settings, the reservation must have the same networks.

  If the blueprint or reservation specifies a network profile for static IP address assignment, an IP address must be available to assign to the new machine.

- If the blueprint or request specifies a location, the compute resource must be associated with that location.

  If the value of the custom property `Vrm.DataCenter.Policy` is **`Exact`** and there is no reservation for a compute resource associated with that location that satisfies all the other criteria, then provisioning fails.

  If the value of `Vrm.DataCenter.Policy` is **`NotExact`** and there is no reservation for a compute resource associated with that location that satisfies all the other criteria, provisioning can proceed on another reservation regardless of location. This option is the default.

- If the blueprint or request specifies the custom property `VirtualMachine.Host.TpmEnabled`, trusted hardware must be installed on the compute resource for the reservation.

- If the blueprint specifies a reservation policy, the reservation must belong to that reservation policy.

  Reservation policies are a way to guarantee that the selected reservation satisfies any additional requirements for provisioning machines from a specific blueprint. For example, you can use reservation policies to limit provisioning to compute resources with a specific template for cloning.

If no reservation is available that meets all of the selection criteria, provisioning fails.

If multiple reservations meet all of the criteria, the reservation from which to provision a requested machine is determined by the following logic:

- A reservation with a lower priority value is selected before a reservation with a higher priority value.

- If multiple reservations have the same priority, the reservation with the lowest percentage of its machine quota allocated is selected.

- If multiple reservations have the same priority and quota usage, machines are distributed among reservations in round-robin fashion.

  Note While round-robin selection of network profiles is not supported, round-robin selection of networks (if any) is supported, which can be associated with different network profiles.

If multiple storage paths are available on a reservation with sufficient capacity to provision the machine volumes, storage paths are selected according to the following logic:

- If the blueprint or request specifies a storage reservation policy, the storage path must belong to that storage reservation policy.

  If the value of the custom property `VirtualMachine.Disk`*`N`*`.StorageReservationPolicyMode` is **`NotExact`** and there is no storage path with sufficient capacity within the storage reservation policy, then provisioning can proceed with a storage path outside the specified storage reservation policy. The default value of `VirtualMachine.Disk`*`N`*`.StorageReservationPolicyMode` is **`Exact`**.

- A storage path with a lower priority value is selected before a storage path with a higher priority value.
- If multiple storage paths have the same priority, machines are distributed among storage paths in round-robin fashion.

Creating a vSphere Reservation for NSX Network and Security in vRealize Automation
You can create a vSphere reservation to work with your associated NSX-T or NSX for vSphere endpoint in vRealize Automation.

General NSX Considerations

If you configured NSX, you can specify NSX transport zone, network reservation policy, and app isolation settings when you create or edit a blueprint. These settings are available on the **NSX Settings** tab on the **Blueprint** and **Blueprint Properties** pages.

The network and security component settings that you add to the blueprint are derived from your NSX for vSphere and NSX-T configuration. For information about configuring NSX, see the *Administration Guide* in NSX for vSphere product documentation or NSX-T product documentation, depending on which application you are using.

Successful provisioning requires the transport zone of the reservation to match the transport zone of a machine blueprint when that blueprint defines machine networks. Similarly, provisioning a machine's routed gateway requires that the transport zone defined in the reservation matches the transport zone defined for the blueprint.

For information about NSX-T-specific topology considerations in your deployments, see Understanding NSX-T Deployment Topologies for Networking, Security, and Load Balancer Configurations.

NSX for vSphere Considerations

When vRealize Automation provisions machines with NAT or routed networking, it provisions a routed gateway as the network router. The Edge or routed gateway is a management machine that consumes compute resources. It also manages the network communications for the provisioned machine components. The reservation used to provision the Edge or routed gateway determines the external network used for NAT and routed network profiles. It also determines the reservation Edge or routed gateway used to configure routed networks. The reservation routed gateway links routed networks together with entries in the routing table.

When you select an Edge or routed gateway and network profile on a reservation for routed networks, select the network path to be used in linking routed networks together. Assign the network path to the external network profile that is used to configure the routed network profile. The list of network profiles available to be assigned to a network path is filtered to match the subnet of the network path based on the subnet mask and primary IP address selected for the network interface.

You can specify an Edge or routed gateway reservation policy to identify which reservations to use when provisioning the machines using the Edge or routed gateway. By default, vRealize Automation uses the same reservations for the routed gateway and the machine components.

If you want to use an Edge or routed gateway in vRealize Automation reservations, configure the routed gateway externally in the NSX environment and then run inventory data collection. For NSX, you must have a working NSX Edge instance before you can configure the default gateway for static routes or dynamic routing details for an Edge services gateway or distributed router. See *NSX Administration Guide*.

You select one or more security groups in the reservation to enforce baseline security policy for all component machines provisioned with that reservation in vRealize Automation. Every provisioned machine is added to these specified security groups.

NSX-T Considerations

When you create a reservation for a vSphere endpoint that is associated to an NSX-T endpoint, you must configure the following information for the reservation:

- Define a transport zone for the blueprint.
- Select a Tier-0 logical router for the provisioned deployment to connect to.
- Map an external network profile to the Tier 0 logical router.

NSX-T NS groups are not supported in reservations.

For more information about NSX-T-specific deployment and topology considerations, see Understanding NSX-T Deployment Topologies for Networking, Security, and Load Balancer Configurations.
Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer
You must allocate resources to machines by creating a reservation before members of a business group can request machine provisioning.

Each business group must have at least one reservation for its members to provision machines of that type. For example, a business group with a vSphere reservation, but not a KVM (RHEV) reservation, cannot request a KVM (RHEV) virtual machine. In this example, the business group must be allocated a reservation specifically for KVM (RHEV) resources.

Procedure

1 Specify Virtual Reservation Information

  Each reservation is configured for a specific business group to grant users access to request machines on a specified compute resource.

2 Specify Resource and Networking Settings for a Virtual Reservation

  Specify resource and network settings for provisioning machines from this vRealize Automation reservation.

3 Specify Custom Properties and Alerts for Virtual Reservations

  You can associate custom properties with a vRealize Automation reservation. You can also configure alerts to send email notifications when reservation resources are low.

Procedure

1 Specify Virtual Reservation Information.

2 Specify Resource and Networking Settings for a Virtual Reservation.

3 Specify Custom Properties and Alerts for Virtual Reservations.

Specify Virtual Reservation Information

Each reservation is configured for a specific business group to grant users access to request machines on a specified compute resource.

You can control the display of reservations when adding, editing, or deleting by using the **Filter By Category** option on the Reservations page. Note that test agent reservations do not appear in the reservations list when filtering by category.

Note After you create a reservation, you cannot change the business group or compute resource associations.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Verify that a tenant administrator created at least one business group.

  See Create a Business Group.
- Verify that a compute resource exists.
- Configure network settings.

  See Configuring Network and Security Component Settings in vRealize Automation.
- (Optional) Configure network profile information.

  See Creating a Network Profile in vRealize Automation.

Procedure

1 Select **Infrastructure > Reservations > Reservations**.

2 Click the **New** icon (✚) and select the type of reservation to create.

  The available virtual reservation types are Hyper-V, KVM, SCVMM, vSphere, and XenServer.

  For example, select **vSphere**.

3 Enter a name in the **Name** text box.

4 Select a tenant from the **Tenant** drop-down menu.

5 Select a business group from the **Business group** drop-down menu.

  Only users in this business group can provision machines by using this reservation.

6 (Optional) Select a reservation policy from the **Reservation policy** drop-down menu.

  This option requires that one or more reservation policies exist. You can edit the reservation later to specify a reservation policy.

  You use a reservation policy to restrict provisioning to specific reservations.

7 Enter a number in the **Priority** text box to set the priority for the reservation.

The priority is used when a business group has more than one reservation. A reservation with priority 1 is used for provisioning over a reservation with priority 2.

8 (Optional) Deselect the **Enable this reservation** check box if you do not want this reservation active.

Results

Do not navigate away from this page. Your reservation is not complete.
Specify Resource and Networking Settings for a Virtual Reservation
Specify resource and network settings for provisioning machines from this vRealize Automation reservation.

You can select a FlexClone data store in your reservation if you have a vSphere environment and storage devices that use Net App FlexClone technology. SDRS is not supported for FlexClone storage devices.

To provision successfully, the reservation must have sufficient available storage. The reservation's storage availability depends on:

- How much storage is available on the data store/cluster.
- How much of that storage is reserved for that data store/cluster.
- How much of that storage is already allocated in vRealize Automation

For example, even if the vCenter Server has storage available for the data store/cluster, if sufficient storage is not reserved in the reservation then provisioning fails with a "No reservation is available to allocate..." error. The allocated storage on a reservation depends on the number of VMs (regardless of their state) on that specific reservation. See the VMware Knowledge Base article *Machine XXX: No reservation is available to allocate within the group XXX. Total XX GB of storage was requested (2151030)* at http://kb.vmware.com/kb/2151030 for more information.

If you are creating or editing a vSphere (vCenter) reservation for use with NSX for vSphere or NSX-T, you can specify transfer zone and tier 1 logical router information by using advanced options for the selected network.

Prerequisites

Specify Virtual Reservation Information.

Procedure

1 Click the **Resouces** tab.

2 Select a compute resource on which to provision machines from the **Compute resource** drop-down menu.

Only templates located on the cluster you select are available for cloning with this reservation.

During provisioning, machines are placed on a host that is connected to the local storage. If the reservation uses local storage, all the machines that are provisioned by the reservation are created on the host that contains that local storage. However, if you use the `VirtualMachine.Admin.ForceHost` custom property, which forces a machine to be provisioned to a different host, provisioning fails. Provisioning also fails if the template from which the machine is cloned is on local storage, but attached to a machine on a different cluster. In this case, provisioning fails because it cannot access the template.

3 (Optional) Enter a number in the **Machine quota** text box to set the maximum number of machines that can be provisioned on this reservation.

   Only machines that are powered on are counted towards the quota. Leave blank to make the reservation unlimited.

4 Specify the amount of memory, in GB, to be allocated to this reservation from the Memory table.

   The overall memory value for the reservation is derived from your compute resource selection.

5 Specify the amount of memory, in GB, to be allocated to this reservation from the Memory table.

   The overall memory value for the reservation is derived from your compute resource selection.

6 Select one or more listed storage paths.

   The available storage path options are derived from your compute resource selection.

   For integrations that use Storage Distributed Resource Scheduler (SDRS) storage, you can select a storage cluster to allow SDRS to automatically handle storage placement and load balancing for machines provisioned from this reservation. The SDRS automation mode must be set to Automatic. Otherwise, select a data store within the cluster for standalone data store behavior. SDRS is not supported for FlexClone storage devices.

   You can select either individual disks in the cluster or a storage cluster, but not both. If you select a storage cluster, SDRS controls storage placement and load balancing for machines that are provisioned from this reservation.

7 If available for the compute resource, select a resource pool in the **Resource Pool** drop-down menu.

8 Click the **Network** tab.

9 Configure a network path for machines provisioned by using this reservation.

   a (Optional) If the option is available, select a storage endpoint from the **Endpoint** drop-down menu.

      The FlexClone option is visible in the endpoint column if a NetApp ONTAP endpoint exists and if the host is virtual. If there is a NetApp ONTAP endpoint, the reservation page displays the endpoint assigned to the storage path. When you add, update, or delete an endpoint for a storage path, the change is visible in all the applicable reservations.

      When you add, update, or delete an endpoint for a storage path, the change is visible in the reservation page.

   b Select one or more **Network Adapters** for the machines to be provisioned for this reservation.

   c (Optional) Select an available **Network Profile** for each selected network adapter.

   d (Optional) If the Advanced settings are available, select a **Transport zone** and one or more **Tier 0 logical routers** to be used when deploying a blueprint that contains load balancers.

      A transport zone defines which clusters the network adapters span. If you specify a transport zone in a reservation and in a blueprint, the transport zone values must match.

   You can select more than one network adapter in a reservation, but only one network is used when provisioning a machine.

Results

You can save the reservation now by clicking **Save**. Or you can add custom properties to further control reservation specifications. You can also configure email alerts to send notifications when resources allocated to this reservation become low.
Specify Custom Properties and Alerts for Virtual Reservations
You can associate custom properties with a vRealize Automation reservation. You can also configure alerts to send email notifications when reservation resources are low.

Custom properties and email alerts are optional configurations for the reservation. If you do not want to associate custom properties or set alerts, click **Save** to finish creating the reservation.

You can add as many custom properties as apply to your needs.

Important Notifications are only sent if email alerts are configured and notifications are enabled.

If configured, alerts are generated daily, rather than when the specified thresholds are reached.

Prerequisites

Specify Resource and Networking Settings for a Virtual Reservation.

Procedure

1 Click the **Properties** tab.

2 Click **New**.

3 Enter a valid custom property name.

4 If applicable, enter a property value.

5 (Optional) Check the **Encrypted** check box to encrypt the property value.

6 (Optional) Check the **Prompt User** check box to require that the user enter a value.

   This option cannot be overridden when provisioning.

7 (Optional) Add any additional custom properties.

8 Click the **Alerts** tab.

9 Enable the **Capacity Alerts** check box to configure alerts to be sent.

10 Use the slider to set thresholds for the available resource allocation.

11 Enter the AD user or group names (not email addresses) to receive alert notifications in the **Recipients** text box.

   Enter a name on each line. Press Enter to separate multiple entries.

12 Select **Send alerts to group manager** to include group managers in the email alerts.

   The email alerts are sent to the users included in the business group **Send manager emails to** list.

13 Specify a reminder frequency (days).

14 Click **Save**.

Results

The reservation is saved and appears in the Reservations list.

What to do next

You can configure optional reservation policies or begin preparing for provisioning.

Users who are authorized to create blueprints can create them now.

Edit a Reservation to Assign a Network Profile

You can assign a network profile to a reservation, for example to enable static IP assignment for machines that are provisioned on that reservation.

You can also assign a network profile to a blueprint by using the custom property `VirtualMachine.NetworkN.ProfileName` on the **Properties** tab of the **New Blueprint** or **Blueprint Properties** page.

If you specify a network profile in a reservation and a blueprint, the blueprint values take precedence.

Note This information does not apply to Amazon Web Services.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Create a network profile. See Creating a Network Profile in vRealize Automation.

Procedure

1 Select **Infrastructure > Reservations > Reservations**.

2 Point to a reservation and click **Edit**.

3 Click the **Network** tab.

4 Assign a network profile to a network path.

   a Select a network path on which to enable static IP addresses.

   The network path options are derived from settings on the **Resources** tab.

   b Map an available network profile to the path by selecting a profile from the **Network Profile** drop-down menu.

   c (Optional) Repeat this step to assign network profiles to additional network paths on this reservation.

5 Click **OK**.

Reservation Policies

You can use a reservation policy to control how reservation requests are processed. When you provision machines from the blueprint, provisioning is restricted to the resources specified in your reservation policy.

Reservation policies provide an optional means of controlling how reservation requests are processed. You can apply a reservation policy to a blueprint to restrict the machines provisioned from that blueprint to a subset of available reservations.

You can use a reservation policy to collect resources into groups for different service levels, or to make a specific type of resource easily available for a particular purpose. When a user requests a machine, it can be provisioned on any reservation of the appropriate type that has sufficient capacity for the machine. The following scenarios provide a few examples of possible uses for reservation policies:

- To ensure that provisioned machines are placed on reservations with specific devices that support NetApp FlexClone.
- To restrict provisioning of cloud machines to a specific region containing a machine image that is required for a specific blueprint.
- As an additional means of using a Pay As You Go allocation model for machine types that support that capability.

You can add multiple reservations to a reservation policy, but a reservation can belong to only one policy. You can assign a single reservation policy to more than one blueprint. A blueprint can have only one reservation policy.

Note Reservations defined for vCloud Air endpoints and vCloud Director endpoints do not support the use of network profiles for provisioning machines.

Note If you have SDRS enabled on your platform, you can allow SDRS to load balance storage for individual virtual machine disks, or all storage for the virtual machine. If you are working with SDRS datastore clusters, conflicts can occur when you use reservation policies and storage reservation policies. For example, if a standalone datastore or a datastore within an SDRS cluster is selected on one of the reservations in a policy or storage policy, your virtual machine storage might be frozen instead of driven by SDRS. If you request reprovisioning for a machine with storage placement on an SDRS cluster, the machine is deleted if the SDRS automation level is deactivated. For related information about provisioning and SDRS, see the `VirtualMachine.Admin.Datastore.Cluster.ResourceLeaseDurationSec` custom property.

Configure a Reservation Policy

You can create reservation policies to collect resources into groups for different service levels, or to make a specific type of resource easily available for a particular purpose. After you create the reservation policy, you then must populate it with reservations before tenant administrators and business group managers can use the policy effectively in a blueprint.

A reservation policy can include reservations of different types, but only reservations that match the blueprint type are considered when selecting a reservation for a particular request.

Procedure

1 Create a Reservation Policy

  You can use reservation policies to group similar reservations together.

2 Assign a Reservation Policy to a Reservation

  You can assign a reservation policy to a reservation when you create the reservation. You can also edit an existing reservation to assign a reservation policy to it, or change its reservation policy assignment.

Procedure

1 Create a Reservation Policy.

2 Assign a Reservation Policy to a Reservation.

Create a Reservation Policy

You can use reservation policies to group similar reservations together.

Create the reservation policy first, then add the policy to reservations to allow a blueprint creator to use the reservation policy in a blueprint.

The policy is created as an empty container.

You can control the display of reservation policies when adding, editing, or deleting by using the **Filter By Type** option on the Reservation Policies page.

Prerequisites

Log in to vRealize Automation as a **fabric administrator**.

Procedure

1. Select **Infrastructure > Reservations > Reservation Policies**.
2. Click the **New** icon (✚).
3. Enter a name in the **Name** text box.
4. Select **Reservation Policy** from the **Type** drop-down menu.
5. Enter a description in the **Description** text box.
6. Click **OK**.

Assign a Reservation Policy to a Reservation

You can assign a reservation policy to a reservation when you create the reservation. You can also edit an existing reservation to assign a reservation policy to it, or change its reservation policy assignment.

Prerequisites

Create a Reservation Policy.

Procedure

1. Select **Infrastructure > Reservations > Reservations**.
2. Point to a reservation and click **Edit**.
3. Select a reservation policy from the **Reservation Policy** drop-down menu.
4. Click **Save**.

Storage Reservation Policies

You can create storage reservation policies to allow blueprint architects to assign the volumes of a virtual machine to different datastores for the vSphere, KVM (RHEV), and SCVMM platform types or different storage profiles for other resources, such as vCloud Air or vCloud Director resources.

Assigning the volumes of a virtual machine to different datastores or to a different storage profile allows blueprint architects to control and use storage space more effectively. For example, they might deploy the operating system volume to a slower, less expensive datastore, or storage profile, and the database volume to a faster datastore or storage profile.

Some machine endpoints only support a single storage profile, while others support multi-level disk storage. Multi-level disk storage is available for vCloud Director 5.6 and greater endpoints and for vCloud Air endpoints. Multi-level disk storage is not supported for vCloud Director 5.5 endpoints.

When you create a blueprint, you can assign a single datastore or a storage reservation policy that represents multiple datastores to a volume. When they assign a single datastore, or storage profile, to a volume, vRealize Automation uses that datastore or storage profile at provisioning time, if possible. When they assign a storage reservation policy to a volume, vRealize Automation uses one of its datastores, or storage profiles if working with other resources, such as vCloud Air or vCloud Director, at provisioning time.

A storage reservation policy is essentially a tag applied to one or more datastores or storage profiles by a fabric administrator to group datastores or storage profiles that have similar characteristics, such as speed or price. A datastore or storage profile can be assigned to only one storage reservation policy at a time, but a storage reservation policy can have many different datastores or storage profiles.

You can create a storage reservation policy and assign it to one or more datastores or storage profiles. A blueprint creator can then assign the storage reservation policy to a volume in a virtual blueprint. When a user requests a machine that uses the blueprint, vRealize Automation uses the storage reservation policy specified in the blueprint to select a datastore or storage profile for the machine’s volume.

Note If you have SDRS enabled on your platform, you can allow SDRS to load balance storage for individual virtual machine disks, or all storage for the virtual machine. If you are working with SDRS datastore clusters, conflicts can occur when you use reservation policies and storage reservation policies. For example, if a standalone datastore or a datastore within an SDRS cluster is selected on one of the reservations in a policy or storage policy, your virtual machine storage might be frozen instead of driven by SDRS. If you request reprovisioning for a machine with storage placement on an SDRS cluster, the machine is deleted if the SDRS automation level is deactivated. For related information about provisioning and SDRS, see the `VirtualMachine.Admin.Datastore.Cluster.ResourceLeaseDurationSec` custom property.

Storage and memory that are assigned to a provisioned machine by a reservation are released when the machine to which they are assigned is deleted in vRealize Automation by the Destroy action. The storage and memory are not released if the machine is deleted in the vCenter Server.

For example, you cannot delete a reservation that is associated with machines in an existing deployment. If you move or delete deployed machines manually in the vCenter Server, vRealize Automation continues to recognize the deployed machines as live and prevents you from deleting associated reservations.

Configure a Storage Reservation Policy

You can create storage reservation policies to group datastores that have similar characteristics, such as speed or price. After you create the storage reservation policy, you must populate it with datastores before using the policy in a blueprint.

Procedure

1 Create a Storage Reservation Policy

You can use a storage reservation policy to group datastores that have similar characteristics, such as speed or price.

2 Assign a Storage Reservation Policy to a Datastore

You can associate a storage reservation policy to a compute resource. After the storage reservation policy is created, populate it with datastores. A datastore can belong to only one storage reservation policy. Add multiple datastores to create a group of datastores for use with a blueprint.

Procedure

1 Create a Storage Reservation Policy.

2 Assign a Storage Reservation Policy to a Datastore.

Create a Storage Reservation Policy

You can use a storage reservation policy to group datastores that have similar characteristics, such as speed or price.

The policy is created as an empty container.

You can control the display of reservation policies when adding, editing, or deleting by using the **Filter By Type** option on the Reservation Policies page.

Prerequisites

Log in to vRealize Automation as a **fabric administrator**.

Procedure

1 Select **Infrastructure > Reservations > Reservation Policies**.

2 Click the **New** icon (✚).

3 Enter a name in the **Name** text box.

4 Select **Storage Reservation Policy** from the **Type** drop-down menu.

5 Enter a description in the **Description** text box.

6 Click **OK**.

Assign a Storage Reservation Policy to a Datastore

You can associate a storage reservation policy to a compute resource. After the storage reservation policy is created, populate it with datastores. A datastore can belong to only one storage reservation policy. Add multiple datastores to create a group of datastores for use with a blueprint.

Prerequisites

Create a Storage Reservation Policy.

Procedure

1 Select **Infrastructure > Compute Resources > Compute Resources**.

2 Point to a compute resource and click **Edit**.

3 Click the **Configuration** tab.

4 Locate the datastore to add to your storage reservation policy in the Storage table.

5 Click the **Edit** icon (✎) next to the desired **Storage Path** object.

6 Select a storage reservation policy from the **Storage Reservation Policy** column drop-down menu.

   After you provision a machine, you cannot change its storage reservation policy if doing so would change the storage profile on a disk.

7 Click **OK**.

8 (Optional) Assign additional datastores to your storage reservation policy.

9 Click **OK**.

Workload Placement

When you deploy a blueprint, workload placement uses collected data to recommend where to deploy the blueprint based on available resources. vRealize Automation and vRealize Operations Manager work together to provide placement recommendations for workloads in the deployment of new blueprints.

While vRealize Automation manages organizational policies, such as business groups, reservations, and quotas, it integrates with the capacity analytics of vRealize Operations Manager to place machines. Workload placement is only available for vSphere endpoints.

Workload Placement Terms Used

Several terms are used with workload placement.

- Clusters in vSphere map to compute resources in vRealize Automation.
- Reservations include compute and storage, where the storage can consist of individual datastores or datastore clusters. A reservation can include multiple datastores, datastore clusters, or both.
- Multiple reservations can refer to the same cluster.
- Virtual machines can move to multiple clusters.

- When workload placement is enabled, the provisioning workflow uses the placement policy to recommend where to deploy the blueprint.

Provisioning Blueprints with Workload Placement

When you use workload placement to provision blueprints, the provisioning workflow uses the reservations in vRealize Automation, and the placement optimization from vRealize Operations Manager.

1. vRealize Automation provides the governance rules to allow placement destinations.
2. vRealize Operations Manager provides placement optimization recommendations according to analytics data.
3. vRealize Automation continues the provisioning process according to the placement recommendations from vRealize Operations Manager.

If vRealize Operations Manager cannot provide a recommendation, or the recommendation cannot be used, then vRealize Automation falls back to its default placement logic.

When a developer selects a catalog item and completes the form to request the catalog item, vRealize Automation accounts for the following considerations to provision the virtual machines.

Table 4-16. Considerations to Provision Virtual Machines

Consideration	Effect
Policies	The vRealize Automation reservation policy might indicate more than one reservation.
Reservations	vRealize Automation evaluates the request, and determines which reservations can satisfy the constraints made in the request. ■ If placement is enabled and based on vRealize Operations Manager analytics, vRealize Automation passes the list of reservations to vRealize Operations Manager to determine which reservation is best suited for placement based on operational metrics. ■ If placement is not based on vRealize Operations Manager, vRealize Automation decides the placement based on priorities and availability. The reservations are updated to track that resources have been consumed. If vRealize Operations Manager recommends a cluster or datastore that vRealize Automation considers to be out of capacity or no longer applicable, vRealize Automation logs the exception. vRealize Automation allows provisioning to continue according to its default placement mechanisms.

To identify resources for a virtual machine, vRealize Automation provides a list of candidate reservations. Each candidate in the list can include a cluster and one or more datastores or datastore clusters. vRealize Operations Manager uses the candidate reservations to create the list of destination candidates and locate the best target.

The policy in vRealize Operations Manager sets the level of balance, utilization, and buffer space for the cluster. For a single reservation, which is a cluster or datastore cluster, vRealize Automation validates whether the recommendation is viable placement destination.

- If the destination is viable, vRealize Automation deploys the blueprint according to the recommendation.

- If the destination is not viable, vRealize Automation uses the default placement behavior to place the virtual machines.

Placement considerations must also account for health and utilization problems. While the cloud administrator and virtual infrastructure administrator manage the infrastructure, developers care about the health of their applications. To support developers, the workload placement strategy must also consider health and utilization problems.

Table 4-17. Considerations for Health and Utilization Problems

Workload Problem	Placement Solution
Developer notices a health problem in the environment.	vRealize Automation is provisioning blueprints in clusters that are experiencing problems, or that are overutilized because of large workloads. vRealize Automation must integrate with the capacity analytics in vRealize Operations Manager to ensure that blueprints are provisioned in clusters that have sufficient capacity.
Developer notices a utilization problem.	The clusters in the environment are underutilized. vRealize Automation must integrate with the capacity analytics that vRealize Operations Manager provides to ensure that blueprints are provisioned in a cluster where the utilization is maximized.

Users Who Provision Blueprints

The following users perform actions to provision blueprints.

Table 4-18. Users and Roles to Provision Blueprints

Step	User	Action	Role Required
1	Cloud Administrator or Virtual Infrastructure (VI) Administrator	Ensures that the initial placement of virtual machines meets organizational policies, and that they are optimized according to the operational analytics data.	IaaS Admin role
1	Fabric Administrator	Defines the reservations, reservation policies, and placement policy in vRealize Automation.	Fabric Administrator role, Infrastructure Architect
1	IaaS Administrator	Defines the endpoints for vSphere and vRealize Operations Manager, which are necessary for workload placement.	IaaS Admin role
2	Infrastructure Architect	As a blueprint architect who works directly with virtual machine component types, assigns the reservation policies to virtual machines when authoring a blueprint. Specifies the reservation policy as a property of the machine component in the blueprint.	Infrastructure Architect

Table 4-18. Users and Roles to Provision Blueprints (continued)

Step	User	Action	Role Required
3	Infrastructure Architect, Application Architect, Software Architect, and XaaS Architect	Creates and publishes the blueprint to provision the virtual machines. Only the Infrastructure Architect works directly with machine components. The other architect roles can reuse infrastructure blueprints in nesting, but they cannot edit the machine component settings. The blueprint can include a single component, or it can include nested blueprints, XaaS components, multiple virtual machines in a multi-tier application, and so on. vRealize Automation places the virtual machines according to the configuration of the reservations, and optionally includes the reservation policy at the machine component level for the blueprint. For example, your blueprint might include two machines, with a different policy applied to each machine. vRealize Automation also optimizes the virtual machines according to the operational analytics data that vRealize Operations Manager provides.	Infrastructure Architect
4	Cloud Administrator or VI Administrator	Selects the policies that govern the initial placement of the virtual machines that vRealize Automation provisions. The Administrator can: ■ Select the policies by using an API. ■ Use the default placement policy, which uses each server in vRealize Automation in turn to balance workloads. This approach does not require input from vRealize Operations Manager.	IaaS Admin role, Infrastructure Architect
5	VI Administrator	Builds the custom data center and custom groups in vRealize Operations Manager. Then, the VI Administrator applies the policy used to consolidate and balance workloads to those custom data centers.	IaaS Admin role, Infrastructure Architect
6	Fabric Administrator	Selects the placement policy in vRealize Automation. Use the workload placement policy to have vRealize Automation determine where to place machines when you deploy new blueprints. The placement policy requires input from vRealize Operations Manager	Fabric Administrator role
7	Developer	Requests a blueprint to provision virtual machines. The blueprint can consist of multiple machines to run a three-tier application.	
8	Developer	When the developer deploys the blueprint, vRealize Operations Manager searches for a placement policy that fits the relevant clusters for the request.	

For more information about the placement policy, see Placement Policy.

To configure workload placement, see Configuring Workload Placement.

Distributed Resource Scheduler (DRS) Is Required to Place Virtual Machines

vSphere DRS is the placement engine that vRealize Automation and vRealize Operations Manager use to provision and place virtual machines.

For vRealize Automation to suggest the best placement for the virtual machines, you must enable DRS on the cluster, and set it to fully automated. vRealize Automation then uses the vSphere DRS APIs to determine the correct placement for the virtual machines.

vRealize Automation integrates with the vRealize Operations Manager placement service. vRealize Operations Manager only provides placement recommendations for clusters that have DRS enabled and fully automated.

Effect of vRealize Automation Storage Reservation Policies

The presence of vRealize Automation storage reservation policies affects workload placement with vRealize Operations Manager.

When workload placement with vRealize Operations Manager is enabled, vRealize Automation passes a list of available reservations to vRealize Operations Manager, and vRealize Operations Manager evaluates them for storage placement based on operational analysis.

Note Workload placement with vRealize Operations Manager only supports virtual machines with one or more disks, where just one storage reservation policy is present. Multiple policy combinations are not supported for disk placement because individual disk placement is not supported.

When a blueprint contains storage reservation policies, workload placement recommendations from vRealize Operations Manager change in the following ways:

Configuration	Placement
Virtual machines with one or more disks, where none specifies a storage reservation policy	Placement occurs as usual. vRealize Operations Manager evaluates the full, unfiltered list of candidate reservations.
Virtual machines with one or more disks, where all specify the same storage reservation policy	Candidate reservations are filtered at the storage level so that vRealize Operations Manager only evaluates datastores that match that storage reservation policy.
Virtual machines with multiple disks, where some specify the same storage policy but others specify no storage reservation policy	■ When the storage allocation type is COLLECTED, the default, all disks are treated as if they share that same policy. vRealize Operations Manager evaluates datastores that match that storage reservation policy. ■ When the storage allocation type is DISTRIBUTED, virtual machines cannot be placed according to vRealize Operations Manager recommendations because individual disk placement is not supported. Placement defaults to vRealize Automation placement algorithms instead. You can set the storage allocation type by using a custom property.

Configuration	Placement
Virtual machines with multiple disks, where the disks specify different storage reservation policies	Because they have conflicting storage reservation policy requirements, these virtual machines cannot be placed according to vRealize Operations Manager recommendations. Placement defaults to vRealize Automation placement algorithms instead.
Virtual machines that require a specific storage path	These virtual machines are not placed through a vRealize Operations Manager recommendation because you already specified a storage path. Placement might or might not match what vRealize Operations Manager would have recommended. You can set the storage path by using a custom property.

Placement Errors—When vRealize Operations Manager based placement cannot occur, an error describes the reason. Reasons might include the unsupported conditions described in the preceding list, or environmental factors such as failed communication between vRealize Operations Manager and vRealize Automation.

To review errors, go to **Requests > Execution**. Near the upper right, click **View Placement Errors**.

Limitations on Workload Placement

When you use the placement policy for workload placement to place machines when you deploy new blueprints, be aware of the limitations.

- In vRealize Operations Manager, the vRealize Automation solution identifies the clusters and virtual machines that vRealize Automation manages.
- When vRealize Automation manages the child objects of a data center or custom data center container in vRealize Operations Manager, the ability to rebalance or move those objects is not available. You cannot turn on or turn off the exclusion of actions on vRealize Automation managed objects.
- For the objects that vRealize Automation manages, the workload placement behavior is as follows:
  - When a custom data center or data center includes a cluster that vRealize Automation manages, workload placement does not allow you to rebalance the cluster.
  - When a cluster includes virtual machines that vRealize Automation manages, workload placement does not allow you to move those virtual machines.
- vRealize Operations Manager does not support workload placement on resource pools in vCenter Server.
- vRealize Operations Manager 7.5 and greater supports vSAN datastores for workload placement. For related information, see vRealize Operations Manager7.5 release notes.

Permissions to Configure Workload Placement

You must have permissions in vRealize Automation and vRealize Operations Manager to configure workload placement and the placement policy.

To configure workload placement in vRealize Automation, you must have the Fabric Administrator role. See User Roles Overview.

In vRealize Operations Manager, you must create a user role for workload placement, and assign permissions to the role.

- On the user account, assign the read-only permission to vSphere Hosts and Clusters, and vSphere Storage, in the object hierarchy.
- To have the user role use API calls in workload placement, assign read and write permissions on APIs. Select **Administration > Access Control > Permissions**, and select **REST APIs > All other Read, Write APIs**.

vRealize Automation uses the vRealize Operations Manager role when you register the endpoint, and to request placement recommendations during provisioning on behalf of users who request catalog items.

For more information, see Access Control in the vRealize Operations Manager Information Center.

Placement Policy

You can use the placement policy to have vRealize Automation determine where to place machines when you deploy new blueprints. The placement policy uses the analytics of vRealize Operations Manager to identify workloads on your clusters so that it can suggest placement destinations.

You must perform several steps before you can use the placement policy. In vRealize Automation, you create endpoints for the vRealize Operations Manager and vCenter Server instances. Then, you create a fabric group, and add reservations to your vCenter Server endpoint.

To ensure that vRealize Operations Manager provides workload placement analytics to vRealize Automation, you must:

- Install the vRealize Automation Solution in the vRealize Operations Manager instance that is being used for workload placement.
- Configure vRealize Operations Manager to monitor the vCenter Server.

To configure vRealize Automation and vRealize Operations Manager for workload placement, see Configuring Workload Placement.

Locating the Placement Policy

In your vRealize Automation instance, select **Infrastructure > Reservations > Placement Policy**.

To use the workload placement analytics that vRealize Operations Manager provides, select **Use vRealize Operations Manager for placement recommendations**

If you do not use the workload placement policy, vRealize Automation uses default placement method.

Configuring Workload Placement

To use the placement policy to place machines when you deploy new blueprints, you configure vRealize Automation to use the analytics that vRealize Operations Manager provides. You also configure vRealize Operations Manager to apply a policy to consolidate and balance workloads to your cluster compute resources.

In vRealize Automation, you configure endpoints, create a fabric group, and add reservations. In vRealize Operations Manager, you configure a policy to support workload balance, and apply that policy to a custom group that includes your custom compute resources.

Prerequisites

Before the placement policy can suggest placement destinations for blueprints, you must perform several steps.

- Understand the placement policy. See Placement Policy.
- Verify that an endpoint exists in vRealize Automation for the vRealize Operations Manager instance being used for workload placement. See Create a vRealize Operations Manager Endpoint.
- Verify that an endpoint exists in vRealize Automation for the vCenter Server instance. See Create a vSphere Endpoint in vRealize Automation and Associate to NSX .
- Add reservations to the vCenter Server endpoint. See Reservations.
- Add a fabric group, and verify that your user is a fabric group administrator. See Create a Fabric Group.
- Verify that vRealize Operations Manager is monitoring the same infrastructure that vRealize Automation is monitoring, to ensure that they include the same vCenter Server instances. See VMware vSphere Solution in vRealize Operations Manager in the vRealize Operations Manager Information Center.
- Understand reservations, storage reservation, blueprints, and delegate providers. See other related topics in the vRealize Automation Information Center.
- Understand and define the fill and balance settings in the vRealize Operations Manager policy used for workload placement. See Workload Automation Details in the vRealize Operations Manager Information Center.

Procedure

1. Configure vRealize Automation for Workload Placement

   To use workload placement analytics to place machines when you deploy new blueprints, you must prepare the vRealize Automation instance.

2. Configure vRealize Operations Manager for Workload Placement in vRealize Automation

   To provide workload placement analytics to vRealize Automation to place machines when you deploy new blueprints, you must prepare the vRealize Operations Manager instance.

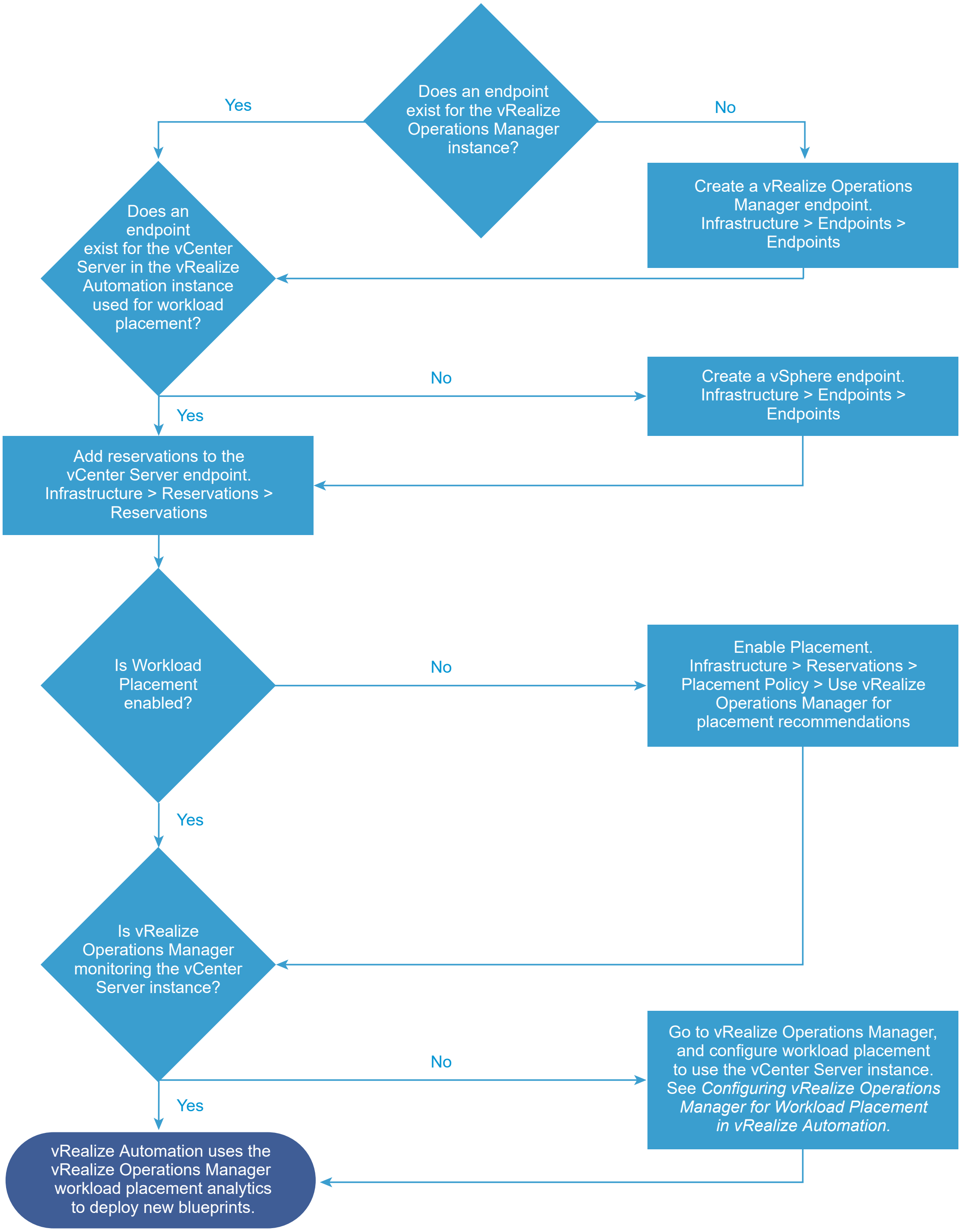
Results

You configured vRealize Automation and vRealize Operations Manager to use workload placement analytics to suggest placement destinations for new blueprints.

Configure vRealize Automation for Workload Placement

To use workload placement analytics to place machines when you deploy new blueprints, you must prepare the vRealize Automation instance.

To prepare your vRealize Automation instance to use the placement policy, you configure endpoints, create a fabric group, and add reservations.


Does an endpoint exist for the vRealize Operations Manager instance?
Yes
No
Create a vRealize Operations Manager endpoint. Infrastructure > Endpoints > Endpoints
Does an endpoint exist for the vCenter Server in the vRealize Automation instance used for workload placement?
No
Create a vSphere endpoint. Infrastructure > Endpoints > Endpoints
Yes
Add reservations to the vCenter Server endpoint. Infrastructure > Reservations > Reservations
Is Workload Placement enabled?
No
Enable Placement. Infrastructure > Reservations > Placement Policy > Use vRealize Operations Manager for placement recommendations
Yes
Is vRealize Operations Manager monitoring the vCenter Server instance?
No
Go to vRealize Operations Manager, and configure workload placement to use the vCenter Server instance. See *Configuring vRealize Operations Manager for Workload Placement in vRealize Automation.*
Yes
vRealize Automation uses the vRealize Operations Manager workload placement analytics to deploy new blueprints.

Prerequisites

- To use workload placement, understand the requirements. See Configuring Workload Placement.
- In vRealize Automation, add a specific user role and permissions for vRealize Operations Manager to validate credentials. See User Roles Overview.

Procedure

1. In your vRealize Automation instance, add an endpoint for the vRealize Operations Manager instance, and click **OK**.

   a. Select **Infrastructure > Endpoint > Endpoints**.

   b. Select **New > Management > vRealize Operations Manager**.

   c. Enter the general information for the **vRealize Operations Manager** endpoint.

      You do not need to specify properties for the endpoint.

2. In your vRealize Automation instance, add an endpoint for the vCenter Server instance, and click **OK**.

   a. Select **Infrastructure > Endpoint > Endpoints**.

   b. Select **New > Virtual > vSphere (vCenter)**.

   c. Enter the general information, properties, and associations for the vCenter Server endpoint.

   After you add endpoints, and vRealize Automation collects data from them, the compute resources for those endpoints are available. You can then add those compute resources to the fabric group that you create.

3. Create a fabric group so that other users can create reservations and enable the placement policy.

   a. Select **Infrastructure > Endpoint > Fabric Groups**.

   b. Click **New**, and enter information about the fabric group.

Option	Description
Name	Enter a meaningful name for the fabric group.
Description	Enter a useful description.
Fabric administrators	Enter the email address for each person to designate as a fabric administrator.
Compute resources	Select the compute resource clusters that the administrators can manage.

   After you add compute resources to a fabric group, and vRealize Automation collects data from them, fabric administrators can create reservations for the compute resources.

4 Create reservations for the compute resources in the vCenter Server instance.

   a Select **Infrastructure > Reservations > Reservations**.

   b Select **New > vSphere (vCenter)**.

   c On each tab, enter the information for the reservation.

Option	Action
General	Select a reservation policy, the priority for the policy, and click **Enable this reservation**.
Resources	Select the machine quota, memory, and storage. You do not have to select a resource pool.
Network	Select the network adapter. You do not have to select a network profile.
Properties	If necessary, add custom properties to the reservation.
Alert	If necessary, select **Capacity alerts** to notify recipients when the capacity exceeds the threshold for the reservation.

5 Enable the placement policy.

   a Select **Infrastructure > Reservations > Placement Policy**.

   b Select the check box named **Use vRealize Operations Manager for placement recommendations**.

Results

You configured vRealize Automation to use the analytics of vRealize Operations Manager to place machines when users deploy blueprints.

What to do next

Configure vRealize Operations Manager to monitor the vCenter Server instance, and apply a workload placement policy to your cluster compute resources. See Configure vRealize Operations Manager for Workload Placement in vRealize Automation.

Configure vRealize Operations Manager for Workload Placement in vRealize Automation

To provide workload placement analytics to vRealize Automation to place machines when you deploy new blueprints, you must prepare the vRealize Operations Manager instance.

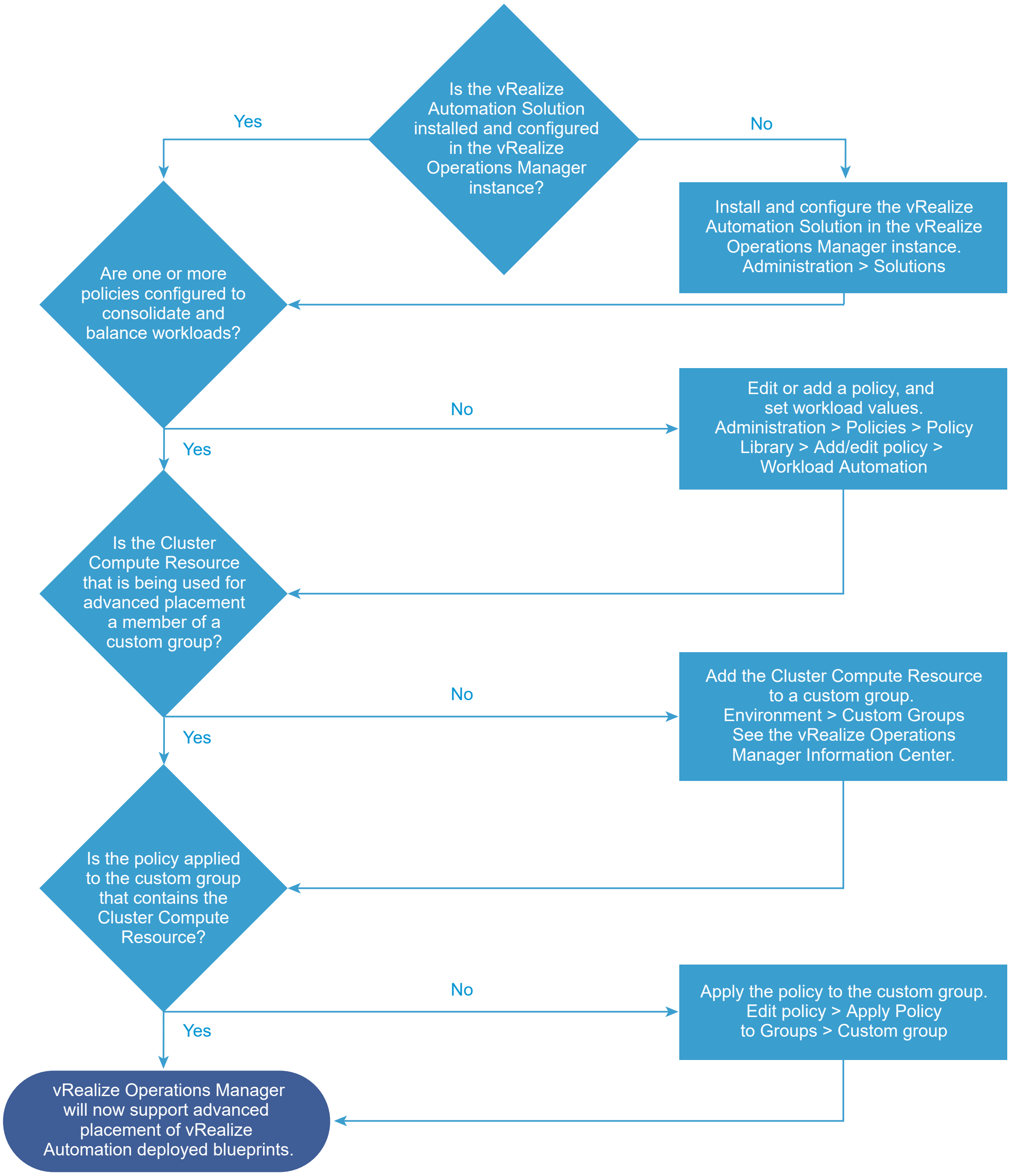
Caution You must install the vRealize Automation solution, which includes the management pack, on only a single vRealize Operations Manager instance.

To prepare your vRealize Operations Manager instance to provide analytics to vRealize Automation, you install and configure the vRealize Automation solution. You must also configure a policy, and apply the policy to your cluster compute resources.

After you configure the vRealize Automation solution, you cannot move or rebalance any virtual machines that vRealize Automation manages.

If the vRealize Automation solution is not installed in the vRealize Operations Manager instance, workload placement can still move or rebalance virtual machines that vRealize Automation manages.

To allow workload placement to move virtual machines, those virtual machines must reside in a data center or custom data center.


Is the vRealize Automation Solution installed and configured in the vRealize Operations Manager instance?
Yes
No
Install and configure the vRealize Automation Solution in the vRealize Operations Manager instance. Administration > Solutions
Are one or more policies configured to consolidate and balance workloads?
No
Edit or add a policy, and set workload values. Administration > Policies > Policy Library > Add/edit policy > Workload Automation
Yes
Is the Cluster Compute Resource that is being used for advanced placement a member of a custom group?
No
Add the Cluster Compute Resource to a custom group. Environment > Custom Groups See the vRealize Operations Manager Information Center.
Yes
Is the policy applied to the custom group that contains the Cluster Compute Resource?
No
Apply the policy to the custom group. Edit policy > Apply Policy to Groups > Custom group
Yes
vRealize Operations Manager will now support advanced placement of vRealize Automation deployed blueprints.

Prerequisites

- Configure vRealize Automation to use workload placement analytics. See Configure vRealize Automation for Workload Placement.

- Verify that the vRealize Automation Solution is installed and configured in the vRealize Operations Manager instance that is being used for workload placement. For details about this solution, see the Management Pack for vRealize Automation on Solution Exchange. For information about how workload placement works in vRealize Operations Manager, see Workload Automation Details and related topics in the vRealize Operations Manager documentation.

Procedure

1 In the instance of vRealize Operations Manager that manages workload placement, install and configure the vRealize Automation solution.

   The solution might already be installed.

   a To see the solutions that are installed in vRealize Operations Manager, click **Administration > Solutions**.

   b Verify whether the vRealize Automation solution is already installed.

     If the vRealize Automation solution does not appear in the list, download and install the solution. See Management Pack for vRealize Automation on Solution Exchange.

   c If the solution appears in the list, select the **VMware vRealize Automation solution**, and click **Configure**.

   d Configure the vRealize Automation solution, and save the settings.

     For more information to configure the solution, see Solutions in vRealize Operations Manager in the vRealize Operations Manager Information Center.

2 If you do not use the vRealize Operations Manager Default Policy, you must create a custom group. Then, add your cluster compute resources to the custom group.

   To apply a policy other than Default Policy to your clusters, add a custom group. You then apply the policy to the custom group. If you use Default Policy, you do not have to create a custom group, because Default Policy applies to all objects.

   a Click **Environment > Custom Groups**.

   b If a custom group does not exist for your clusters, create a custom group.

     For details, see User Scenario: Creating Custom Object Groups in the vRealize Operations Manager Information Center.

   c Add the cluster to the custom group, and save the custom group.

3 Configure a policy to consolidate and balance workloads on your clusters, and apply that policy to the custom group.

   You configure a policy in vRealize Operations Manager to establish the settings for consolidation, balance, fill, CPU, memory, and disk space. For example, you modify the setting named Consolidate Workloads to determine the best placement for new managed workloads based on the cluster status and capacity. You also modify the threshold setting for Balance Workloads to the level of aggressiveness required to place workloads. You can configure one or more policies, and apply them to your cluster compute resources.

a To locate the policies, click **Administration > Policies > Policy Library** .

b To set workload values, click **Add/Edit Policy**, and click **Workload Automation**.

  The settings named Consolidate Workloads and Cluster Headroom apply to the initial placement of virtual machines.

  - When you set Consolidate Workloads to **none**, workload placement balances the workload across all the clusters to which the policy is applied. When you set Consolidate Workloads to a value other than none, workload placement fills the busiest cluster first.
  - Cluster Headroom is the buffer space reserved in a cluster, as a percentage of the total capacity. For example, if you set the cluster headroom to 20%, that buffer might prevent workload placement from placing virtual machines on that cluster. The reason it prevents the placement is because the cluster has 20% less of the free capacity for CPU, memory, or disk space.

c In the policy workspace, click **Apply Policy to Groups**.

d Select the custom group.

e Save the policy.

Results

You configured vRealize Operations Manager so that vRealize Automation uses the workload placement analytics to suggest placement destinations of machines when users deploy blueprints.

What to do next

Wait for vRealize Automation and vRealize Operations Manager to collect data from the endpoints and objects in your environment. Then, when you deploy new blueprints, vRealize Automation displays the workload placement recommendations, destination candidates, and selected placement for your confirmation.

Troubleshooting Workload Placement

If you experience problems with workload placement, use the troubleshooting information to resolve them.

The vRealize Automation Solution Is Required for Workload Placement to Operate Properly

Workload placement is based on individual machines, and placement is done at the machine level. When vRealize Automation and vRealize Operations Manager are installed together, the vRealize Automation Solution must also be installed.

The solution, which includes the management pack and adapter, identifies the clusters on which the `rebalance container` or `move VM` actions are deactivated. The rebalance action is deactivated on the custom data center to which the cluster belongs.

- For unmanaged vRealize Automation clusters that belong to a custom data center that does not have any managed vRealize Automation clusters, the `move VM` and `rebalance container` actions are enabled. For managed vRealize Automation clusters, these actions are deactivated.
- In vRealize Operations Manager, the vRealize Automation Adapter causes VMs on clusters that map reservations not to be available for move or rebalance.

Caution The vRealize Automation solution must only be installed on a single vRealize Operations Manager instance.

High Availability Is Enabled, but Must Be Deactivated

When HA is enabled, if vRealize Operations Manager is down, the timeout used for workload placement to call vRealize Operations Manager might fail.

vRealize Automation logs workload placement errors in the `catalina.out` log file.

vSphere Endpoints in vRealize Automation Are Not Monitored

vRealize Operations Manager is not monitoring the vSphere vCenter Server instance that contains the reservation clusters.

If vRealize Operations Manager does not recognize the vRealize Automation candidate reservations for a cluster, datastore, or datastore cluster when it attempts to place them, it ignores them. In the placement response, vRealize Operations Manager communicates to vRealize Automation that it does not recognize them.

As a result, in the placement details on the request execution, vRealize Automation displays a warning icon on the candidate reservation to indicate that it is unrecognized.

When Mismatches Occur, vRealize Automation Appears at the Top of the List

vRealize Automation and vRealize Operations Manager manage different views of the infrastructure. But they must both manage the same instances of vCenter Server in the same infrastructure.

Must identify disconnects and mismatches, and display details.

What to Do If the vRealize Automation Adapter Is Down

The initial placement always honors the list of destination candidates that it receives from vRealize Operations Manager, such as when a user adds a cluster immediately after installation.

If the vRealize Automation solution, which includes the management pack and adapter, is not available in the vRealize Operations Manager, the `move VM` and `rebalance container` actions are available.

Continuous Optimization Using vRealize Operations Manager

Continuous optimization provides ongoing, autonomous management of vRealize Automation workloads by vRealize Operations Manager.

With continuous optimization, you take advantage of workload rebalancing and relocation, and use vRealize Automation with vRealize Operations Manager beyond initial workload placement. As virtualization resources move or come under heavier or lighter load, vRealize Automation provisioned workloads can move as needed.

- Continuous optimization automatically creates a new datacenter in vRealize Operations Manager.

  There is one new datacenter for each vRealize Automation vCenter endpoint.

- The newly created datacenter contains every vRealize Automation managed cluster associated with the endpoint.

  Note Do not manually create a mixed datacenter of vRealize Automation and non-vRealize Automation clusters.

- You can run continuous optimization only from the newly created vRealize Automation based datacenter.
- Optimization does not support different reservation requirements among clusters in the vCenter, which might occur when you have different business groups.

  Optimization is at the vRealize Automation based datacenter level, and different reservation requirements across clusters can prevent success. When that happens, an error appears, stating that some destination clusters or storage did not meet requirements, which prevented some optimization moves.

- Optimization never creates a new vRealize Automation or vRealize Operations Manager policy violation.
  - If you have existing policy violations, optimization can fix vRealize Operations Manager Operational Intent issues.
  - If you have existing policy violations, optimization cannot fix vRealize Operations Manager Business Intent issues.

    For example, if you manually moved a virtual machine to a cluster that was not a part of its reservation policy, vRealize Operations Manager does not detect a violation nor try to resolve it. To fix Business Intent issues, you must use vRealize Automation to move the workload.

- This release obeys Operational Intent at the datacenter level. All member vRealize Automation clusters are optimized to the same settings.

  To set a different Operational Intent for clusters, you must configure them in separate vRealize Automation datacenters, associated with separate vCenter endpoints. Having different test and production clusters might be one example situation.

- vRealize Operations Manager queries vRealize Automation for allowable placement based on vRealize Automation policies and reservations.
- vRealize Operations Manager placement tags cannot be applied to vRealize Automation provisioned workloads.

In addition, scheduled optimization involving multiple machines is supported. Regularly scheduled optimizations are not all-or-nothing processes. If conditions interrupt machine movement, successfully relocated machines stay relocated, and the next vRealize Operations Manager cycle attempts to relocate the remainder as is usual for vRealize Operations Manager. Such a partially completed optimization causes no negative effect in vRealize Automation.

Locate Unbalanced Workloads in vRealize Automation

vRealize Automation can reveal when workloads are heavily provisioned to the same cluster.

Procedure

1. To see where workloads are provisioned, click **Infrastructure > Compute Resources > Compute Resources**.

   Take note of any uneven machine placement.

2. Reservations can cause heavy provisioning to the same cluster. To review reservations, click **Infrastructure > Reservations > Reservations**.

   Note the priority and how that might affect machine placement.

Enabling Continuous Optimization

When you add the vRealize Automation adapter in vRealize Operations Manager, vRealize Operations Manager automatically creates a new, dedicated datacenter for vRealize Automation based workloads.

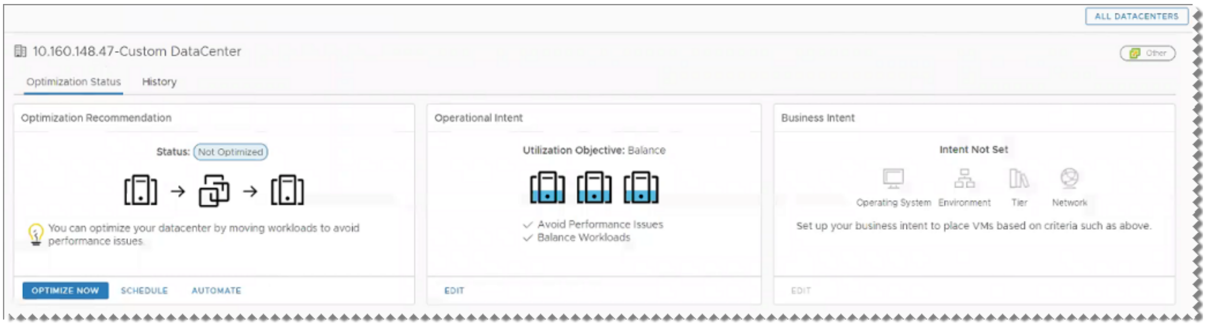
Other than adding the adapter, there are no separate installation steps for continuous optimization. You may begin configuring and using vRealize Operations Manager for workload relocation in the new datacenter. See the Continuous Optimization Example.

Continuous Optimization Example

The following example shows a rebalancing workflow for vRealize Automation continuous optimization with vRealize Operations Manager.

1. From the vRealize Operations Manager home page, click **Workload Optimization**.
2. Select the automatically created vRealize Automation datacenter.
3. Under **Operational Intent**, click **Edit**, and select **Balance**.

   You cannot select or edit Business Intent, which is deactivated when the datacenter is for vRealize Automation optimization.



You can optimize your datacenter by moving workloads to avoid performance issues.

Set up your business intent to place VMs based on criteria such as above.

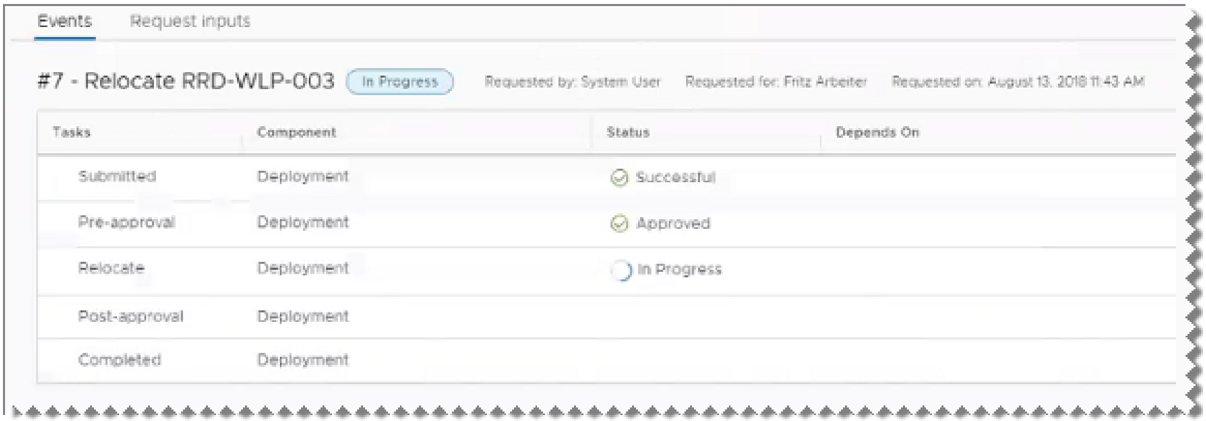
4 Under **Optimization Recommendation**, click **Optimize Now**.

vRealize Operations Manager displays a before-and-after diagram of the proposed operation.

5 Click **Next**.

6 Click **Begin Action**.

7 In vRealize Automation, monitor the operation in progress by clicking **Deployments** and looking at event status.

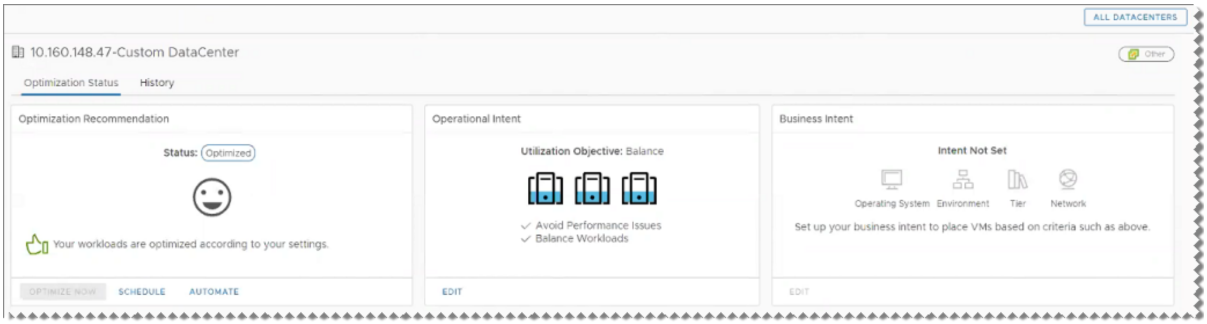


#7 - Relocate RRD-WLP-003 In Progress Requested by: System User Requested for: Fritz Arbeiter Requested on: August 13, 2018 11:43 AM

Tasks	Component	Status	Depends On
Submitted	Deployment	Successful	
Pre-approval	Deployment	Approved	
Relocate	Deployment	In Progress	
Post-approval	Deployment		
Completed	Deployment		

When rebalancing finishes, vRealize Automation refreshes. The Compute Resources page shows that machines have moved.

In vRealize Operations Manager, the next data collection refreshes the display to show that optimization is complete.



Your workloads are optimized according to your settings.

Set up your business intent to place VMs based on criteria such as above.

In vRealize Operations Manager, you can review the operation by clicking **Administration > History > Recent Tasks**.

Locate vRealize Automation Datacenters in vRealize Operations Manager

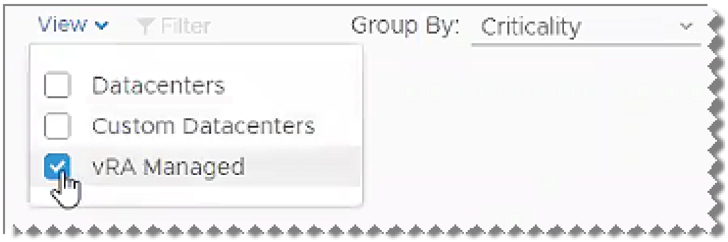
You can use vRealize Operations Manager to display only the vRealize Automation managed datacenters.

Procedure

1 From the vRealize Operations Manager home page, click **Workload Optimization**.

2 Near the top right, click the **View** drop-down.

3 Select only the vRealize Automation managed datacenters.



Managing Key Pairs

Key pairs are used to provision and connect to a cloud instance. A key pair is used to decrypt Windows passwords or to log in to a Linux machine.

Key pairs are required for provisioning with Amazon Web Services. For Red Hat OpenStack, key pairs are optional.

Existing key pairs are imported as part of data collection when you add a cloud endpoint. A fabric administrator can also create and manage key pairs by using the vRealize Automation console. If you delete a key pair from the vRealize Automation console, it is also deleted from the cloud service account.

In addition to managing key pairs manually, you can configure vRealize Automation to generate key pairs automatically per machine or per business group.

- A fabric administrator can configure the automatic generation of key pairs at a reservation level.
- If the key pair is going to be controlled at the blueprint level, the fabric administrator must select **Not Specified** on the reservation.
- A tenant administrator or business group manager can configure the automatic generation of key pairs at a blueprint level.
- If key pair generation is configured at both the reservation and blueprint level, the reservation setting overrides the blueprint setting.

Create a Key Pair

You can create key pairs for use with endpoints by using vRealize Automation.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- Create a cloud endpoint and add your cloud compute resources to a fabric group. See Choosing an Endpoint Scenario and Create a Fabric Group.

Procedure

1. Select **Infrastructure > Reservations > Key Pairs**.
2. Click **New**.
3. Enter a name in the **Name** text box.
4. Select a cloud region from the **Compute resource** drop-down menu.
5. Click **OK**.

Results

The key pair is ready to use when the Secret Key column has the value ************.

Upload the Private Key for a Key Pair

You can upload the private key for a key pair in PEM format.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- You must already have a key pair. See Create a Key Pair.

Procedure

1. Select **Infrastructure > Reservations > Key Pairs**.
2. Locate the key pair for which you want to upload a private key.
3. Click the **Edit** icon (✎).
4. Use one of the following methods to upload the key.
   - Browse for a PEM-encoded file and click **Upload**.
   - Paste the text of the private key, beginning with `-----BEGIN RSA PRIVATE KEY-----` and ending with `-----END RSA PRIVATE KEY-----`.
5. Click the **Save** icon (✓).

Export the Private Key from a Key Pair

You can export the private key from a key pair to a PEM-encoded file.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.

- A key pair with a private key must exist. See Upload the Private Key for a Key Pair.

Procedure

1 Select **Infrastructure > Reservations > Key Pairs**.

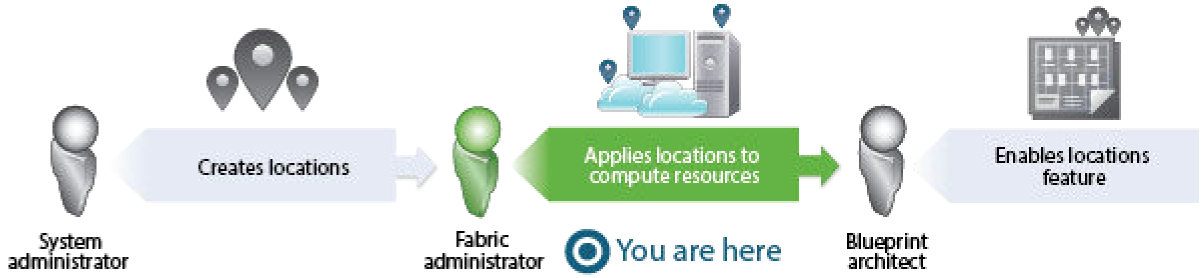
2 Locate the key pair from which to export the private key.

3 Click the **Export** icon (📤).

4 Browse to the location that you want to save the file and click **Save**.

Scenario: Apply a Location to a Compute Resource for Cross Region Deployments

As a fabric administrator, you want to label your compute resources as belonging to your Boston or London datacenter to support cross region deployments. When your blueprint architects enable the locations feature on their blueprints, users are able to choose whether to provision machines in your Boston or London datacenter.



You have a datacenter in London, and a datacenter in Boston, and you don't want users in Boston provisioning machines on your London infrastructure or vice versa. To ensure that Boston users provision on your Boston infrastructure, and London users provision on your London infrastructure, you want to allow users to select an appropriate location for provisioning when they request machines.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator**.
- As a system administrator, define the datacenter locations. See Scenario: Add Datacenter Locations for Cross Region Deployments .

Procedure

1 Select **Infrastructure > Compute Resources > Compute Resources**.

2 Point to the compute resource located in your Boston datacenter and click **Edit**.

3 Select Boston from the **Locations** drop-down menu.

4 Click **OK**.

5 Repeat this procedure as necessary to associate your compute resources to your Boston and London locations.

Results

IaaS architects can enable the locations feature so users can choose to provision machines in Boston or London when they fill out their catalog item request forms. See Enable Users to Select Datacenter Locations for Cross Region Deployments .

Provisioning a vRealize Automation Deployment Using a Third-Party IPAM Provider

You can obtain IP addresses and ranges for use in a vRealize Automation network profile from a supported third-party IPAM solution provider, such as Infoblox.

The IP address ranges in the network profile are used in an associated reservation, which you specify in a blueprint. When an entitled user requests machine provisioning using the blueprint catalog item, an IP address is obtained from the third-party IPAM-specified range of IP addresses. After machine deployment, you can discover the IP address used by querying its vRealize Automation item details page.

Table 4-19. Preparing for Provisioning a vRealize Automation Deployment Using Infoblox IPAM Checklist

Task	Description	Details
Obtain, import, and configure the third-party IPAM solution provider plug-in or package.	Obtain and import the vRealize Orchestrator plug-in, run the vRealize Orchestrator configuration workflows, and register the IPAM provider endpoint type in vRealize Orchestrator. If the VMware Solution Exchange at https://marketplace.vmware.com/vsx does not contain the IPAM provider package that you need, you can create your own using an IPAM Solution Provider SDK and supporting documentation. See the vRealize Automation Example Third-Party IPAM Package page at code.vmware.com/web/sdk.	See Checklist For Providing Third-Party IPAM Provider Support.
Create an third-party IPAM solution provider endpoint.	Create a new IPAM endpoint in vRealize Automation.	See Create a Third-Party IPAM Provider Endpoint.
Specify third-party IPAM solution provider endpoint settings in an external network profile.	Create an external network profile and specify the defined IPAM endpoint in vRealize Automation.	See Create an External Network Profile by Using A Third-Party IPAM Provider.
Optionally specify third-party IPAM solution provider endpoint settings in a routed network profile.	Create an on-demand network profile and specify the defined IPAM endpoint in vRealize Automation.	See Create a Routed Network Profile By Using a Third-Party IPAM Endpoint or Create a NAT Network Profile By Using a Third-Party IPAM Endpoint in vRealize Automation.
Define a reservation to use the network profile.	Create a reservation that calls the network profile in vRealize Automation.	See Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer.

Table 4-19. Preparing for Provisioning a vRealize Automation Deployment Using Infoblox IPAM Checklist (continued)

Task	Description	Details
Define a blueprint that uses the network profile.	Create a blueprint that uses the reservation in vRealize Automation.	See Chapter 5 Providing Service Blueprints to Users.
Publish the blueprint to the catalog to make it available for use.	Publish the blueprint to the catalog in vRealize Automation. Add any required entitlements.	See Publish a Blueprint.
Request machine provisioning by using the blueprint catalog item.	Use the blueprint catalog item to request machine provisioning in vRealize Automation.	See Managing the Service Catalog.

Configuring XaaS Resources

By configuring XaaS endpoints you can connect the vRealize Automation to your environment. When you configure vRealize Orchestrator plug-ins as endpoints, you use the vRealize Automation user interface to configure the plug-ins instead of using the vRealize Orchestrator configuration interface.

To use vRealize Orchestrator capabilities and the vRealize Orchestrator plug-ins to expose VMware and third-party technologies to vRealize Automation, you can configure the vRealize Orchestrator plug-ins by adding the plug-ins as endpoints. This way, you create connections to different hosts and servers, such as vCenter Server instances, a Microsoft Active Directory host, and so on.

When you add a vRealize Orchestrator plug-in as an endpoint by using the vRealize Automation UI, you run a configuration workflow in the default vRealize Orchestrator server. The configuration workflows are located in the **vRealize Automation > XaaS > Endpoint Configuration** workflows folder.

Important Configuring a single plug-in in vRealize Orchestrator and in the vRealize Automation console is not supported and results in errors.

Configure the Active Directory Plug-In as an Endpoint

You add an endpoint and configure the Active Directory plug-in to connect to a running Active Directory instance and manage users and user groups, Active Directory computers, organizational units, and so on.

After you add an Active Directory endpoint, you can update it at any time.

Prerequisites

- Verify that you have access to a Microsoft Active Directory instance. See the Microsoft Active Directory documentation.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > vRO Configuration > Endpoints**.

2 Click the **New** icon (✚).

3 In the **Plug-in** drop-down menu, select **Active Directory**.

4 Click **Next**.

5 Enter a name and, optionally, a description.

6 Click **Next**.

7 Configure the Active Directory server details.

a Enter the IP address or the DNS name of the host on which Active Directory runs in the **Active Directory host IP/URL** text box.

b Enter the lookup port of your Active Directory server in the **Port** text box.

vRealize Orchestrator supports the Active Directory hierarchical domains structure. If your domain controller is configured to use Global Catalog, you must use port 3268. You cannot use the default port 389 to connect to the Global Catalog server. In addition to ports 389 and 3268, you can use 636 for LDAPS.

c Enter the root element of the Active Directory service in the **Root** text box.

For example, if your domain name is *mycompany.com*, then your root Active Directory is `dc=mycompany,dc=com`.

This node is used for browsing your service directory after entering the appropriate credentials. For large service directories, specifying a node in the tree narrows the search and improves performance. For example, rather than searching in the entire directory, you can specify `ou=employees,dc=mycompany,dc=com`. This root element displays all the users in the Employees group.

d (Optional) To activate encrypted certification for the connection between vRealize Orchestrator and Active Directory, select **Yes** from the **Use SSL** drop-down menu.

The SSL certificate is automatically imported without prompting for confirmation even if the certificate is self-signed.

e (Optional) Enter the domain in the **Default Domain** text box.

For example, if your domain name is *mycompany.com*, type **`@mycompany.com`**.

8 Configure the shared session settings.

The credentials are used by vRealize Orchestrator to run all the Active Directory workflows and actions.

a Enter the user name for the shared session in the **User name for the shared session** text box.

a Enter the password for the shared session in the **Password for the shared session** text box.

9 Click **Finish**.

Results

You added an Active Directory instance as an endpoint. XaaS architects can use XaaS to publish Active Directory plug-in workflows as catalog items and resource actions.

What to do next

- To use vRealize Automation blueprints to manage your Active Directory users in your environment, create an XaaS blueprint based on Active Directory. For an example, see Create an XaaS Blueprint and Action for Creating and Modifying a User .
- To use vRealize Automation to create Active Directory records when a machine is deployed, you can create different Active Directory policies and apply them to different business groups and blueprints. See Create and Apply Active Directory Policies.

Configure the HTTP-REST Plug-In as an Endpoint

You can add an endpoint and configure the HTTP-REST plug-in to connect to a REST host.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.
- Verify that you have access to a REST host.

Procedure

1 Select **Administration > vRO Configuration > Endpoints**.

2 Click the **New** icon (✚).

3 Select **HTTP-REST** from the **Plug-in** drop-down menu.

4 Click **Next**.

5 Enter a name and, optionally, a description.

6 Click **Next**.

7 Provide information about the REST host.

   a Enter the name of the host in the **Name** text box.

   b Enter the address of the host in the **URL** text box.

   Note If you use Kerberos access authentication, you must provide the host address in FDQN format.

c (Optional) Enter the number of seconds before a connection times out in the **Connection timeout (seconds)** text box.

The default value is 30 seconds.

d (Optional) Enter the number of seconds before an operation times out in the **Operation timeout (seconds)** text box.

The default value is 60 seconds.

8 (Optional) Configure proxy settings.

a Select **Yes** to use a proxy from the **Use Proxy** drop-down menu.

b Enter the IP of the proxy server in the **Proxy address** text box.

c Enter the port number to communicate with the proxy server in the **Proxy port** text box.

9 Click **Next**.

10 Select the authentication type.

Option	Action
None	No authentication is required.
OAuth 1.0	Uses OAuth 1.0 protocol. You must provide the required authentication parameters under OAuth 1.0. a Enter the key used to identify the consumer as a service provider in the **Consumer key** text box. b Enter the secret to establish ownership of the consumer key in the **Consumer secret** text box. c (Optional) Enter the access token that the consumer uses to gain access to the protected resources in the **Access token** text box. d (Optional) Enter the secret that the consumer uses to establish ownership of a token in the **Access token secret** text box.
OAuth 2.0	Uses OAuth 2.0 protocol. Enter the authentication token in the **Token** text box.
Basic	Provides basic access authentication. The communication with the host is in shared session mode. a Enter the user name for the shared session in the **Authentication user name** text box. b Enter the password for the shared session in the **Authentication password** text box.
Digest	Provides digest access authentication that uses encryption. The communication with the host is in shared session mode. a Enter the user name for the shared session in the **Authentication user name** text box. b Enter the password for the shared session in the **Authentication password** text box.

Option	Action
NTLM	Provides NT LAN Manager (NTLM) access authentication within the Window Security Support Provider (SSP) framework. The communication with the host is in shared session mode. a Provide the user credentials for the shared session. ■ Enter the user name for the shared session in the **Authentication user name** text box. ■ Enter the password for the shared session in the **Authentication password** text box. b Configure the NTLM details ■ (Optional) Enter the workstation name in the **Workstation for NTLM authentication** text box. ■ Enter the domain name in the **Domain for NTLM authentication** text box.
Kerberos	Provides Kerberos access authentication. The communication with the host is in shared session mode. a Enter the user name for the shared session in the **Authentication user name** text box. b Enter the password for the shared session in the **Authentication password** text box.

11 Click **Finish**.

Results

You configured the endpoint and added a REST host. XaaS architects can use XaaS to publish HTTP-REST plug-in workflows as catalog items and resource actions.

Configure the PowerShell Plug-In as an Endpoint

You can add an endpoint and configure the PowerShell plug-in to connect to a running PowerShell host, so that you can call PowerShell scripts and cmdlets from vRealize Orchestrator actions and workflows.

Prerequisites

- Verify that you have access to a Windows PowerShell host. For more information about Microsoft Windows PowerShell, see the Windows PowerShell documentation.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > vRO Configuration > Endpoints**.

2 Click the **New** icon (+).

3 Select **PowerShell** from the **Plug-in** drop-down menu.

4 Click **Next**.

5 Enter a name for the PowerShell endpoint.

6 (Optional) Enter a description for the PowerShell endpoint.

7 Click **Next**.

8 Specify the PowerShell host details.

   a Enter the name of the host in the **Name** text box.

   b Enter the IP address or the FDQN of the host in the **Host/IP** text box.

9 Configure the WinRM settings for the PowerShell host.

   a Enter the port number to use for communication with the host in the **Port** text box under the PowerShell host details.

   b Select a transport protocol from the **Transport protocol** drop-down menu.

   Note If you use the HTTPS transport protocol, the certificate of the remote PowerShell host is imported to the vRealize Orchestrator keystore.

   c Select the authentication type from the **Authentication** drop-down menu.

   Note To use Kerberos authentication, enable it on the WinRM service. For information about configuring Kerberos authentication, see *Using the PowerShell Plug-In*.

10 Enter the credentials for a shared session communication with the PowerShell host in the **User name** and **Password** text boxes.

11 Click **Finish**.

Results

You added a Windows PowerShell host as an endpoint. XaaS architects can use the XaaS to publish PowerShell plug-in workflows as catalog items and resource actions.

Configure the SOAP Plug-In as an Endpoint

You can add an endpoint and configure the SOAP plug-in to define a SOAP service as an inventory object, and perform SOAP operations on the defined objects.

Prerequisites

- Verify that you have access to a SOAP host. The plug-in supports SOAP Version 1.1 and 1.2, and WSDL 1.1 and 2.0.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > vRO Configuration > Endpoints**.

2 Click the **New** icon (✚).

3 From the **Plug-in** drop-down menu, select **SOAP**.

4 Click **Next**.

5 Enter a name and, optionally, a description.

6 Click **Next**.

7 Provide the details about the SOAP host.

   a Enter the name of the host in the **Name** text box.

   b Select whether to provide the WSDL content as text from the **Provide WSDL content** drop-down menu.

Option	Action
Yes	Enter the WSDL text in the **WSDL content** text box.
No	Enter the correct path in the **WSDL URL** text box.

   c (Optional) Enter the number of seconds before a connection times out in the **Connection timeout (in seconds)** text box.

   The default value is 30 seconds.

   d (Optional) Enter the number of seconds before an operation times out in the **Request timeout (in seconds)** text box.

   The default value is 60 seconds.

8 (Optional) Specify the proxy settings.

   a To use a proxy, select **Yes** from the **Proxy** drop-down menu.

   b Enter the IP of the proxy server in the **Address** text box.

   c Enter the port number to communicate with the proxy server in the **Port** text box.

9 Click **Next**.

10 Select the authentication type.

Option	Action
None	No authentication is required.
Basic	Provides basic access authentication. The communication with the host is in shared session mode. a Enter the user name for the shared session in the **User name** text box. b Enter the password for the shared session in the **Password** text box.
Digest	Provides digest access authentication that uses encryption. The communication with the host is in shared session mode. a Enter the user name for the shared session in the **User name** text box. b Enter the password for the shared session in the **Password** text box.

Option	Action
NTLM	Provides NT LAN Manager (NTLM) access authentication in the Window Security Support Provider (SSP) framework. The communication with the host is in shared session mode. a Provide the user credentials. ■ Enter the user name for the shared session in the **User name** text box. ■ Enter the password for the shared session in the **Password** text box. b Provide the NTLM settings. ■ Enter the domain name in the **NTLM domain** text box. ■ (Optional) Enter the workstation name in the **NTLM workstation** text box.
Negotiate	Provides Kerberos access authentication. The communication with the host is in shared session mode. a Provide the user credentials. 1 Enter the user name for the shared session in the **User name** text box. 2 Enter the password for the shared session in the **Password** text box. b Enter the Kerberos service SPN in the **Kerberos service SPN** text box.

11 Click **Finish**.

Results

You added a SOAP service. XaaS architects can use XaaS to publish SOAP plug-in workflows as catalog items and resource actions.

Configure the vCenter Server Plug-In as an Endpoint

You can add an endpoint and configure the vCenter Server plug-in to connect to a running vCenter Server instance to create XaaS blueprints to manage vSphere inventory objects.

Prerequisites

- Install and configure vCenter Server. See *vSphere Installation and Setup*.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > vRO Configuration > Endpoints**.

2 Click the **New** icon (✚).

3 Select **vCenter Server** from the **Plug-in** drop-down menu.

4 Click **Next**.

5 Enter a name and, optionally, a description.

6 Click **Next**.

7 Provide information about the vCenter Server instance.

   a Enter the IP address or the DNS name of the machine in the **IP or host name of the vCenter Server instance to add** text box.

      This is the IP address or DNS name of the machine on which the vCenter Server instance you want to add is installed.

   b Enter the port to communicate with the vCenter Server instance in the **Port of the vCenter Server instance** text box.

      The default port is 443.

   c Enter the location of the SDK to use for connecting to your vCenter Server instance in the **Location of the SDK that you use to connect to the vCenter Server instance** text box.

      For example, `/sdk`.

8 Click **Next**.

9 Define the connection parameters.

   a Enter the HTTP port of the vCenter Server instance in the **HTTP port of the vCenter Server instance - applicable for VC plugin version 5.5.2 or earlier** text box.

   b Enter the credentials for vRealize Orchestrator to use to establish the connection to the vCenter Server instance in the **User name of the user that Orchestrator will use to connect to the vCenter Server instance** and **Password of the user that Orchestrator will use to connect to the vCenter Server instance** text boxes.

      The user that you select must be a valid user with privileges to manage vCenter Server extensions and a set of custom defined privileges.

10 Click **Finish**.

Results

You added a vCenter Server instance as an endpoint. XaaS architects can use the XaaS to publish vCenter Server plug-in workflows as catalog items and resource actions.

Create a Microsoft Azure Endpoint

You can create a Microsoft Azure endpoint to facilitate a credentialed connection between vRealize Automation and an Azure deployment.

An endpoint establishes a connection to a resource, in this case an Azure instance, that you can use to create virtual machine blueprints. You must have an Azure endpoint to use as the basis of blueprints for provisioning Azure virtual machines. If you use multiple Azure subscriptions, you need endpoints for each subscription ID.

As an alternative, you can create an Azure connection directly from vRealize Orchestrator using the Add an Azure Connection command located under **Library > Azure > Configuration** in the vRealize Orchestrator workflow tree. For most scenarios, creating a connection through the endpoint configuration as described herein is the preferred option.

Azure endpoints are supported by vRealize Orchestrator and XaaS functionality. You can create, delete, or edit an Azure endpoint. If you change an existing endpoint and do not run any updates on the Azure portal through the updated connection for several hours, problems may occur. You must restart the vRealize Orchestrator service using the `service vco-service restart` command. Failure to restart the service may result in errors.

Prerequisites

- Configure a Microsoft Azure instance and obtain a valid Microsoft Azure subscription from which you can use the subscription ID. See Microsoft Azure Endpoint Configuration for more information about configuring Azure and obtaining a subscription ID.
- Verify that your vRealize Automation deployment has at least one tenant and one business group.
- Create an Active Directory application as described in https://azure.microsoft.com/en-us/documentation/articles/resource-group-create-service-principal-portal.
- Make note of the following Azure related information, as you will need it during endpoint and blueprint configuration.
  - subscription ID
  - tenant ID
  - storage account name
  - resource group name
  - location
  - virtual network name
  - client application ID
  - client application secret key
  - virtual machine image URN
- The vRealize Automation Azure implementation supports a subset of the Microsoft Azure supported regions. See Azure Supported Regions.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1. Select **Administration > vRO Configuration > Endpoints**.
2. Click the **New** icon (✚).
3. On the Plug-in tab, click the **Plug-in** drop-down menu and select **Azure**.
4. Click **Next**.
5. Enter a name and, optionally, a description.
6. Click **Next**.

7 Populate the text boxes on the Details tab as appropriate for the endpoint.

Parameter	Description
Connection settings	
Connection name	Unique name for the new endpoint connection. This name appears in the vRealize Orchestrator interface to help you identify a particular connection.
Azure subscription id	The identifier for your Azure subscription. The ID defines the storage accounts, virtual machines and other Azure resources to which you have access.
Azure Environment	The geographic region for the deployed Azure resource. vRealize Automation supports all current Azure regions based on the subscription ID.
Resource manager settings	
Azure service URI	The URI through which you gain access to your Azure instance. The default value of `https://management.azure.com/` is appropriate for many typical implementations. This box is auto-populated when you select an environment.
Tenant Id	The Azure tenant ID that you want the endpoint to use.
Client Id	The Azure client identifier that you want the endpoint to use. This is assigned when you create an Active Directory application.
Client secret	The key used with an Azure client ID. This key is assigned when you create an Active Directory application.
Azure storage URI	The URI through which you gain access to your Azure storage instance. This box is auto-populated when you select an environment.
Proxy Settings	
Proxy host	If your company uses a proxy Web server, enter the host name of that server.
Proxy port	If your company uses a proxy Web server, enter the port number of that server.

8 (Optional) Click Properties and add supplied custom properties, property groups, or your own custom property definitions.

9 Click **Finish**.

What to do next

Create appropriate resource groups, storage accounts, and network security groups in Azure. You should also create load balancers if appropriate for your implementation.

Action	Options
Create an Azure resource group	- Create the resource group using the Azure portal. See the Azure documentation for specific instructions. - Use the appropriate vRealize Orchestrator workflow found under the `Library/Azure/Resource/Create resource group`. - In vRealize Automation, create and publish an XaaS blueprint that contains the vRealize Orchestrator workflow. You can request the resource group after attaching it to the service and entitlements. **Note** The Resource Group resource type is not supported or managed by vRealize Automation.
Create an Azure storage account	- Use Azure to create a storage account. See the Azure documentation for specific instructions. - Use the appropriate vRealize Orchestrator workflow found under `Library/Azure/Storage/Create storage account`. - In vRealize Automation, create and publish an XaaS blueprint that contains the vRealize Orchestrator workflow. You can request the storage account after attaching it to the service and entitlements.
Create an Azure network security group	- Use Azure to create a security group. See the Azure documentation for specific instructions. - Use the appropriate vRealize Orchestrator workflow found under the `Library/Azure/Network/Create Network security group` . - In vRealize Automation, create and publish an XaaS blueprint that contains the vRealize Orchestrator workflow. You can request the security group after attaching it to the service and entitlements.

Azure Supported Regions

The vRealize Automation Azure implementation supports a subset of the Microsoft Azure supported regions.

The following Azure regions are supported by the Azure implementation within vRealize Automation.

- East Asia
- Southeast Asia
- Central US
- East US
- East US 2
- West US
- West US 2
- North Central US
- South Central US
- North Europe
- West Europe
- Japan West
- Japan East
- Brazil South
- Australia East
- Australia Southeast
- South India
- Central India
- West India
- Canada Central
- Canada East
- West Central US
- Korea Central
- Korea South
- UK West
- UK South
- China East
- China North

Creating and Configuring Containers

You can use the Containers tab in vRealize Automation to open the Containers for vRealize Automation integrated application and create and configure the containers and container network settings to make available to vRealize Automation blueprint architects.

You can define containers by using new and existing templates and images in the integrated Containers application. You can then add container components, and their associated network settings, to vRealize Automation blueprints.

Managing Container Hosts and Clusters

You can view and manage the hosts that you add from the Clusters page. In the context of Containers, the host is a virtual machine or infrastructure that lets you run containers.

The Clusters page, under the Infrastructure tab, contains the controls for adding new clusters and hosts. To add a host in your Containers environment, you must add it to a cluster. You can monitor the state of the provision requests of existing hosts and view event logs for your containers from any page in the Library and Deployments tabs. The Requests and Event Log panels are located on the right side of the pages.

Create a Container Host Cluster

You must add a host to a cluster to deploy containers.

Prerequisites

Select a business group from the top-left side of the Containers tab.

Procedure

1. Log in to the vRealize Automation console as a **container administrator**.
2. Click the **Containers** tab.
3. Click **Infrastructure > Container Host Clusters**.

4 Click **Cluster**.

5 Enter a cluster name and description.

6 Select either Docker virtual container host (VCH) in the **Type** drop-down menu.

7 Enter your host IP address or host name using the `http(s)://<hostname>:<port>` URL format.

8 Select your login credentials from the list.

   Containers supports credentials authentication and public-private key authentication. You can add your credentials from the **Identity Management** page.

9 Click **Save**.

Results

You successfully created a container host cluster.

Using Container Deployment Policies

You can link deployment policies to hosts and container definitions. You use deployment policies in Containers for vRealize Automation to set a preference for the specific host and quotas for when you deploy a container.

Deployment policies that are applied to a container have a higher priority than placements that are applied to container hosts.

Note Deployment policies are deprecated and will be removed in a future release of vRealize Automation.

Set a Deployment Policy on a Host

Set a preference for the specific host and quotas for when you deploy a container.

Note Deployment policies are deprecated and will be removed in a future release of vRealize Automation.

Prerequisites

Add a host to a cluster.

Procedure

1 Log in to the vRealize Automation console as a **container administrator**.

2 Click the **Containers** tab.

3 Select **Infrastructure > Container Host Clusters**.

4 Click on the cluster that contains the host you want to edit.

5 Click **Resources**.

6 Click the options icon on the host you want to configure and click **Edit**.

7 Select the deployment policy and click **Update**.

Set a Deployment Policy for a Container Definition

Set a deployment policy for a container definition.

Note Deployment policies are deprecated and will be removed in a future release of vRealize Automation.

Procedure

1 Click the **Containers** tab.

2 Click **Container Hot Clusters** to start provisioning the container.

3 Select an existing container from the list.

4 In the provisioning options, click **Policy**.

5 From the **Deployment Policy** drop-down list, select an existing policy.

6 Provision the container or save it as a template.

Configuring Container Settings

You can define a single container or a multi-container application by using new and existing container configuration properties and settings.

In addition to the core Containers for vRealize Automation settings, the following vRealize Automation settings are available for deployments that use container components:

- Health configuration
- Links
- Exposed services
- Cluster size and scale in-and scale out parameters

Configure Health Checks in Containers

You can configure a health check method to update the status of a container based on custom criteria.

You can use HTTP or TCP protocols when running a command on the container. You can also specify a health check method.

Prerequisites

- Verify that Containers for vRealize Automation is enabled in your supported vRealize Automation deployment.
- Verify that you have **container administrator** or **container architect** role privileges.

Procedure

1 Log in to vRealize Automation.

2 Click the **Containers** tab.

3 Select **Library > Templates** in the left pane.

4 Edit the template or image.

Option	Description
To edit a template	a Click **Edit** in the upper-right section of the template that you want to open. b Click **Edit** in the upper-right section of the container that you want to open.
To edit an image.	Click the arrow next to the image's **Provision** button, and click **Enter additional info**.

5 Click the **Health Config** tab.

6 Select a health mode.

Table 4-20. Health Configuration Modes

Mode	Description
None	Default. No health checks are configured.
HTTP	If you select **HTTP**, you must provide an API to access and an HTTP method and version to use. The API is relative and you do not need to enter the address of the container. You can also specify a timeout period for the operation and set health thresholds. For example, a healthy threshold of 2 means that two consecutive successful calls must occur for the container to be considered healthy and in the RUNNING status. An unhealthy threshold of 2 means that two unsuccessful calls must occur for the container to be considered unhealthy and in the ERROR status. For all the states in between the healthy and unhealthy thresholds, the container status is DEGRADED.
TCP connection	If you select **TCP connection**, you must only enter a port for the container. The health check attempts to establish a TCP connection with the container on the provided port. You can also specify a timeout value for the operation and set healthy or unhealthy thresholds as with HTTP.
Command	If you select **Command**, you must enter a command to be run on the container. The success of the health check is determined by the exit status of the command.
Ignore health check on provision	Uncheck this option to force health check on provision. By forcing it, a container is not considered provisioned until one successful health check passes.
Autodeploy	Automatic redeployment of containers when they are in ERROR state.

7 Click **Save**.

Configure Links in Containers

Links and exposed services address communication across container services and load balancing across hosts. You can configure link settings for your containers in Containers.

You can use links to enable communication between multiple services in your application. Links in Containers are similar to Docker links, but connect containers across hosts. A link consists of two parts: a service name and an alias. The service name is the name of the service or template being called. The alias is the host name that you use to communicate with that service.

For example, if you have an application that contains a Web and database service and you define a link in the Web service to the database service by using an alias of **`my-db`**, the Web service application opens a TCP connection to `my-db:{`*`PORT_OF_DB`*`}`. The *`PORT_OF_DB`* is the port that the database listens to, regardless of the public port that is assigned to the host by the container settings. If MySQL is checking for updates on its default port of 3306, and the published port for the container host is 32799, the Web application accesses the database at *`my-db`*`:3306`.

Note It is recommended that you use networks instead of links. Links are now a legacy Docker feature with significant limitations when linking container clusters, including:

- Docker does not support multiple links with the same alias. It is recommended that you allow Containers for vRealize Automation to generate link aliases for you.
- You cannot update the links of a container runtime. When scaling up or down a linked cluster, the dependent container's links will not be updated.

Prerequisites

- Verify that Containers for vRealize Automation is enabled in your supported vRealize Automation deployment.
- Verify that you have **container administrator** or **container architect** role privileges.
- Verify that a bridge network is available for linking services.
- Verify that the internal port of the target service is published. For cross communication, the service can be mapped to any other port but must be accessible from outside the host.
- Verify that the service hosts are able to access each other.

Procedure

1 Log in to vRealize Automation.

2 Click the Containers tab.

3 Select **Library > Templates** in the left pane.

4 Edit the template or image.

Option	Description
To edit a template	a Click **Edit** in the upper-right section of the template that you want to open. b Click **Edit** in the upper-right section of the container that you want to open.
To edit an image.	Click the arrow next to the image's **Provision** button, and click **Enter additional info**.

5 Click the **Basic** tab.

6 In the **Services** text box, enter a comma-separated list of services that the container is dependant on.

7 In the **Alias** text box, enter a descriptive name for the service or comma-separated list of services.

8 Click **Save**.

Configure Exposed Services in Containers

You can use a unique host name for a load balancer by providing an address and a placeholder in your container settings.

The placeholder determines the location of an automatically generated part of the URL. This value is unique for each host name. The address supports the %s format character to specify where the placeholder is located.

Note If the placeholder is not used, it is positioned as a prefix or suffix of the host name, depending on the system configuration.

It is recommended you use a load balancer that can target requests to each node if you build an application which includes a service that must be publicly exposed and which must also scale in and out. After you provision the application, the load balancer configuration is updated whenever the service is scaled in or out by vRealize Automation.

Prerequisites

- Verify that Containers for vRealize Automation is enabled in your supported vRealize Automation deployment.
- Verify that you have **container administrator** or **container architect** role privileges.

Procedure

1 Log in to vRealize Automation.

2 Click the **Containers** tab.

3 Select **Library > Templates** in the left pane.

4 Edit the template or image.

Option	Description
To edit a template	a Click **Edit** in the upper-right section of the template that you want to open. b Click **Edit** in the upper-right section of the container that you want to open.
To edit an image.	Click the arrow next to the image's **Provision** button, and click **Enter additional info**.

5 Click the **Network** tab.

6 In the **Address** text box, enter the location of the placeholder.

The address host acts as a virtual host. To access the address host, you can add mapping information in the `etc/hosts` file or use a DNS that maps the container address to the host name.

7 In the **Container Port** text box, enter the port number used to expose the service.

Use the sample format provided in the form. If your container application exposes more than one port, specify which internal port or ports can expose the service.

8 Click **Save**.

Configure Cluster Size and Scale in Containers

You can create container clusters by using Containers placement settings to specify cluster size.

When you configure a cluster, the Containers provisions the specified number of containers. Requests are load balanced among all containers in the cluster.

You can modify the cluster size on a provisioned container or application to increase or decrease the size of the cluster by one. When you modify the cluster size at runtime, all affinity filters and placement rules are considered.

Prerequisites

- Verify that Containers for vRealize Automation is enabled in your supported vRealize Automation deployment.
- Verify that you have **container administrator** or **container architect** role privileges.

Procedure

1 Log in to vRealize Automation.

2 Click the **Containers** tab.

3 Select **Library > Templates** in the left pane.

4 Edit the template or image.

Option	Description
To edit a template	a Click **Edit** in the upper-right section of the template that you want to open. b Click **Edit** in the upper-right section of the container that you want to open.
To edit an image.	Click the arrow next to the image's **Provision** button, and click **Enter additional info**.

5 Click the **Policy** tab.

6 Set the container cluster size.

7 Click **Save**.

Configuring and Using Templates and Images in Containers

Containers uses templates to provision containers.

A template is a reusable configuration for provisioning a container or a suite of containers. In a template, you can define a multi-tier application that consists of linked services.

A service is defined as one or more containers of the same type or image.

You can create a custom container template based on an existing template on the **Templates** page or import a properly formatted YAML file. You can also provision a container template or image.

Create a Custom Container Template

You can create a custom template and use it to define a container.

A template is a reusable configuration that you can use for provisioning a container or a suite of containers.

The Templates page displays template images that are available to you based on registries that you define. You can create a custom template, based on an existing template image or import a template or Docker Compose file. See Import a Container Template or Docker Compose File.

You can also create a custom template or image by using the **Provision > Enter additional info** option described in Provision a Container from a Template or Image.

Prerequisites

- Verify that you have **container administrator** role privileges.

Procedure

1 Log in to the vRealize Automation console as a **container administrator**.

2 Click the **Containers** tab.

3 Select **Library > Templates** in the left pane.

   A list displays the templates and images that are available for provisioning.

   - Configured templates in the Images view.
   - Existing or custom templates in the **Template** view.
   - All available templates and images based on your specified registries in the **All** view.

   The **Import** and **Export** options are also available to import or export templates and images.

4 Click the arrow next to the **Provision** button of an image you want to include in the template.

5 Click **Enter additional info**.

6 Click **Save as Template** to save your changes as a new container template in Containers for vRealize Automation.

What to do next

You can edit a template for future provisioning. Existing applications that were provisioned from the template are not affected by changes that you make to the template after provisioning.

Import a Container Template or Docker Compose File

You can use an imported Docker Container template or a Docker Compose YAML file as a custom template in the Containers for vRealize Automation.

If using a YAML file, enter the content of the YAML file as text or browse to and upload the YAML file. The YAML file represents the template, the configuration for the different containers, and their connections. The supported format types are Docker Compose YAML and Containers for vRealize Automation YAML.

Containers for vRealize Automation YAML is similar to Docker Compose, but it uses the vRealize Automation blueprint YAML format visible in the vRealize Automation REST API or in vRealize CloudClient. The Containers for vRealize Automation YAML allows you to import existing Docker Compose applications and modify, provision, and manage them by using Containers.

Prerequisites

- Verify that Containers for vRealize Automation is enabled in your supported vRealize Automation deployment.
- Log in to vRealize Automation as a **container administrator**.

For information about the YAML format used by vRealize Automation service REST APIs, see *vRealize Automation API Reference*.

Procedure

1 Click the **Containers** tab.

2 Select **Library > Templates** in the left pane.

A list displays the templates and images that are available for provisioning.

- Configured templates in the Images view.
- Existing or custom templates in the **Template** view.
- All available templates and images based on your specified registries in the **All** view.

The **Import** and **Export** options are also available to import or export templates and images.

3 Click the **Import template or Docker Compose** icon.

The Import Template page appears.

4 Provide the YAML file content.

Option	Description
Load from File	Click **Load from File** to browse to and select the YAML file from a directory.
Enter template or Docker Compose	Paste the content of a properly formatted YAML file in the **Enter template or Docker Compose** text box.

5 Click **Import**.

The new template appears in the **Templates** view.

Provision a Container from a Template or Image

You can provision a container from a template or image in your Templates view.

The provisioning process creates a container based on the configuration settings that exist in the template or image from which you provision.

You can provision a container from a template or image either by using existing configuration settings or by editing configuration settings and then provisioning.

You can also edit and save configuration settings to create a new, customized container template or image.

Prerequisites

- Verify that Containers for vRealize Automation is enabled in your supported vRealize Automation deployment.
- Log in to vRealize Automation as a **container administrator**.

Procedure

1 Click the **Containers** tab.

2 Select **Library > Templates** in the left pane.

A list displays the templates and images that are available for provisioning.

- Configured templates in the Images view.

- Existing or custom templates in the **Template** view.
- All available templates and images based on your specified registries in the **All** view.

The **Import** and **Export** options are also available to import or export templates and images.

3 Use the **All**, **Images**, or **Templates** view options to view the image or template to provision.

4 Provision the template or image.

Option	Description
Provision using existing settings.	a Click **Provision**. The Provision Requests view displays information about provisioning success.
Provision by editing settings.	a Click the arrow next to the **Provision** button. b Click **Enter additional info**. c Enter the additional information for the container in the **Provision a Container form**. d When you have completed the form updates, click **Provision** to provision using the modified settings. e Click **Save as Template** to save your changes as a new container template in Containers for vRealize Automation. The Provision Requests view displays information about provisioning success.

Export a Container Template or Docker Compose File

You can export a container template as a Docker Compose YAML file or a Containers for vRealize Automation YAML file.

You can import a template, modify it programmatically by using the vRealize Automation REST API or vRealize CloudClient, or graphically in Containers. You can then export the modified file. For example, you can import in Docker Compose format and export in the blueprint YAML format used in the vRealize Automation composition-service API. However, some configurations that are specific to Containers, such as health configuration and affinity constraints are not included if you export the template in Docker Compose format.

Prerequisites

- Verify that Containers for vRealize Automation is enabled in your supported vRealize Automation deployment.
- Log in to vRealize Automation as a **container administrator**.

For information about the YAML format used by vRealize Automation service REST APIs, see *vRealize Automation API Reference*.

Procedure

1 Click the **Containers** tab.

2 Select **Library > Templates** in the left pane.

   A list displays the templates and images that are available for provisioning.

   - Configured templates in the Images view.
   - Existing or custom templates in the **Template** view.
   - All available templates and images based on your specified registries in the **All** view.

   The **Import** and **Export** options are also available to import or export templates and images.

3 Point to a template and click its **Export** icon.

4 When prompted, select an output format type:

   - **YAML Blueprint**

     This format adheres to the blueprint YAML format used in the vRealize Automation composition-service API.

   - **Docker Compose**

     This format adheres to the YAML format used in the Docker Compose application.

5 Click **Export**.

6 Save the file or open it with an appropriate application when prompted.

Using Container Registries

A Docker registry is a stateless, server-side application. You can use registries in Containers for vRealize Automation to store and distribute Docker images.

To configure a registry, you need to provide its address, a custom registry name, and optionally credentials. The address must start with HTTP or HTTPS to designate whether the registry is secured or unsecured. If the connection type is not provided, HTTPS is used by default.

Note For HTTP you must declare port 80; for HTTPS you must declare port 443. If no port is specified, the Docker engine expects port 5000, which can result in broken connections.

Note It is recommended you do not use HTTP registries because HTTP is considered insecure. If you want to use HTTP, you must modify the `DOCKER_OPTS` property on each host as follows:

`DOCKER_OPTS="--insecure-registry myregistrydomain.com:5000"`.

For more information, see the Docker documentation at https://docs.docker.com/registry/insecure/.

Containers can interact with both Docker Registry HTTP API V1 and V2 in the following manner:

V1 over HTTP (unsecured, plain HTTP registry)

You can freely search this kind of registry, but you must manually configure each Docker host with the `--insecure-registry` flag to provision containers based on images from insecure registries. You must restart the Docker daemon after setting the property.

V1 over HTTPS

Use behind a reverse proxy, such as NGINX. The standard implementation is available through open source at https://github.com/docker/docker-registry.

V2 over HTTPS

The standard implementation is open sourced at https://github.com/docker/distribution.

V2 over HTTPS with basic authentication

The standard implementation is open sourced at https://github.com/docker/distribution.

V2 over HTTPS with authentication through a central service

You can run a Docker registry in standalone mode, in which there are no authorization checks. Supported third-party registries are JFrog Artifactory and Harbor. Docker Hub is enabled by default for all tenants and is not present in the registry list, but it can be deactivated with a system property.

Note Docker does not normally interact with secure registries configured with certificates signed by unknown authority. The container service handles this case by automatically uploading untrusted certificates to all docker hosts and enabling the hosts to connect to these registries. If a certificate cannot be uploaded to a given host, the host is automatically deactivated.

Create and Manage Container Registries

You can configure multiple registries to gain access to both public and private images.

Registries are public or private stores, from which you upload or download images. You can deactivate, edit, or delete the registries that you created. The images shown in the **Templates** tab are based on the registries that you define.

When you create or manage registries, you can click the **Credentials** or **Certificate** buttons to add or manage credentials and certificates.

Prerequisites

- Log in to vRealize Automation as a **container administrator**.
- Verify that at least one host is configured and available for container network configuration.

Procedure

1. Click the **Containers** tab.
2. Select **Library > Global Registries** .
3. Click **Registry** to create a new registry.

4 Enter the registry address.

5 Enter a name for the registry.

6 Select your login credentials from the drop-down list.

7 (Optional) Click **Verify** to confirm that the configured parameters are valid.

8 Click **Save** to add the registry.

Add Image to Favorites

For quick access to your most commonly used or preferred images, you can add images as your favorites.

When an image is added as a favorite, it appears on the Repositories homepage without searching. Only container administrators can add and remove images from favorites, while all users can view the favorite images per repository. Images marked as favorite display a star next to their name.

Procedure

1 From the Repositories page, select the registry from the drop-down menu and search for the desired image.

2 Click the arrow next to **Provision** and select **Add Image to Favorites**.

   A notification appears signifying that the image was successfully added to favorites and a star is added next to the image's name.

Results

The image appears on the Repositories page without searching. To remove the image from favorites, on the Repositories page, click the arrow next to **Provision** and select **Remove Image From Favorites**.

Configuring Network Resources for Containers

You can create, modify, and attach network configurations to containers and container templates in the Containers for vRealize Automation application.

When you provision a container, the network configuration is embedded and available. You can customize the network settings for container components that you added to a vRealize Automation blueprint.

Create a New Network for Containers

If a suitable network configuration is not available, you can create a new one in vRealize Automation.

Prerequisites

- Verify that you have **container administrator**, **container architect**, or **IaaS administrator** role privileges.

- Verify that at least one host is configured and available for container network configuration.

Procedure

1. Log in to vRealize Automation.
2. Click the **Containers** tab.
3. Select **Deployments > Networks** in the left pane.

   The main panel displays the existing network configurations that can be provisioned as a part of the container deployment. The network configurations include both those collected from added Docker hosts and those created in vRealize Automation. The icons representing the network configurations display the network and IPAM drivers, subnet, gateway, and IP range information, the number of containers using the network configuration, and the number of hosts.
4. Click **+Network**.
5. Enter a name for the network.

   When you finish creating the new configuration, the name value will be appended with a unique identifier.
6. (Optional) To add more detailed configuration settings, select the **Advanced** check box.

   Additional network configuration settings appear in the Add Network panel.

7 Configure the advanced network configuration settings.

Option	Description
IPAM configuration	**Subnet** Provide subnet and gateway addresses that are unique to this network configuration. They must not overlap with any other networks on the same container host.

Custom properties	Optionally, specify custom properties for the new network configuration.
	`containers.ipam.driver`
	For use with containers only. Specifies the IPAM driver to be used when adding a Containers network component to a blueprint. The supported values depend on the drivers installed in the container host environment in which they are used. For example, a supported value might be `infoblox` or `calico` depending on the IPAM plug-ins that are installed on the container host.
	This property name and value are case-sensitive. The property value is not validated when you add it. If the specified driver does not exist on the container host at provisioning time, an error message is returned and provisioning fails.
	`containers.network.driver`
	For use with containers only. Specifies the network driver to be used when adding a Containers network component to a blueprint. The supported values depend on the drivers installed in the container host environment in which they are used. By default, Docker-supplied network drivers include bridge, overlay, and macvlan, while Virtual Container Host (VCH)-supplied network drivers include the bridge driver. Third-party network drivers such as `weave` and `calico` might also be available, depending on what network plug-ins are installed on the container host.
	This property name and value are case-sensitive. The property value is not validated when you add it. If the specified driver does not exist on the container host at provisioning time, an error message is returned and provisioning fails.

Note If you create the network without advanced settings, vRealize Automation supplies the settings automatically.

8 From the drop-down menu, select the host to which you want to connect the network.

9 Click **Create**.

Add a Network to a Container Template

You can add a network configuration to a container template to connect the containers to each other. This network configuration is automatically implemented for any applications that use the template. You can add either an existing network or configure and add a new network, as necessary.

Prerequisites

- Verify that you have a template available. If not, you must first create one.
- Verify that you have **container administrator**, **container architect**, or **IaaS administrator** role privileges.
- Verify that at least one host is configured and available for container network configuration.

Procedure

1. Log in to vRealize Automation.
2. Click the **Containers** tab.
3. Select **Library > Templates** in the left pane.

   An array of icons displays the templates and images that are available for provisioning.
4. (Optional) Modify the view to show only templates by clicking **View: Templates** in the upper right header above the icons.
5. Click **Edit** in the upper-right section of the template that you want to customize.

   The Edit Template page appears, displaying the container icons and a blank icon with a plus-sign.
6. Point to the blank icon.

   The **Add Network** icon appears.
7. Click the **Add Network** icon.

   The Add Network panel appears.
8. Add an existing network or create and add a new network.

Option	Description
Add an existing network.	a Click the **Existing** check box. b Click inside the **Name** field to display a list of existing networks. c Select the network you want to use and click **Save**.
Configure and add a new network.	a Enter a name for the network. b To add more detailed configuration settings, click the **Advanced** check box. c Click **Save**.

9. Connect the network to a container, by dragging the network connector icon from the container to any point on the horizontal icon representing the network.

Configuring Volumes for Containers

You can create, modify, and attach volumes to containers and container templates in the Containers for vRealize Automation application.

Containers for vRealize Automation uses Docker volumes for persistent data management. With volumes you can perform the following tasks:

- Share volumes between different containers within the same host.
- Update data instantly.
- Save the volume data after the container is deleted.

Create a New Volume for Containers

To extend your container storage, you must first create a data volume.

Prerequisites

- Verify that you have **container administrator**, **container architect**, or **IaaS administrator** role privileges.
- Verify that at least one host is configured and available for container volume configuration.

Procedure

1. Log in to vRealize Automation.
2. Click the **Containers** tab.
3. Select **Deployments > Volumes** in the left pane.

   The main panel displays the existing volume configurations that can be connected to the deployed containers. The volume configurations include both those collected from added Docker hosts and those created in vRealize Automation. The volume instances display the driver, scope, and driver options.
4. Click **+Volume**.
5. Enter a name for the volume.

   When you finish creating the configuration, the name value is appended with a unique identifier.
6. In the **Driver** text box, enter the driver of the volume plug-in you want to use. If you do not enter anything, local is used as the default value.
7. (Optional) To add more detailed configuration settings, click the **Advanced** check box.

   Additional configuration settings appear.
8. (Optional) Configure the advanced volume settings.

Option	Description
Driver Options	Specify the driver options you want to use. The options depend on the volume plug-in you are using.
Custom properties	Specify custom properties for the new configuration.

9. From the drop-down menu, select the host to which you want to connect the volume.

10 Click **Create**.

The Create Volume panel disappears and the added volume appears in the Volumes tab.

What to do next

Add a Volume to a Container Template

Add a Volume to a Container Template

Connect a volume to a container by adding it to a template.

Prerequisites

- Verify that you have a template available. If not, you must first create one.
- Verify that you have **container administrator**, **container architect**, or **IaaS administrator** role privileges.
- Verify that at least one host is configured and available for container volume configuration.

Procedure

1 Log in to vRealize Automation.

2 Click the **Containers** tab.

3 Select **Library > Templates** in the left pane.

An array of icons displays the templates and images that are available for provisioning.

4 (Optional) Modify the view to show only templates by clicking **View: Templates** in the upper right header above the icons.

5 Click **Edit** in the upper-right section of the template that you want to customize.

The Edit Template page appears, displaying the container icons, including a blank icon with a plus-sign.

6 Hover the cursor over the blank icon with the plus sign until the **Add Volume** icon appears.

7 Click the **Add Volume** icon.

8 Add an existing volume or create and add a new volume.

Option	Description
Add an existing volume.	a Click the **Existing** check box. b Click inside the **Name** field to display a list of existing volumes. c Select the volume you want to use and click **Save**.
Configure and add a new volume.	a Enter a name for the volume. b In the **Driver** text box, enter the driver of the volume plug-in you want to use. If you are not using an external storage system, enter `local`. c To add more detailed configuration settings, click the **Advanced** check box. d Click **Save**.

The Add Volume panel disappears and the added volume appears as a horizontal icon below the container icons in the Edit Template page. A volume icon also displays on the bottom border of the container icons.

9 Connect the volume to a container, by dragging the volume connector icon from the container to any point on the horizontal icon representing the volume.

10 (Optional) Click on the container path to change the location where the volume is mounted.

What to do next

Provision a Container from a Template or Image

Creating and Configuring PKS Containers

Pivotal Container Service (PKS) enables enterprises and service providers to simplify the deployment and operations of Kubernetes-based container services.

Using PKS containers offers the following key features:

- High availability
  - PKS has a built-in fault tolerance complete with routine health checks and self-correcting capabilities for Kubernetes clusters.
- Advanced networking and security
  - PKS is deeply integrated with NSX-T for advanced container networking including micro-segmentation, load balancing, and security policies.
- Streamlined operations
  - PKS provides cluster deployment and lifecycle management of Kubernetes.
- Multi-tenancy
  - PKS supports multi-tenancy for workload isolation and privacy within enterprise and cloud service.

Adding a PKS Endpoint

Before creating a PKS container, you have to add a PKS endpoint.

The first step to create a PKS container is to add a PKS endpoint. PKS endpoints allow you to link plans, existing Kubernetes clusters, and business groups.

Prerequisites

- Container administrator privilege
- PKS Credentials
- UAA address
- PKS endpoint address

Procedure

1 Navigate to credentials using the path **Identity Management > Credentials** to create and save your PKS credentials.

2 Select **PKS Endpoints > Create Endpoint**.

3 Enter the details of your PKS endpoint and test connection before saving.

  If the test fails, verify that the PKS credentials, UAA address, and PKS endpoint address are correct. You may have to ping the addresses to verify that they are active. Retry connection.

4 Click **Create** to save your PKS endpoint.

  Note If a Verify Certificate window appears you can select **Show Certificate** to view certificate details. Click **Yes** to proceed and save your endpoint.

Results

Your PKS endpoint is saved. After saving your PKS endpoint, you can click the endpoint to view the available Kubernetes clusters associated with it. If the cluster has not been registered within vRealize Automation, the Requested column will have the value **No**. To register it you need to add a cluster. If you want to edit your endpoint, click the PKS endpoint name and modify its details. You can remove the endpoint by selecting and clicking **Remove**.

Assigning PKS Endpoints to Business Groups

After creating a PKS endpoint, you can assign to specific business groups to grant access.

After creating a PKS endpoint, you can grant specific business groups access to it by assigning plans to it. This is used to restrict and limit the access of certain groups to certain functionality.

Note You can create plans separately in PKS . Plans cannot be added or modified in vRealize Automation.

Prerequisites

- Container administrator privilege
- Existing PKS endpoint

Procedure

1 Open your PKS endpoint and click **Plan Assignments**.

2 Select the desired group from the groups list and plan from the plans list.

  Note Using the + and - buttons, you can assign multiple plans to each business group and assign the same plan to multiple business groups.

3 Click **Save** to save your plan assignments.

Requesting a New PKS Cluster

If your desired cluster configuration does not exist, you can request a new cluster for an existing PKS endpoint.

As either a container developer or container administrator, you can request a new cluster for your PKS endpoint. Each PKS endpoint can contain multiple clusters. After a new cluster has been created, you can add it to your environment using **Add cluster** and provision it as desired.

Prerequisites

- An existing PKS endpoint
- Container developer or container administrator privilege

Procedure

1. Select **PKS Clusters > New Cluster**.
2. Select the PKS endpoint.

   After selecting a PKS endpoint, the plan is automatically populated according to the plans available for your business group.
3. Enter the details of the cluster.

   Note Although the number of worker nodes is defined by the plan, you can modify the number according to your needs.
4. Select how to connect to this cluster:
   - Master host name - Connects using the hostname of the cluster, assuming a DNS record exists.
   - Master Node IP - Connects using the IP address of the cluster.
5. Click **Create**.

Results

The new cluster is created and appears on the PKS Clusters homepage.

Adding a PKS Cluster

After a PKS endpoint is created, you can register the available associated clusters to vRealize Automation.

After creating a PKS endpoint, you can register the associated clusters by adding a cluster in vRealize Automation. Once the clusters are registered you can provision single images on them.

Prerequisites

- Container administrator privilege
- PSK endpoint with available clusters

Procedure

1. Verify that you are adding a cluster to the correct business group. The business group name is listed in the upper left pane. To switch between business groups, click **Group**.
2. Select **PKS Cluster > Add Cluster**.
3. Select the PKS endpoint to populate the available clusters.
4. Select how to connect to this cluster:
   - Master host name - Connects using the hostname of the cluster, assuming a DNS record exists.
   - Master Node IP - Connects using the IP address of the cluster.
5. Click **Add**.

Results

The cluster appears on the PKS Clusters page.

PKS Cluster Details

The details of a cluster provide information and tools to edit and interact with the cluster.

You can view and modify existing PKS clusters by clicking the clusters name from the **PKS Clusters** page. Also, the details of the cluster contain interactive tools you can use to interact with the cluster for more complex configurations.

Note You can only edit the number of worker nodes of a cluster.

Dashboard

The dashboard field status indicates that the Kubernetes dashboard is installed. If installed, you can access the dashboard by clicking **Installed** and logging in.

Note The dashboard must be configured on the cluster for basic authentication. Without basic authentication, you cannot log in.

Kubeconfig

The kubeconfig link is a downloadable configuration file for the cluster. You can use this configuration file, as container developer, to connect to and configure the Kubernetes cluster within the command-line prompt window. For example, using the `kubectl` command.

Provisioning Single Images on a Kubernetes Cluster

The containers functionality within vRealize Automation allows you to provision a single image on a PKS cluster.

After a PKS cluster has been added, you can provision a single image on it as a combination of a Kubernetes pod and deployment.

Prerequisites

- Container developer privilege
- PKS cluster

Procedure

1. Navigate to **Library > Repositories**.
2. Select desired registry from the drop-down menu.
3. Search for an existing image within that registry using the repositories text box.
4. Click **Provision** on the desired image tile.
5. Enter provisioning details and click **Provision**.

Results

The selected image is provisioned on the Kubernetes cluster and appears in the sidebar **Requests** window. It is also displayed under **Kubernetes > Deployments** and **Kubernetes > Pods** for verification purposes.

Note You can also provision your cluster by downloading the kubeconfig file and using the command `kubectl`. For more information, see PKS Cluster Details .

Installing Additional Plug-Ins on the Default vRealize Orchestrator Server

You can install additional packages and plug-ins on the default vRealize Orchestrator server by using the vRealize Orchestrator configuration interface.

You can install additional plug-ins on the default vRealize Orchestrator server and use the workflows with XaaS.

You can also import additional packages on the default vRealize Orchestrator server for configuration as vRealize Automation external IPAM provider endpoint types. For example, for information about obtaining, importing, and configuring the Infoblox IPAM package, see Checklist For Providing Third-Party IPAM Provider Support.

Package files (`.package`) and plug-in installation files (`.vmoapp` or `.dar`) are available from the VMware Solution Exchange at https://solutionexchange.vmware.com/store/category_groups/cloud-management. For information about plug-in files, see vRealize Orchestrator Plug-Ins Documentation at https://www.vmware.com/support/pubs/vco_plugins_pubs.html.

For more information about installing new plug-ins, see *Installing and Configuring VMware vCenter Orchestrator*.

Working With Active Directory Policies

Active Directory policies define the properties of a machine record, for example, domain, as well as the organizational unit in which the record is created using a vRealize Automation blueprint.

If you apply a policy to a business group, all the machine requests from the business group members are added to the specified organizational unit. You can create different policies for different organizational units, and then apply the different policies to different business groups.

Using Custom Properties to Override an Active Directory Policy

Using the provided Active Directory custom properties, you can override the Active Directory policy, domain, organizational unit, and other values on a particular blueprint when it is deployed.

The list of the provided Active Directory custom properties is included in the Custom Properties E topic. The custom property prefix is `ext.policy.activedirectory`.

In addition to the provided properties, you can create your own custom properties. You must prefix you custom properties with `ext.policy.activedirectory`. For example, `ext.policy.activedirectory.domain.extension` or `ext.policy.activedirectory.yourproperty`. The properties are passed to your custom vRealize Orchestrator Active Directory workflows.

For more information about custom properties, see Using Custom Properties. For the values that you are overriding, you might need to create a property definition. For example, you might create a property definition that retrieves the available Active Directory policies from vRealize Automation. Alternatively, you might create definition that allows the requesting user to select from two or more alternative organizational units. See Using Property Definitions.

Create and Apply Active Directory Policies

You create one or more Active Directory policies so that you can assign different policies to different business groups. You can use the different policies to add machine records to different organizational units based on business group membership.

If necessary, you can override the assigned Active Directory policy.

Procedure

1 Create an Active Directory Policy

   You create an Active Directory policy to define where records are added in an Active Directory instance when your users deploy machines. You can assign a policy to a business group so that all machines deployed by the business group members result in a record created in the specified organizational unit.

2 Scenario: Add a Custom Property to Blueprints to Override an Active Directory Policy

   As a blueprint architect for the development business group, you have a blueprint that includes an application machine and a database machine. You want the database machine record added to an organizational unit that is different from the applied Active Directory policy.

Create an Active Directory Policy

You create an Active Directory policy to define where records are added in an Active Directory instance when your users deploy machines. You can assign a policy to a business group so that all machines deployed by the business group members result in a record created in the specified organizational unit.

You create different Active Directory policies when you want machines deployed by different business groups to have different domains or to be added to different Active Directory instances.

Prerequisites

- Verify that you created an Active Directory endpoint. See Configure the Active Directory Plug-In as an Endpoint.
- If you use an external vRealize Orchestrator server, verity that it is set up correctly. See Configure an External vRealize Orchestrator Server.
- Log in to vRealize Automation as a **tenant administrator**.

Procedure

1 Select **Administration > Active Directory Policies**.

2 Click the **New** icon (✚).

3 Configure the Active Directory policy details.

Option	Description
ID	Enter the permanent value. The value cannot include any spaces or special characters. You cannot change this value at a later time. You can only re-create the policy with a different ID.
Description	Describe of the policy.
Active Directory Endpoint	Select the Active Directory endpoint for which this policy is created.
Domain	Enter the root domain. The format is *mycompany.com*.
Organizational Unit	Enter the organizational unit distinguished name for this policy. The hierarchy must be entered as a comma-separated list. For example, ou=development,dc=corp,dc=domain,dc=com.

4 Click **OK**.

Results

The vRealize Orchestrator Active Directory endpoint is added to the list. You can apply the policy in business groups or use the policy in blueprints or business groups.

What to do next

- To provide multiple policy options, create more policies.
- To add records to Active Directory based on business group membership when a blueprint is deployed, add the appropriate Active Directory policy to a business group. See Create a Business Group. You can apply the policy when you create the business group, or you can add it later.

- To override the Active Directory policy for the business group for a particular blueprint, add Active Directory custom properties to the blueprint. See Scenario: Add a Custom Property to Blueprints to Override an Active Directory Policy.

Scenario: Add a Custom Property to Blueprints to Override an Active Directory Policy

As a blueprint architect for the development business group, you have a blueprint that includes an application machine and a database machine. You want the database machine record added to an organizational unit that is different from the applied Active Directory policy.

You have an existing policy that is applied to the development business group. The policy adds machine records to ou=development,dc=corp,dc=domain,dc=com. You want all database machines to be added to ou=databases,dc=corp,dc=domain,dc=com. In a blueprint that includes a database server, you override the Active Directory organizational unit to add the database machine record to ou=databases,dc=corp,dc=domain,dc=com.

This scenario makes the following assumptions:

- Your Active Directory includes organizational units for development and databases.
- You have a test blueprint that is included in a service and the service is entitled.

In addition to this simple example of how you can override the policy, you can use custom properties with Active Directory policy to make other changes to Active Directory when you deploy blueprints. See Working With Active Directory Policies.

Prerequisites

- Verify that you have at last one Active Directory policy. See Create an Active Directory Policy. For example, you create a development policy that adds records to ou=development,dc=corp,dc=domain,dc=com.
- Verify that you have a business group to which you applied an Active Directory policy. See Create a Business Group. For example, your development business group uses the development policy.

Procedure

1 In your test blueprint, select the database machine in the canvas.

2 Click the **Properties** tab.

3 Click the **Custom Properties** tab.

4 Click the **New** icon (✚).

5 Add the custom property to change the default organizational unit.

   a In the **Name** text box, enter `ext.policy.activedirectory.orgunit`.

   b In the **Value** text box, enter `ou=databases,dc=corp,dc=domain,dc=com`.

   c Deselect **Overridable**.

   d Click **OK**.

6 Click **Finish**.

Results

The test blueprint includes the custom property, but your users do not see the custom property in the request form.

What to do next

Request your test blueprint. Verify that the record for the database machine was added to the database organizational unit, and that the record for the application machine is added to the development organizational unit. When you are satisfied with the results, you can add the custom property to your production blueprints.

User Preferences for Notifications and Delegates

You use the user preference to override the default configurations for your system approver notifications and your notification language preferences.

To access your user preferences, click your user name on the vRealize Automation header and select **Preferences**.

The following options are specific to you as the logged in user.

Table 4-21. User Preference Options

Option	Description
Assign Delegates	Allows you to reassign your approval requests to other users. For example, you are an approver for catalog requests, but you are going on holiday. You delegate all your approval notifications to one or more approvers. This assignment immediately forwards the requests to your delegate. The delegates are active until you remove them from the list.
Notifications	Allows you to change your notification language so that the email messages are sent to you in the language of your choice rather than the default language. Select the language and add the notification subscription that supports your language preference.

Providing Service Blueprints to Users

5

You deliver on-demand services to users by creating catalog items and actions, then carefully controlling who can request those services by using entitlements and approvals.

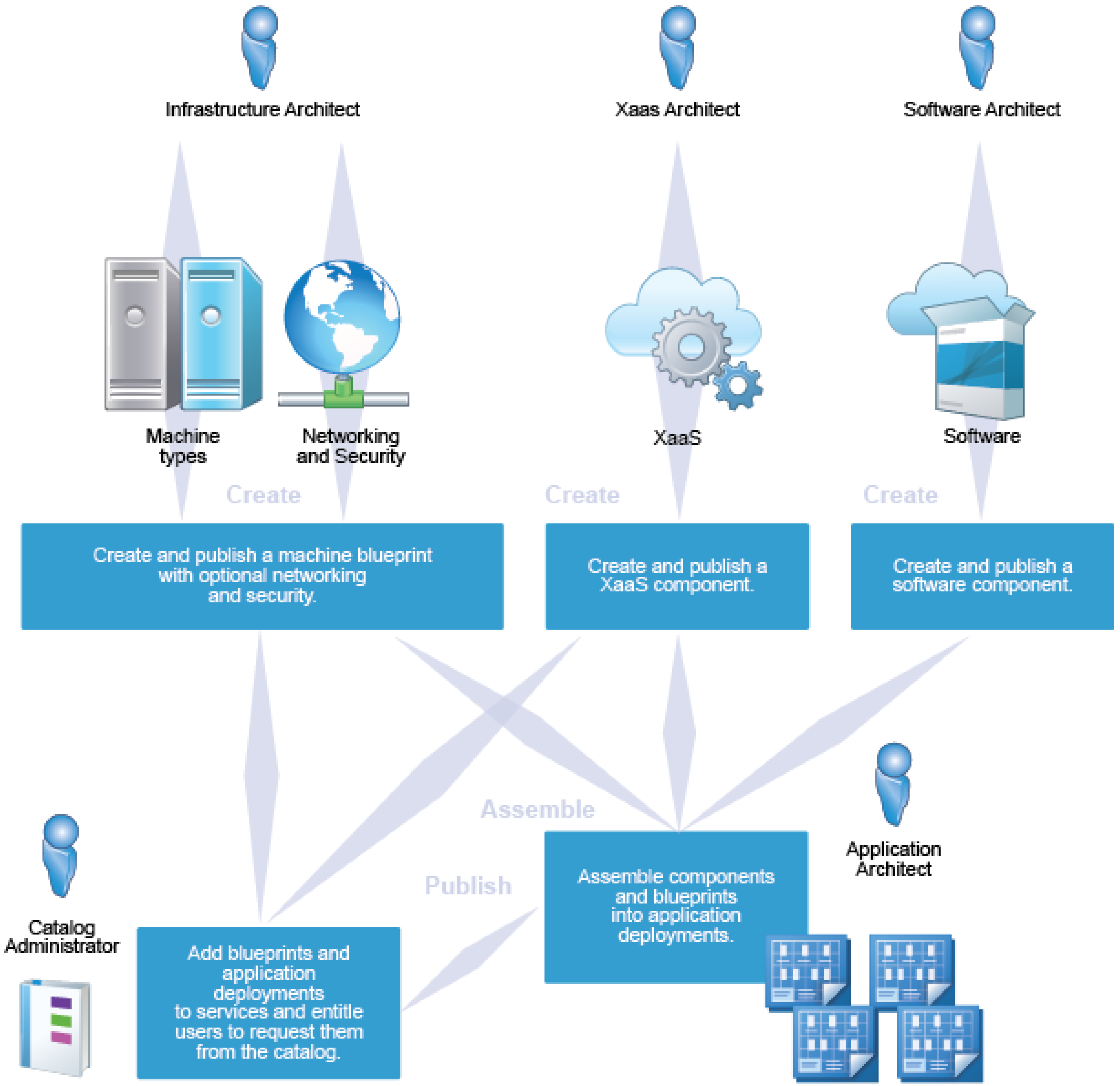
This chapter includes the following topics:

Designing Blueprints

Blueprint architects build Software components, machine blueprints, and custom XaaS blueprints and assemble those components into the blueprints that define the items users request from the catalog. The catalog can display a default request form, or you can create a custom form for each published blueprint.

You can create and publish blueprints for a single machine, or a single custom XaaS blueprint, but you can also combine machine components and XaaS blueprints with other building blocks to design elaborate catalog item blueprints that include multiple machines, networking and security, software with full life cycle support, and custom XaaS functionality.

Depending on the catalog item you want to define, the process can be as simple as a single infrastructure architect publishing one machine component as a blueprint, or the process can include multiple architects creating many different types of components to design a complete application stack for users to request.


Infrastructure Architect
Xaas Architect
Software Architect
Machine types
Networking and Security
XaaS
Software
Create
Create
Create
Create and publish a machine blueprint with optional networking and security.
Create and publish a XaaS component.
Create and publish a software component.
Assemble
Publish
Application Architect
Catalog Administrator
Assemble components and blueprints into application deployments.
Add blueprints and application deployments to services and entitle users to request them from the catalog.

Software Components

You can create and publish software components to install software during the machine provisioning process and support the software life cycle. For example, you can create a blueprint for developers to request a machine with their development environment already installed and configured. Software components are not catalog items by themselves, and you must combine them with a machine component to create a catalog item blueprint. See Designing Software Components.

Machine Blueprints

You can create and publish simple blueprints to provision single machines or you can create more complex blueprints that contain additional machine components and optionally any combination of the following component types:

- Software components
- Existing blueprints
- NSX network and security components
- XaaS components
- Containers components
- Custom or other components

See Designing Machine Blueprints.

XaaS Blueprints

You can publish your vRealize Orchestrator workflows as XaaS blueprints. For example, you can create a custom resource for Active Directory users, and design an XaaS blueprint to allow managers to provision new users in their Active Directory group. You create and manage XaaS components outside of the design tab. You can reuse published XaaS blueprints to create application blueprints, but only in combination with at least one machine component. See Designing XaaS Blueprints and Resource Actions.

Application Blueprints with Multi-Machine, XaaS, and Software Components

You can add any number of machine components, Software components, and XaaS blueprints to a machine blueprint to deliver elaborate functionality to your users.

For example, you can create a blueprint for managers to provision a new hire setup. You can combine multiple machine components, software components, and a XaaS blueprint for provisioning new Active Directory users. The QE Manager can request your New Hire catalog item, and their new quality engineering employee is provisioned in Active Directory and given two working virtual machines, one Windows and one Linux, each with all the required software for running test cases in these environments.

Building Your Design Library

You can build out a library of reusable blueprint components that your architects can assemble into application blueprints for delivering elaborate on-demand services to your users.

Build out a library of the smallest blueprint design components: single machine blueprints, Software components, and XaaS blueprints, then combine these base building blocks in new and different ways to create elaborate catalog items that deliver increasing levels of functionality to your users.

Note that sample blueprints are available at the VMware Solution Exchange at https://solutionexchange.vmware.com and at https://code.vmware.com.

Table 5-1. Building Your Design Library

Catalog Item	Role	Components	Description	Details
Machines	Infrastructure architect	Create machine blueprints on the **Blueprints** tab.	You can create machine blueprints to rapidly deliver virtual, private and public, or hybrid cloud machines to your users. Published machine blueprints are available for catalog administrators to include in the catalog as standalone blueprints, but you can also combine machine blueprints with other components to create more elaborate catalog items that include multiple machine blueprints, Software, or XaaS blueprints.	Configure a Machine Blueprint
NSX Network and security on machines	Infrastructure architect	Add NSX network and security components to vSphere machine blueprints on the **Blueprints** tab.	You can configure network and security components such as network profiles and security groups, to allow virtual machines to communicate with each other over physical and virtual networks securely and efficiently. You must combine network and security components with at least one vSphere machine component before catalog administrators can include them in the catalog. You can only apply NSX network and security components to vSphere machine blueprints.	Designing Blueprints with NSX Settings
Software on machines	Software architect To successfully add software components to the design canvas, you must also have business group member, business group administrator, or tenant administrator role access to the target catalog.	Create and publish Software Components on the **Software** tab, then combine them with machine blueprints on the **Blueprints** tab.	Add Software components to your machine blueprints to standardize, deploy, configure, update, and scale complex applications in cloud environments. These applications can range from simple Web applications to elaborate custom applications and packaged applications. Software components cannot appear in the catalog alone. You must create and publish your Software components and then assemble an application blueprint that contains at least one machine.	Create a Software Component

Table 5-1. Building Your Design Library (continued)

Catalog Item	Role	Components	Description	Details
Custom IT Services	XaaS architects	Create and publish XaaS blueprints on the **XaaS** tab.	You can create XaaS catalog items that extend vRealize Automation functionality beyond machine, networking, security, and software provisioning. Using existing vRealize Orchestrator workflows and plug-ins, or custom scripts you develop in vRealize Orchestrator, you can automate the delivery of any IT services. Published XaaS blueprints are available for catalog administrators to include in the catalog as standalone blueprints, but you can also combine them with other components on the **Blueprints** tab to create more elaborate catalog items.	Designing XaaS Blueprints and Resource Actions
Assemble published blueprint building blocks into new catalog items	■ Application architect ■ Infrastructure architect ■ Software architect	Combine additional machine blueprints, XaaS blueprints, and Software components with at least one machine component or machine blueprint on the **Blueprints** tab.	You can reuse published components and blueprints, combining them in new ways to create IT service packages that deliver elaborate functionality to your users.	Assembling Composite Blueprints

Designing Machine Blueprints

Machine blueprints are the complete specification for a machine, determining a machine's attributes, the manner in which it is provisioned, and its policy and management settings. Depending on the complexity of the catalog item you are building, you can combine one or more machine components in the blueprint with other components in the design canvas to create more elaborate catalog items that include networking and security, Software components, XaaS components, and other blueprint components.

Space-Efficient Storage for Virtual Provisioning

Space-efficient storage technology eliminates the inefficiencies of traditional storage methods by using only the storage actually required for a machine's operations. Typically, this is only a fraction of the storage actually allocated to machines. vRealize Automation supports two methods of provisioning with space-efficient technology, thin provisioning and FlexClone provisioning.

When standard storage is used, the storage allocated to a provisioned machine is fully committed to that machine, even when it is powered off. This can be a significant waste of storage resources because few virtual machines actually use all of the storage allocated to them, just as few physical machines operate with a 100% full disk. When a space-efficient storage technology is used, the storage allocated and the storage used are tracked separately and only the storage used is fully committed to the provisioned machine.

Thin Provisioning

Thin provisioning is supported for all virtual provisioning methods. Depending on your virtualization platform, storage type, and default storage configuration, thin provisioning might always be used during machine provisioning. For example, for vSphere ESX Server integrations using NFS storage, thin provisioning is always employed. However, for vSphere ESX Server integrations that use local or iSCSI storage, thin provisioning is only used to provision machines if the custom property `VirtualMachine.Admin.ThinProvision` is specified in the blueprint. For more information about thin provisioning, please see the documentation provided by your virtualization platform.

Net App FlexClone Provisioning

You can create a blueprint for Net App FlexClone provisioning if you are working in a vSphere environment that uses Network File System (NFS) storage and FlexClone technology.

You can only use NFS storage, or machine provisioning fails. You can specify a FlexClone storage path for other types of machine provisioning, but the FlexClone storage path behaves like standard storage.

The following is a high-level overview of the sequence of steps required to provision machines that use FlexClone technology:

1. An IaaS administrator creates a NetApp ONTAP endpoint. See Endpoint Settings Reference.
2. An IaaS administrator runs data collection on the endpoint to enable the endpoint to be visible on the compute resource and reservation pages.

   The FlexClone option is visible on a reservation page in the endpoint column if a NetApp ONTAP endpoint exists and if the host is virtual. If there is a NetApp ONTAP endpoint, the reservation page displays the endpoint assigned to the storage path.
3. A fabric administrator creates a vSphere reservation, enables FlexClone storage, and specifies an NFS storage path that uses FlexClone technology. See Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer.
4. An infrastructure architect or other authorized user creates a blueprint for FlexClone provisioning.

Understanding and Using Blueprint Parameterization

You can use component profiles to parameterize blueprints. Rather than create a separate small, medium, and large blueprint for a particular deployment type, you can create a single blueprint with a choice of small, medium, or large size virtual machine. Users can select one of these sizes when they deploy the catalog item.

Component profiles minimize blueprint sprawl and simplify your catalog offerings. You can use component profiles to define vSphere machine components in a blueprint. The available component profile types are `Size` and `Image`. When you add component profiles to a machine component, the component profile settings override other settings on the machine component, such as number of CPUs or amount of storage.

Component profiles are only available for vSphere machine components.

For information about defining value sets for the `Size` and `Image` component profiles, see Defining Component Profile Settings.

For information about adding component profiles and selected value sets for a vSphere machine component in a blueprint, see vSphere Machine Component Settings in vRealize Automation.

For information about adding component profile information by using settings imported from an OVF, see Configuring a Blueprint to Provision from an OVF.

For information about using component profiles when requesting machine provisioning, see Request Machine Provisioning By Using a Parameterized Blueprint.

You can create approval policies to require pre-approval when requesting machine provisioning of blueprints relative to value set conditions for the `Size` and `Image` component profile. For more information, see Examples of Approval Policies Based on the Virtual Machine Policy Type.

Note

For information about using blueprint parameterization when requesting machine provisioning from the catalog, see Request Machine Provisioning By Using a Parameterized Blueprint.

Configure a Machine Blueprint

Configure and publish a machine component as a standalone blueprint that other architects can reuse as a component in application blueprints, and catalog administrators can include in catalog services.

This procedure provides a simple overview of the blueprint creation process. For added detail, see the following:

- Designing Blueprints with NSX Settings
- Understanding and Using Blueprint Parameterization
- Blueprint Properties Settings
- Configuring a Blueprint to Provision from an OVF
- Exporting and Importing Blueprints and Content
- Creating Microsoft Azure Blueprints and Incorporating Resource Actions
- Adding Configuration Management Capabilities to vSphere Blueprints

Prerequisites

- Log in to vRealize Automation as an **infrastructure architect**.
- Complete external preparations for provisioning, such as creating templates, WinPEs, and ISOs, or gather the information about external preparations from your administrators.
- Configure your tenant. See Configuring Tenant Settings.
- Configure your IaaS resources. See Checklist for Configuring IaaS Resources.
- See Preparing Your Environment for vRealize Automation Management.

Procedure

1. Select **Design > Blueprints**.
2. Click the **New** icon (➕).
3. Follow the prompts on the **New Blueprint** dialog box to configure general settings.
4. Click **OK**.
5. Click **Machine Types** in the Categories area to display a list of available machine types.
6. Drag the type of machine you want to provision onto the design canvas.
7. Enter information on each of the tabs to configure machine provisioning details as described in Blueprint Properties Settings.
8. Click **Finish**.
9. Select your blueprint and click **Publish**.

Results

You configured and published a machine component as a standalone blueprint. Catalog administrators can include this machine blueprint in catalog services and entitle users to request this blueprint. Other architects can reuse this machine blueprint to create more elaborate application blueprints that include Software components, XaaS blueprints, or additional machine blueprints.

What to do next

You can combine a machine blueprint with Software components, XaaS blueprints, or additional machine blueprints to create more elaborate application blueprints. See Assembling Composite Blueprints and Understanding Nested Blueprint Behavior.

Machine Blueprint Settings

You can define configuration settings and custom properties for the overall blueprint.

Blueprint Properties Settings

You can specify settings that apply to the entire blueprint by using the **Blueprint Properties** page when you create the blueprint. After you create the blueprint, you can edit these settings on the Blueprint Properties page.

General Tab

Settings on the General tab apply to the overall vRealize Automation blueprint.

Table 5-2. **General** Tab Settings

Setting	Description
Name	Enter a name for your blueprint.
Identifier	The identifier field automatically populates based on the name you entered. You can edit this field now, but after you save the blueprint you can never change it. Identifiers are permanent and unique within your tenant. You can use them to programmatically interact with blueprints and to create property bindings.
Description	Summarize your blueprint for the benefit of other architects. This description also appears to users on the request form.
Deployment limit	Specify the maximum number of deployments that can be created when this blueprint is used to provision machines.
Lease days: Minimum and **Maximum**	Enter a minimum and maximum value to allow users to choose from within a range of lease lengths. When the lease ends, the deployment is either destroyed or archived. If you do not specify a minimum or maximum value, the lease is set to never expire. Enter the lease information for your machines in your vRealize Automation blueprint, not in the source endpoint application. If you specify the lease information in an external application, it is not recognized in vRealize Automation.
Archive days	You can specify an archival period to temporarily retain deployments instead of destroying deployments as soon as their lease expires. Specify 0 to destroy the deployment when its lease expires. The archive period begins on the day the lease expires. When the archive period ends, the deployment is destroyed. The default is 0.
Propagate updates to existing deployments	Broadened minimum-maximum ranges for CPU, memory, or storage are pushed to active deployments that were provisioned from the blueprint. The new range must fully encompass the old range. For example, for an original minimum 32 and maximum 128 (32, 128), a change such as (16, 128) or (32, 256) or (2, 1000) can take effect upon reconfiguration or scale-out, but a change of (33, 512) or (4, 64) cannot. The changes take effect upon the next reconfigure or scale-out action. For more information, see Action Menu Commands for Provisioned Resources.

NSX Settings Tab

If you configured NSX, you can specify NSX transport zone, network reservation policy, and app isolation settings when you create or edit a blueprint. These settings are available on the **NSX Settings** tab on the **Blueprint** and **Blueprint Properties** pages.

For information about NSX settings, see New Blueprint and Blueprint Properties Page Settings with NSX in vRealize Automation.

Properties Tab

Custom properties that you add at the blueprint level apply to the entire blueprint, including all components. However, they can be overridden by other custom properties. For information about the order of precedence for custom properties, see Understanding Custom Properties Precedence.

Table 5-3. **Properties** Tab Settings

Tab	Setting	Description
Property Groups	Property groups are reusable groups of properties that simplify the process of adding custom properties to blueprints.	
	Add	Add one or more existing property groups and apply them to the overall blueprint. The following Containers-related property groups are supplied: ■ Container host properties with certificate authentication ■ Container host properties with user/ password authentication
	Move up /Move down	Control the order of precedence given to each property group in relation to one another by prioritizing the groups. The first group in the list has the highest priority, and its custom properties have first precedence. You can also slide to reorder.
	View properties	View the custom properties in the selected property group.
	View merged properties	If a custom property is included in more than one property group, the value included in the property group with the highest priority takes precedence.
Custom Properties	You can add individual custom properties instead of property groups.	
	New	Add an individual custom property and apply it to the overall blueprint.
	Name	Enter the property name. For a list of custom property names and descriptions, see Chapter 8 Custom Properties and the Property Dictionary .

Table 5-3. **Properties** Tab Settings (continued)

Tab	Setting	Description
	Value	Enter the value for the custom property.
	Encrypted	Encrypt the property value, for example, if the value is a password.
	Overridable	The blueprint user can override the property value. If you select **Show in request**, users can see and edit property values when they request catalog items.
	Show in request	The property name and value is visible to users on the provisioning request form. Select **Overridable** if allow users to provide a value.

vSphere Machine Component Settings in vRealize Automation

Understand the settings and options that you can configure for a vSphere machine component in the vRealize Automation blueprint design canvas.

General Tab

Configure general settings for a vSphere machine component.

Table 5-4. **General** Tab Settings

Setting	Description
ID	Enter a name for your machine component, or accept the default.
Description	Summarize your machine component for the benefit of other architects.
Display location on request	In a cloud environment, such as vCloud Air, this allows users to select a region for their provisioned machines. For a virtual environment, you can allow users to select a data center location at which to provision a requested machine. A system administrator must add data center information to a locations file. A fabric administrator must edit a compute resource to associate it with a location. See Scenario: Add Datacenter Locations for Cross Region Deployments and Scenario: Apply a Location to a Compute Resource for Cross Region Deployments .
Reservation policy	Apply a reservation policy to a blueprint to restrict the machines provisioned from that blueprint to a subset of available reservations. Only the reservation policies that are applicable to the current tenant are available. For information about creating reservation policies, see Configure a Reservation Policy.

Table 5-4. **General** Tab Settings (continued)

Setting	Description
Machine prefix	Machine prefixes are used to name provisioned machines. If you select **Use group default**, machines are named based on the default machine prefix for your business group. If you do not specify a prefix, one is generated for you based your business group name. Only the machine prefixes that are applicable to the current tenant are available. If your fabric administrator configures other machine prefixes for you to select, you can apply one prefix to all machines provisioned from your blueprint, no matter who the requestor is. For information about creating machine prefixes, see Configure Machine Prefixes.
Instances: Minimum and **Maximum**	Configure the maximum and minimum number of instances users can request for a deployment or for a scale in or scale out action. Entering the same value in the **Minimum** and **Maximum** fields configures exactly how many instances to provision. XaaS components are not scalable and are not updated during a scale operation. If you are using XaaS components in your blueprint, you might create a resource action for users to run after a scale operation, which might either scale or update your XaaS components as required. You can deactivate scaling by configuring the number of instances to allow for each machine component.

Build Information Tab

Configure build information settings for a vSphere machine component.

Table 5-5. **Build Information** Tab

Setting	Description
Blueprint type	For record-keeping and licensing purposes, select whether machines provisioned from this blueprint are classified as Desktop or Server.
Action	The options you see in the action drop-down menu depend on the type of machine you select. The following actions are available: ■ **Create** Create the machine component specification without use of a cloning option. ■ **Clone** Make copies of a virtual machine from a template and customization object. ■ **Linked Clone** Provision a space-efficient copy of a virtual machine called a linked clone. Linked clones are based on a snapshot of a VM and use a chain of delta disks to track differences from a parent machine. Before you provision linked clone VMs, power off the VM snapshot. ■ **NetApp FlexClone** If your reservations use NetApp FlexClone storage, you can clone space-efficient copies of machines.

Table 5-5. **Build Information** Tab (continued)

Setting	Description
Provisioning workflow	The options you see in the provisioning workflow drop-down menu depend on the type of machine you select, and the action you select. - **BasicVmWorkflow** Provision a machine with no guest operating system. - **ExternalProvisioningWorkflow** Create a machine by starting from either a virtual machine instance or cloud-based image. - **ImportOvfWorkflow** Allows you to deploy a vSphere virtual machine from an OVF template in the same manner as a CloneWorkflow allows you to deploy a vSphere virtual machine from a virtual machine template. You can import to a vSphere component in a machine blueprint or to an `Image` component profile for a parameterized blueprint. - **LinuxKickstartWorkflow** Provision a machine by booting from an ISO image, using a kickstart or autoYaSt configuration file and a Linux distribution image to install the operating system on the machine. - **VirtualSccmProvisioningWorkflow** Provision a machine and pass control to an SCCM task sequence to boot from an ISO image, deploy a Windows operating system, and install the vRealize Automation guest agent. - **WIMImageWorkflow** Provision a machine by booting into a WinPE environment and installing an operating system using a Windows Imaging File Format (WIM) image of an existing Windows reference machine. When using a WIM provisioning workflow in a blueprint, specify a storage value that accounts for the size of each disk to be used on the machine. Use the total value of all disks as the minimum storage value for the machine component. Also specify a size for each disk that is large enough to accommodate the operating system.
Clone from	Select a machine template to clone from. You can refine the list of available templates by using the **Filters** option in each column drop-down menu. For Linked Clone, you only see machines that have available snapshots to clone from and that you manage as a tenant administrator or business group manager. You can only clone from templates that exist on machines that you manage as a business group manager or tenant administrator.

Table 5-5. **Build Information** Tab (continued)

Setting	Description
Clone from snapshot	For Linked Clone, select an existing snapshot to clone from based on the selected machine template. Machines only appear in the list if they already have an existing snapshot, and if you manage that machine as a tenant administrator or business group manager. If you select **Use current snapshot**, the clone is defined with the same characteristics as the latest state of the virtual machine. If you instead want to clone relative to an actual snapshot, click the drop-down menu option and select the specific snapshot from the list. **Note** Use of the term snapshot can be confusing. If you select an existing snapshot, the option creates a new disk that is parented by the snapshot. The **Use current snapshot** option has no base disk to use as a parent and silently performs a full clone action. As a workaround, you can create snapshots on the base disk, or use a vRealize Orchestrator workflow to create a snapshot and then clone immediately from the snapshot. This option is only available for the Linked Clone action.
Customization spec	Specify an available customization specification. A customization spec is required only if you are cloning with static IP addresses. You cannot perform customization of Windows machines without a customization specification. For Linux clone machines, you can perform customization by using a customization spec, an external script, or both.

Machine Resources Tab

Specify CPU, memory, and storage settings for a vSphere machine component.

Table 5-6. **Machine Resources** Tab

Setting	Description
CPUs: Minimum and **Maximum**	Enter a minimum and maximum number of CPUs that can be used by provisioned machines.
Memory (MB): Minimum and **Maximum**	Enter the minimum and maximum amount of memory that can be used by provisioned machines.
Storage (GB): Minimum and **Maximum**	Enter a minimum and maximum amount of storage that can be used by provisioned machines. When using a WIM provisioning workflow in a blueprint, specify a storage value that accounts for the size of each disk to be used on the machine. Use the total value of all disks as the minimum storage value for the machine component. Also specify a size for each disk that is large enough to accommodate the operating system.

Storage Tab

You can add storage volume settings, including one or more storage reservation policies, to the machine component to control storage space.

Table 5-7. **Storage** Tab Settings

Setting	Description
ID	Enter an ID or name for the storage volume.
Capacity (GB)	Enter the storage capacity for the storage volume.
Drive Letter/Mount Path	Enter a drive letter or mount path for the storage volume. This option is used during provisioning in association with a guest agent. It cannot be changed after machine provisioning. If you are not using a guest agent, this option is ignored.
Label	Enter a label for the drive letter and mount path for the storage volume. This option is used during provisioning in association with a guest agent. It cannot be changed after machine provisioning. If you are not using a guest agent, this option is ignored.
Storage Reservation Policy	Enter the existing storage reservation policy to use with this storage volume. Only the storage reservation policies that are applicable to the current tenant are available.
Custom Properties	Enter any custom properties to use with this storage volume.
Maximum volumes	Enter the maximum number of allowed storage volumes that can be used when provisioning from the machine component. Enter 0 to prevent others from adding storage volumes. The default value is 60.
Allow users to see and change storage reservation policies	Select the check box to allow users to remove an associated reservation policy or specify a different reservation policy when provisioning.

Network Tab

You can configure network settings for a vSphere machine component based on NSX network and load balancer settings that are configured outside vRealize Automation. You can use settings from one or more existing and on-demand NSX network components in the design canvas.

For related information, see Configuring Network and Security Component Settings in vRealize Automation and New Blueprint and Blueprint Properties Page Settings with NSX in vRealize Automation.

Table 5-8. **Network** Tab Settings

Setting	Description
Network	Select a network component from the drop-down menu. Only network components that exist in the design canvas are listed. Only the network profiles that are applicable to the current tenant are available. The network you select determines network type and also whether the cluster to be deployed on the network is managed by NSX for vSphere or NSX-T.
Assignment Type	Accept the default assignment derived from the network component or select an assignment type from the drop-down menu. The **DHCP** and **Static** option values are derived from settings in the network component.
Address	Specify the IP address for the network. The option is available only for the static address type.
Load Balancing	Enter the service to use for load balancing.
Custom Properties	Display custom properties that are configured for the selected network component or network profile.
Maximum network adapters	Specify the maximum number of network adapters, or NICs, to allow for this machine component. The default is unlimited. Set to 0 to deactivate adding NICs for the machine components.

Security Tab

You can configure security settings for a vSphere machine component based on NSX settings that are configured outside vRealize Automation. You an optionally use settings from existing and on-demand NSX security components in the design canvas.

The security settings from existing and on-demand security group and security tag components in the design canvas are automatically available.

For information about adding and configuring NSX network and security components before using security tab settings on a vSphere machine component, see Configuring Network and Security Component Settings in vRealize Automation.

For information about specifying NSX information that applies to all the vSphere machine components in the blueprint, see New Blueprint and Blueprint Properties Page Settings with NSX in vRealize Automation.

Table 5-9. **Security** Tab Settings

Setting	Description
Name	Display the name of an NSX security group or tag. The names are derived from security components in the design canvas. Select the check box next to a listed security group or tag to use that group or tag for provisioning from this machine component.
Type	Indicate if the security element is an on-demand security group, an existing security group, or a security tag.
Description	Display the description defined for the security group or tag.
Endpoint	Display the endpoint used by the NSX security group or tag.

Properties Tab

Specify custom property and property group information for a vSphere machine component.

You can add individual and groups of custom properties to the machine component by using the **Properties** tab. You can add also custom properties and property groups to the overall blueprint by using the **Properties** tab when you create or edit a blueprint by using the **Blueprint Properties** page.

You can use the **Custom Properties** tab to add and configure options for existing custom properties. Custom properties are supplied with vRealize Automation and you can also create property definitions.

Table 5-10. **Properties > Custom Properties** Tab Settings

Setting	Description
Name	Enter the name of a custom property or select an available custom property from the drop-down menu. Properties only appear in the drop-down menu if your tenant administrator or fabric administrator created property definitions.
Value	Enter or edit a value to associate with the custom property name. For example, set the value as `true` to allow entitled users to connect to VMs by using SSH.
Encrypted	You can choose to encrypt the property value, for example if the value is a password.
Overridable	You can specify that the property value can be overridden by the next or subsequent person who uses the property. If you select **Show in request**, users can edit property values when they request catalog items.
Show in Request	You can display the property name and value to users when they request machine provisioning. Select the overridable option if you want users to provide a value.

You can use the **Property Groups** tab to add and configure settings for existing custom property groups. You can create your own property groups or use property groups that have been created for you.

Table 5-11. **Properties** > **Property Groups** Tab Settings

Setting	Description
Name	Select an available property group from the drop-down menu.
Move Up and **Move Down**	Control the precedence level of property groups in descending order. The first-listed property group has precedence over the next-listed property group and so on.
View Properties	Display the custom properties in the selected property group.
View Merged Properties	Display custom properties in the order they appear in the list of property groups. Where the same property appears in more than one group, the property appears once in the list based on when it is first encountered.

Profiles Tab

Component profiles provide a means of parameterizing blueprints. For example, rather than creating separate blueprints, you can create a small, medium, and large capability in a single blueprint. You can select a blueprint size during deployment. Component profiles are designed to simplify your catalog.

If you have created value sets for the supplied vRealize Automation component profiles `Size` and `Image`, you can configure those machine component settings in the blueprint. You can also select a different value set when you deploy the catalog item.

Component profiles are only available for vSphere machine components.

A component profile overrides settings on the machine component, such as number of CPUs and storage.

The component profile value set is applied to all vSphere machines in a cluster.

You cannot reconfigure machines by using the `Size` or `Image` component profiles. The range of CPU, memory, and storage is calculated from the profile remains available for reconfigure actions. For example, use a small (1 CPU, 1024 MB memory, and 10 GB storage), medium (3 CPUs, 2048 MB memory, 12 GB storage) and large (5 CPUs, 3072 MB memory, 15 GB storage) `Size` value set. The available ranges during machine reconfiguration are 1-5 CPUs, 1024-3072 memory, and 1-15 GB storage.

For more information, see Understanding and Using Blueprint Parameterization.

For more information, see Defining Component Profile Settings.

Table 5-12. **Profiles** Tab Settings

Setting	Description
Add	Add the `Size` or `Image` component profile.
Edit Value Sets	Assign one or more value sets for the selected component profile by selecting from a list of defined value sets. You can select one of the value sets as the default.
Remove	Remove the `Size` or `Image` component profile.

vCloud Air Machine Component Settings

Understand the settings and options that you can configure for a vCloud Air machine component in the vRealize Automation blueprint design canvas.

General Tab

Configure general settings for a vCloud Air machine component.

Table 5-13. **General** Tab Settings

Setting	Description
ID	Enter a name for your machine component, or accept the default.
Description	Summarize your machine component for the benefit of other architects.
Display location on request	In a cloud environment, such as vCloud Air, this allows users to select a region for their provisioned machines. For a virtual environment, you can allow users to select a data center location at which to provision a requested machine. A system administrator must add data center information to a locations file. A fabric administrator must edit a compute resource to associate it with a location. See Scenario: Add Datacenter Locations for Cross Region Deployments and Scenario: Apply a Location to a Compute Resource for Cross Region Deployments .
Reservation policy	Apply a reservation policy to a blueprint to restrict the machines provisioned from that blueprint to a subset of available reservations. Only the reservation policies that are applicable to the current tenant are available. For information about creating reservation policies, see Configure a Reservation Policy.

Table 5-13. General Tab Settings (continued)

Setting	Description
Machine prefix	Machine prefixes are used to name provisioned machines. If you select **Use group default**, machines are named based on the default machine prefix for your business group. If you do not specify a prefix, one is generated for you based your business group name. Only the machine prefixes that are applicable to the current tenant are available. If your fabric administrator configures other machine prefixes for you to select, you can apply one prefix to all machines provisioned from your blueprint, no matter who the requestor is. For information about creating machine prefixes, see Configure Machine Prefixes.
Instances: Minimum and **Maximum**	Configure the maximum and minimum number of instances users can request for a deployment or for a scale in or scale out action. Entering the same value in the **Minimum** and **Maximum** fields configures exactly how many instances to provision. XaaS components are not scalable and are not updated during a scale operation. If you are using XaaS components in your blueprint, you might create a resource action for users to run after a scale operation, which might either scale or update your XaaS components as required. You can deactivate scaling by configuring the number of instances to allow for each machine component.

Build Information Tab

Configure build information settings for a vCloud Air machine component.

Table 5-14. Build Information Tab

Setting	Description
Blueprint type	For record-keeping and licensing purposes, select whether machines provisioned from this blueprint are classified as Desktop or Server.
Action	The options you see in the action drop-down menu depend on the type of machine you select. The only provisioning action available for a vCloud Air machine component is Clone. ■ **Clone** Make copies of a virtual machine from a template and customization object.

Table 5-14. **Build Information** Tab (continued)

Setting	Description
Provisioning workflow	The options you see in the provisioning workflow drop-down menu depend on the type of machine you select, and the action you select. The only provisioning action available for a vCloud Air machine component is CloneWorkflow. ■ **CloneWorkflow** Make copies of a virtual machine, either by Clone, Linked Clone, or NetApp Flexclone.
Clone from	Select a machine template to clone from. You can refine the list of available templates by using the **Filters** option in each column drop-down menu. For Linked Clone, you only see machines that have available snapshots to clone from and that you manage as a tenant administrator or business group manager. You can only clone from templates that exist on machines that you manage as a business group manager or tenant administrator.

Machine Resources Tab

Specify CPU, memory and storage settings for your vCloud Air machine component.

Table 5-15. **Machine Resources** Tab

Setting	Description
CPUs: Minimum and **Maximum**	Enter a minimum and maximum number of CPUs that can be used by provisioned machines.
Memory (MB): Minimum and **Maximum**	Enter the minimum and maximum amount of memory that can be used by provisioned machines.
Storage (GB): Minimum and **Maximum**	Enter a minimum and maximum amount of storage that can be used by provisioned machines.

Storage Tab

You can add storage volume settings, including one or more storage reservation policies, to the machine component to control storage space.

Table 5-16. **Storage** Tab Settings

Setting	Description
ID	Enter an ID or name for the storage volume.
Capacity (GB)	Enter the storage capacity for the storage volume.
Drive Letter/Mount Path	Enter a drive letter or mount path for the storage volume. This option is used during provisioning in association with a guest agent. It cannot be changed after machine provisioning. If you are not using a guest agent, this option is ignored.

Table 5-16. **Storage** Tab Settings (continued)

Setting	Description
Label	Enter a label for the drive letter and mount path for the storage volume. This option is used during provisioning in association with a guest agent. It cannot be changed after machine provisioning. If you are not using a guest agent, this option is ignored.
Storage Reservation Policy	Enter the existing storage reservation policy to use with this storage volume. Only the storage reservation policies that are applicable to the current tenant are available.
Custom Properties	Enter any custom properties to use with this storage volume.
Maximum volumes	Enter the maximum number of allowed storage volumes that can be used when provisioning from the machine component. Enter 0 to prevent others from adding storage volumes. The default value is 60.
Allow users to see and change storage reservation policies	Select the check box to allow users to remove an associated reservation policy or specify a different reservation policy when provisioning.

Properties Tab

Optionally specify custom property and property group information for your vCloud Air machine component.

You can add individual and groups of custom properties to the machine component by using the **Properties** tab. You can add also custom properties and property groups to the overall blueprint by using the **Properties** tab when you create or edit a blueprint by using the **Blueprint Properties** page.

You can use the **Custom Properties** tab to add and configure options for existing custom properties. Custom properties are supplied with vRealize Automation and you can also create property definitions.

Table 5-17. **Properties > Custom Properties** Tab Settings

Setting	Description
Name	Enter the name of a custom property or select an available custom property from the drop-down menu. Properties only appear in the drop-down menu if your tenant administrator or fabric administrator created property definitions.
Value	Enter or edit a value to associate with the custom property name. For example, set the value as `true` to allow entitled users to connect to VMs by using SSH.
Encrypted	You can choose to encrypt the property value, for example if the value is a password.

Table 5-17. **Properties > Custom Properties** Tab Settings (continued)

Setting	Description
Overridable	You can specify that the property value can be overridden by the next or subsequent person who uses the property. If you select **Show in request**, users can edit property values when they request catalog items.
Show in Request	You can display the property name and value to users when they request machine provisioning. Select the overridable option if you want users to provide a value.

You can use the **Property Groups** tab to add and configure settings for existing custom property groups. You can create your own property groups or use property groups that have been created for you.

Table 5-18. **Properties > Property Groups** Tab Settings

Setting	Description
Name	Select an available property group from the drop-down menu.
Move Up and **Move Down**	Control the precedence level of property groups in descending order. The first-listed property group has precedence over the next-listed property group and so on.
View Properties	Display the custom properties in the selected property group.
View Merged Properties	Display custom properties in the order they appear in the list of property groups. Where the same property appears in more than one group, the property appears once in the list based on when it is first encountered.

Amazon Machine Component Settings

Understand the settings and options that you can configure for an Amazon machine component in the vRealize Automation blueprint design canvas.

General Tab

Configure general settings for an Amazon machine component.

Table 5-19. **General** Tab Settings

Setting	Description
ID	Enter a name for your machine component, or accept the default.
Description	Summarize your machine component for the benefit of other architects.

Table 5-19. **General** Tab Settings (continued)

Setting	Description
Display location on request	In a cloud environment, such as vCloud Air, this allows users to select a region for their provisioned machines. For a virtual environment, you can allow users to select a data center location at which to provision a requested machine. A system administrator must add data center information to a locations file. A fabric administrator must edit a compute resource to associate it with a location. See Scenario: Add Datacenter Locations for Cross Region Deployments and Scenario: Apply a Location to a Compute Resource for Cross Region Deployments .
Reservation policy	Apply a reservation policy to a blueprint to restrict the machines provisioned from that blueprint to a subset of available reservations. Only the reservation policies that are applicable to the current tenant are available. For information about creating reservation policies, see Configure a Reservation Policy.
Machine prefix	Machine prefixes are used to name provisioned machines. If you select **Use group default**, machines are named based on the default machine prefix for your business group. If you do not specify a prefix, one is generated for you based your business group name. Only the machine prefixes that are applicable to the current tenant are available. If your fabric administrator configures other machine prefixes for you to select, you can apply one prefix to all machines provisioned from your blueprint, no matter who the requestor is. For information about creating machine prefixes, see Configure Machine Prefixes.
Instances: Minimum and **Maximum**	Configure the maximum and minimum number of instances users can request for a deployment or for a scale in or scale out action. Entering the same value in the **Minimum** and **Maximum** fields configures exactly how many instances to provision. XaaS components are not scalable and are not updated during a scale operation. If you are using XaaS components in your blueprint, you might create a resource action for users to run after a scale operation, which might either scale or update your XaaS components as required. You can deactivate scaling by configuring the number of instances to allow for each machine component.

Build Information Tab

Configure build information settings for an Amazon machine component.

Table 5-20. Build Information Tab

Setting	Description
Blueprint type	For record-keeping and licensing purposes, select whether machines provisioned from this blueprint are classified as Desktop or Server.
Provisioning workflow	The only provisioning workflow available for an Amazon machine component is CloudProvisioningWorkflow. ■ **CloudProvisioningWorkflow** Create a machine by starting from either a virtual machine instance or cloud-based image.
Amazon machine image	Select an available Amazon machine image. An Amazon machine image is a template that contains a software configuration, including an operating system. Machine images are managed by Amazon Web Services accounts. You can refine the list of Amazon machine image names in the display by using the **Filters** option in the **AMI ID** column drop-down menu.
Key pair	Key pairs are required for provisioning with Amazon Web Services. Key pairs are used to provision and connect to a cloud instance. They are also used to decrypt Windows passwords and to log in to a Linux machine. The following key pair options are available: ■ Not specified Controls key pair behavior at the blueprint level rather than at the reservation level. ■ Auto-generated per business group Specifies that each machine provisioned in the same business group has the same key pair, including machines provisioned on other reservations when the machine has the same compute resource and business group. Because the key pairs are associated with a business group, the key pairs are deleted when the business group is deleted. ■ Auto-generated per machine Specifies that each machine has a unique key pair. The auto-generated per machine option is the most secure method because no key pairs are shared among machines.

Table 5-20. Build Information Tab (continued)

Setting	Description
Enable Amazon network options on machine	Choose whether to allow users to provision a machine in a virtual private cloud (VPC) or a non-VPC location when they submit the request.
Instance types	Select one or more Amazon instance types. An Amazon instance is a virtual server that can run applications in Amazon Web Services. Instances are created from an Amazon machine image and by choosing an appropriate instance type. vRealize Automation manages the machine image instance types that are available for provisioning. For information about using Amazon instance types in vRealize Automation, see Understanding Amazon Instance Types and Add an Amazon Instance Type.

Machine Resources Tab

Specify CPU, memory, storage, and EBS volume settings for your Amazon machine component.

You can also reconfigure all Amazon machine storage volumes in the deployment except for the root volume.

Table 5-21. **Machine Resources** Tab

Setting	Description
CPUs: Minimum and **Maximum**	Enter a minimum and maximum number of CPUs that can be used by provisioned machines.
Memory (MB): Minimum and **Maximum**	Enter the minimum and maximum amount of memory that can be used by provisioned machines.
Storage (GB): Minimum and **Maximum**	Enter a minimum and maximum amount of storage that can be used by provisioned machines.

Table 5-21. **Machine Resources** Tab (continued)

Setting	Description
EBS Storage (GB): Minimum and **Maximum**	Enter a minimum and maximum amount of Amazon Elastic Block Store (EBS) storage volume that can be used by provisioned machines. When destroying a deployment that contains an Amazon machine component, all EBS volumes that were added to the machine during its life cycle are detached, rather than destroyed. vRealize Automation does not provide an option for destroying the EBS volumes.
Delete Volumes	Specifies whether you can delete EC2 volumes individually or in bulk when destroying Amazon deployments. Both Yes and No allow for a bulk destroy action of all volumes in the deployment. The default value is null or empty. ■ Yes - destroy the Amazon deployment and delete volumes. ■ No - destroy the Amazon deployment and keep volumes. ■ null or empty - requires the user to specify the Yes or No value when destroying Amazon deployments. For related information about the Destroy command, see Action Menu Commands for Provisioned Resources.

Properties Tab

Optionally specify custom property and property group information for your Amazon machine component.

You can add individual and groups of custom properties to the machine component by using the **Properties** tab. You can add also custom properties and property groups to the overall blueprint by using the **Properties** tab when you create or edit a blueprint by using the **Blueprint Properties** page.

You can use the **Custom Properties** tab to add and configure options for existing custom properties. Custom properties are supplied with vRealize Automation and you can also create property definitions.

Table 5-22. **Properties** > **Custom Properties** Tab Settings

Setting	Description
Name	Enter the name of a custom property or select an available custom property from the drop-down menu. Properties only appear in the drop-down menu if your tenant administrator or fabric administrator created property definitions.
Value	Enter or edit a value to associate with the custom property name. For example, set the value as `true` to allow entitled users to connect to VMs by using SSH.

Table 5-22. **Properties > Custom Properties** Tab Settings (continued)

Setting	Description
Encrypted	You can choose to encrypt the property value, for example if the value is a password.
Overridable	You can specify that the property value can be overridden by the next or subsequent person who uses the property. If you select **Show in request**, users can edit property values when they request catalog items.
Show in Request	You can display the property name and value to users when they request machine provisioning. Select the overridable option if you want users to provide a value.

You can use the **Property Groups** tab to add and configure settings for existing custom property groups. You can create your own property groups or use property groups that have been created for you.

Table 5-23. **Properties > Property Groups** Tab Settings

Setting	Description
Name	Select an available property group from the drop-down menu.
Move Up and **Move Down**	Control the precedence level of property groups in descending order. The first-listed property group has precedence over the next-listed property group and so on.
View Properties	Display the custom properties in the selected property group.
View Merged Properties	Display custom properties in the order they appear in the list of property groups. Where the same property appears in more than one group, the property appears once in the list based on when it is first encountered.

OpenStack Machine Component Settings

Understand the settings and options you can configure for an OpenStack machine component in the vRealize Automation blueprint design canvas.

General Tab

Configure general settings for an OpenStack machine component.

Table 5-24. **General** Tab Settings

Setting	Description
ID	Enter a name for your machine component, or accept the default.
Description	Summarize your machine component for the benefit of other architects.

Table 5-24. **General** Tab Settings (continued)

Setting	Description
Display location on request	In a cloud environment, such as vCloud Air, this allows users to select a region for their provisioned machines. For a virtual environment, you can allow users to select a data center location at which to provision a requested machine. A system administrator must add data center information to a locations file. A fabric administrator must edit a compute resource to associate it with a location. See Scenario: Add Datacenter Locations for Cross Region Deployments and Scenario: Apply a Location to a Compute Resource for Cross Region Deployments .
Reservation policy	Apply a reservation policy to a blueprint to restrict the machines provisioned from that blueprint to a subset of available reservations. Only the reservation policies that are applicable to the current tenant are available. For information about creating reservation policies, see Configure a Reservation Policy.
Machine prefix	Machine prefixes are used to name provisioned machines. If you select **Use group default**, machines are named based on the default machine prefix for your business group. If you do not specify a prefix, one is generated for you based your business group name. Only the machine prefixes that are applicable to the current tenant are available. If your fabric administrator configures other machine prefixes for you to select, you can apply one prefix to all machines provisioned from your blueprint, no matter who the requestor is. For information about creating machine prefixes, see Configure Machine Prefixes.
Instances: Minimum and **Maximum**	Configure the maximum and minimum number of instances users can request for a deployment or for a scale in or scale out action. Entering the same value in the **Minimum** and **Maximum** fields configures exactly how many instances to provision. XaaS components are not scalable and are not updated during a scale operation. If you are using XaaS components in your blueprint, you might create a resource action for users to run after a scale operation, which might either scale or update your XaaS components as required. You can deactivate scaling by configuring the number of instances to allow for each machine component.

Build Information Tab

Configure build information settings for an OpenStack machine component.

Table 5-25. **Build Information** Tab

Setting	Description
Blueprint type	For record-keeping and licensing purposes, select whether machines provisioned from this blueprint are classified as Desktop or Server.
Provisioning workflow	The following provisioning workflows are available for an OpenStack machine component: ■ **CloudLinuxKickstartWorkflow** Provision a machine by booting from an ISO image, using a kickstart or autoYaSt configuration file and a Linux distribution image to install the operating system on the machine. ■ **CloudProvisioningWorkflow** Create a machine by starting from either a virtual machine instance or cloud-based image. ■ **CloudWIMImageWorkflow** Provision a machine by booting into a WinPE environment and installing an operating system using a Windows Imaging File Format (WIM) image of an existing Windows reference machine. When using a WIM provisioning workflow in a blueprint, specify a storage value that accounts for the size of each disk to be used on the machine. Use the total value of all disks as the minimum storage value for the machine component. Also specify a size for each disk that is large enough to accommodate the operating system.
OpenStack image	Select an available OpenStack image. An OpenStack image is a template that contains a software configuration, including an operating system. The images are managed by OpenStack accounts. You can refine the list of OpenStack image names in the display by using the **Filters** option in the **Names** column drop-down menu.

Table 5-25. **Build Information** Tab (continued)

Setting	Description
Key pair	Key pairs are optional for provisioning with OpenStack. Key pairs are used to provision and connect to a cloud instance. They are also used to decrypt Windows passwords and to log in to a Linux machine. The following key pair options are available: ■ Not specified Controls key pair behavior at the blueprint level rather than at the reservation level. ■ Auto-generated per business group Specifies that each machine provisioned in the same business group has the same key pair, including machines provisioned on other reservations when the machine has the same compute resource and business group. Because the key pairs are associated with a business group, the key pairs are deleted when the business group is deleted. ■ Auto-generated per machine Specifies that each machine has a unique key pair. The auto-generated per machine option is the most secure method because no key pairs are shared among machines.
Flavors	Select one or more OpenStack flavors. An OpenStack flavor is a virtual hardware template that defines the machine resource specifications for instances provisioned in OpenStack. Flavors are managed within the OpenStack provider and are imported during data collection.

Machine Resources Tab

Specify CPU, memory and storage settings for your OpenStack machine component.

Table 5-26. **Machine Resources** Tab

Setting	Description
CPUs: Minimum and **Maximum**	Enter a minimum and maximum number of CPUs that can be used by provisioned machines.
Memory (MB): Minimum and **Maximum**	Enter the minimum and maximum amount of memory that can be used by provisioned machines.
Storage (GB): Minimum and **Maximum**	Enter a minimum and maximum amount of storage that can be used by provisioned machines. When using a WIM provisioning workflow in a blueprint, specify a storage value that accounts for the size of each disk to be used on the machine. Use the total value of all disks as the minimum storage value for the machine component. Also specify a size for each disk that is large enough to accommodate the operating system.

Properties Tab

Optionally specify custom property and property group information for your OpenStack machine component.

You can add individual and groups of custom properties to the machine component by using the **Properties** tab. You can add also custom properties and property groups to the overall blueprint by using the **Properties** tab when you create or edit a blueprint by using the **Blueprint Properties** page.

You can use the **Custom Properties** tab to add and configure options for existing custom properties. Custom properties are supplied with vRealize Automation and you can also create property definitions.

Table 5-27. **Properties** > **Custom Properties** Tab Settings

Setting	Description
Name	Enter the name of a custom property or select an available custom property from the drop-down menu. Properties only appear in the drop-down menu if your tenant administrator or fabric administrator created property definitions.
Value	Enter or edit a value to associate with the custom property name. For example, set the value as `true` to allow entitled users to connect to VMs by using SSH.
Encrypted	You can choose to encrypt the property value, for example if the value is a password.
Overridable	You can specify that the property value can be overridden by the next or subsequent person who uses the property. If you select **Show in request**, users can edit property values when they request catalog items.
Show in Request	You can display the property name and value to users when they request machine provisioning. Select the overridable option if you want users to provide a value.

You can use the **Property Groups** tab to add and configure settings for existing custom property groups. You can create your own property groups or use property groups that have been created for you.

Table 5-28. **Properties** > **Property Groups** Tab Settings

Setting	Description
Name	Select an available property group from the drop-down menu.
Move Up and **Move Down**	Control the precedence level of property groups in descending order. The first-listed property group has precedence over the next-listed property group and so on.

Table 5-28. **Properties > Property Groups** Tab Settings (continued)

Setting	Description
View Properties	Display the custom properties in the selected property group.
View Merged Properties	Display custom properties in the order they appear in the list of property groups. Where the same property appears in more than one group, the property appears once in the list based on when it is first encountered.

Using Network Custom Properties

You can specify network and security information for machine components other than vSphere and blueprints that do not contain NSX by using network custom properties at either the blueprint or machine component level.

The **Network & Security** components are only available for use with vSphere machine components. Non-vSphere machine components do not contain a **Network** or **Security** tab.

For vSphere machine components with associated NSX, use network, security, and load balancing setting in the user interface. For machine components that do not have a **Network** or **Security** tab, you can add network and security custom properties, such as `VirtualMachine.Network0.Name`, to their **Properties** tab in the design canvas. NSX network, security, and load balancer properties are only applicable to vSphere machines.

You can define custom properties individually or as part of an existing property group by using the **Properties** tab when configuring a machine component in the design canvas. The custom properties that you define for a machine component pertain to machines of that type that are provisioned from the blueprint.

For information about the available custom properties, see Custom Properties Grouped by Function and Custom Properties Grouped by Name.

Troubleshooting Blueprints for Clone and Linked Clone

When creating a linked clone or clone blueprint, machine or templates are missing. Using your shared clone blueprint to request machines fails to provision machines.

Problem

When working with clone or linked clone blueprints, you might encounter one of the following problems:

- When you create a linked clone blueprint, no machines appear in the list to clone, or the machine you want to clone does not appear.
- When you create a clone blueprint, no templates appear in the list of templates to clone, or the template you want does not appear.
- When machines are requested by using your shared clone blueprint, provisioning fails.

- Because of data collection timing, a template that has been removed is still visible to users as they create or edit linked clone blueprints.

Be aware that linked clones are not supported when provisioning to SDRS. Linked clones would be created on the same datastore as the parent, but not be rebalanced across the cluster datastores. In such cases, the parent datastore might eventually fill up.

Cause

There are multiple possible causes for common clone and linked clone blueprint problems.

For related information about the **Clone from** and **Clone from snapshot** with **Use current snapshot** options that are available when you create blueprints, see vSphere Machine Component Settings in vRealize Automation.

Table 5-29. Causes for Common Clone and Linked Clone Blueprints Problems

Problem	Cause	Solution
Machines missing	You can only create linked clone blueprints by using machines you manage as a tenant administrator or business group manager.	A user in your tenant or business group must request a vSphere machine. If you have the appropriate roles, you can do this yourself. You can also see unmanaged machines in this dialog. Managed machines may have been imported. There is no requirement that machines be provisioned from vRealize Automation to be visible in this dialog.
Templates missing	Data collection has failed on a given endpoint or no endpoints are available for the component's platform.	■ If your endpoints are clustered and contain multiple compute resources, verify that your IaaS administrator added the cluster containing the templates to your fabric group. ■ For new templates, verify that IT placed the templates on the same cluster included in your fabric group.
Provisioning failure with a shared blueprint	For blueprints, no validation is available to ensure that the template you select exists in the reservation used to provision a machine from your shared clone blueprint.	Consider using entitlements to restrict the blueprint to users who have a reservation on the compute resource where the template exists.

Table 5-29. Causes for Common Clone and Linked Clone Blueprints Problems (continued)

Problem	Cause	Solution
Provisioning failure with a guest agent	The virtual machine might be rebooting immediately after the guest operating system customization is completed, but before the guest agent work items are completed, causing provisioning to fail. You can use the custom property `VirtualMachine.Admin.CustomizeGuestOSDelay` to increase the time delay.	Verify that you have added the custom property `VirtualMachine.Admin.CustomizeGuestOSDelay`. The value must be in HH:MM:SS format. If the value is not set, the default value is one minute (00:01:00).
Clone or linked clone blueprint provisioning fails because the template on which the clone is based cannot be found	It is not possible to provision machines from a blueprint that is cloned from a template that no longer exists. vRealize Automation runs data collection periodically, default every 24 hours. If a template is removed, the change is not reflected until the next data collection and so it is possible to create a blueprint based on a non-existent template.	Redefine the blueprint using an existing template and then request provisioning. As a precaution and as applicable, you can run data collection before defining the clone or linked clone blueprint.

Designing Blueprints with NSX Settings

If you configured vRealize Automation integration with NSX for vSphere or NSX-T, you can use network, security, and load balancer components configure your blueprint for machine provisioning.

You can also add the following NSX network and security settings to the overall blueprint:

- Transport zone

  Contains the networks that are used for the provisioned machine deployment.

- Network reservation policy

  Manages network communication for the provisioned machine deployment.

- App isolation

  Allows only internal traffic between the machines that are used in the provisioned machine deployment.

For more information about vRealize Automation and NSX integration, see this vRA and NSX - Intro to Network and Security Automation blog article and preview content for the Networking and Security with vRealize Automation and NSX course series.

NSX settings are only applicable to vSphere machine component types.

New Blueprint and Blueprint Properties Page Settings with NSX in vRealize Automation

You can specify settings that apply to the entire vRealize Automation blueprint, including some NSX settings, by using the **New Blueprint** page when you create the blueprint. After you create the blueprint, you can edit these settings on the **Blueprint Properties** page.

General Tab

Settings on the General tab apply to the overall vRealize Automation blueprint.

Table 5-30. **General** Tab Settings

Setting	Description
Name	Enter a name for your blueprint.
Identifier	The identifier field automatically populates based on the name you entered. You can edit this field now, but after you save the blueprint you can never change it. Identifiers are permanent and unique within your tenant. You can use them to programmatically interact with blueprints and to create property bindings.
Description	Summarize your blueprint for the benefit of other architects. This description also appears to users on the request form.
Deployment limit	Specify the maximum number of deployments that can be created when this blueprint is used to provision machines.
Lease days: Minimum and **Maximum**	Enter a minimum and maximum value to allow users to choose from within a range of lease lengths. When the lease ends, the deployment is either destroyed or archived. If you do not specify a minimum or maximum value, the lease is set to never expire. Enter the lease information for your machines in your vRealize Automation blueprint, not in the source endpoint application. If you specify the lease information in an external application, it is not recognized in vRealize Automation.
Archive days	You can specify an archival period to temporarily retain deployments instead of destroying deployments as soon as their lease expires. Specify 0 to destroy the deployment when its lease expires. The archive period begins on the day the lease expires. When the archive period ends, the deployment is destroyed. The default is 0.
Propagate updates to existing deployments	Broadened minimum-maximum ranges for CPU, memory, or storage are pushed to active deployments that were provisioned from the blueprint. The new range must fully encompass the old range. For example, for an original minimum 32 and maximum 128 (32, 128), a change such as (16, 128) or (32, 256) or (2, 1000) can take effect upon reconfiguration or scale-out, but a change of (33, 512) or (4, 64) cannot. The changes take effect upon the next reconfigure or scale-out action. For more information, see Action Menu Commands for Provisioned Resources.

NSX Settings Tab

If you configured NSX, you can specify NSX transport zone, network reservation policy, and app isolation settings when you create or edit a blueprint. These settings are available on the **NSX Settings** tab on the **Blueprint** and **Blueprint Properties** pages.

For information about your NSX application, see VMware NSX Data Center for vSphere Documentation or VMware NSX-T Data Center Documentation.

Table 5-31. NSX Settings Tab Settings

Setting	Description
Transport zone	Select an existing NSX transport zone to contain the network or networks that the provisioned machine deployment can use. A transport zone defines which clusters the networks can span. When provisioning machines, if a transport zone is specified in a reservation and in a blueprint, the transport zone values must match. Only the transport zones that are applicable to the current tenant are available. A transport zone is required for blueprints that containNSX for vSphere or NSX-T on-demand network and security objects. For more information, see Applying an NSX Transport Zone to a Blueprint. Specify a transport zone that is appropriate for an NSX for vSphere or NSX-T deployment.
Network reservation policy	For NSX for vSphere, select a networking reservation policy to help determine where to place the edge or DLR in the deployment. When vRealize Automation provisions a machine with NAT or routed networking, it provisions a routed gateway as the network router. The Edge or routed gateway is a management machine that consumes compute resources. It also manages the network communications all machine in that deployment. The reservation used to provision the Edge or routed gateway determines the external network used for NAT and load balancer virtual IP addresses. As a best practice, use separate management clusters for management machines such as NSX Edges. For NSX-T, select a networking reservation policy to help determine where to place the tier 0 logical router in the blueprint deployment. For more information, see Applying an NSX Networking Reservation Policy to a Blueprint. Specify a reservation policy that is appropriate for an NSX for vSphere or NSX-T deployment. Clusters deployed by the blueprint can be managed by NSX for vSphere or NSX-T.
App isolation	Select the **App isolation** check box to use the app isolation security policy configured in NSX for vSphere. The app isolation policy is applied to all the vSphere machine components in the blueprint. You can add security groups and tags to allow vRealize Orchestrator to open the isolated network to allow additional paths in and out of the app isolation. For more information, see Applying NSX App Isolation to a Blueprint.

Properties Tab

Custom properties that you add at the blueprint level apply to the entire blueprint, including all components. However, they can be overridden by other custom properties. For information about the order of precedence for custom properties, see Understanding Custom Properties Precedence.

Table 5-32. **Properties** Tab Settings

Tab	Setting	Description
Property Groups	Property groups are reusable groups of properties that simplify the process of adding custom properties to blueprints.	
	Add	Add one or more existing property groups and apply them to the overall blueprint. The following Containers-related property groups are supplied: ■ Container host properties with certificate authentication ■ Container host properties with user/ password authentication
	Move up /Move down	Control the order of precedence given to each property group in relation to one another by prioritizing the groups. The first group in the list has the highest priority, and its custom properties have first precedence. You can also slide to reorder.
	View properties	View the custom properties in the selected property group.
	View merged properties	If a custom property is included in more than one property group, the value included in the property group with the highest priority takes precedence.
Custom Properties	You can add individual custom properties instead of property groups.	
	New	Add an individual custom property and apply it to the overall blueprint.
	Name	Enter the property name. For a list of custom property names and descriptions, see Chapter 8 Custom Properties and the Property Dictionary .
	Value	Enter the value for the custom property.
	Encrypted	Encrypt the property value, for example, if the value is a password.
	Overridable	The blueprint user can override the property value. If you select **Show in request**, users can see and edit property values when they request catalog items.
	Show in request	The property name and value is visible to users on the provisioning request form. Select **Overridable** if allow users to provide a value.

Applying an NSX Transport Zone to a Blueprint

An NSX administrator can create transport zones to control cluster use of networks.

A transport zone controls which hosts a logical switch can reach. It can span one or more host clusters, including hosts across multiple vCenters.

For blueprints that contain an on-demand NAT or on-demand routed network, specify a transport zone that contains the networks to be used by the provisioned machine deployment.

For blueprints that include an NSX-T endpoint, you must specify a transport zone.

The transport zone that you specify for the blueprint must match the transport zone that you specify for the reservation used by the blueprint. See Applying an NSX Networking Reservation Policy to a Blueprint.

- If your blueprint does not use NSX-T on-demand components, the transport zone value is ignored.
- NSX-T supports multiple overlay transport zones and multiple VLAN transport zones.
- A transport zone is required to create a logical switch. Logical switches are created within transport zones.
- Only the transport zones for the current tenant are exposed when authoring a blueprint. Transport zones are made available if they are used by a reservation in the current tenant.

Applying an NSX Networking Reservation Policy to a Blueprint

When provisioning the blueprint, the reservation policy is used to group the reservations that can be considered for the deployment. Networking information is contained in each reservation.

If there is a transport zone in this reservation policy, it must match the transport zone specified in the blueprint. See Applying an NSX Transport Zone to a Blueprint.

You can apply a network reservation policy at the blueprint level by using the **New Blueprint** or **Blueprint Properties** page.

NSX for vSphere Considerations

For NSX for vSphere, this reservation policy helps determine the placement of the NSX edge or selection of the Distributed Logical Router (DLR) associated to the on-demand networks. This is also referred to as a routed gateway reservation policy or an edge reservation policy.

For example, for NSX for vSphere, a NAT network profile and load balancer enable vRealize Automation to deploy an NSX edge services gateway. A routed network profile uses an NSX for vSphere logical distributed router (DLR). The DLR must be created in NSX before it can be consumed by vRealize Automation. vRealize Automation cannot create DLRs. After data collection, vRealize Automation can use the DLR for virtual machine provisioning.

An NSX edge provides routing services and connectivity to networks that are external to the NSX deployment. The NSX Edge gateway connects isolated, subnets to shared (uplink) networks by providing common gateway services such as NAT, and dynamic routing. Common deployments of NSX edge include multi-tenant environments where the NSX edge creates virtual boundaries for each tenant.

vRealize Automation provisions a routed gateway, for example an edge services gateway, for NAT networks and for load balancers. For routed networks, vRealize Automation uses existing distributed routers.

The reservation used to provision the edge or routed gateway determines the available NAT, private, or routed network profiles and the load balancer virtual IP addresses.

NSX-T Considerations

For NSX-T, this reservation policy helps select a Tier-0 logical router used for the deployment.

Tier-0 logical routers have downlink ports to connect to a Tier-1 logical routers and uplink ports to connect to external networks. vRA connects a Tier-1 logical router to a Tier-0 logical router for northbound physical router access and assigns an edge cluster to a logical router to perform NAT and load balancer services.

Applying NSX App Isolation to a Blueprint

You can enable app isolation to only allow internal traffic between the components provisioned by the blueprint.

An NSX app isolation policy acts as a firewall to block all inbound and outbound traffic to and from the provisioned machines in the deployment. When you specify a defined NSX app isolation policy, the machines provisioned by the blueprint can communicate with each other but cannot connect outside the firewall.

When an app isolation rule is specified, and security rules are also specified by using security groups in the blueprint, the app isolation setting is the last rule processed during blueprint deployment.

You can apply app isolation at the blueprint level by using the **New Blueprint** or **Blueprint Properties** page.

Considerations for NSX for vSphere

The provisioned components are placed in a security group, which is isolated using firewall rules. Enablement requires that the vSphere endpoint is configured to support NSX app isolation.

When using an NSX for vSphere app isolation policy, only internal traffic between the machines provisioned by the blueprint is allowed. When you request provisioning, a security group is created for the machines to be provisioned. An app isolation policy is created in NSX for vSphere and applied to the security group. Firewall rules are defined in the security policy to allow only internal traffic between the components in the deployment.

When provisioning with a blueprint that uses both an NSX for vSphere edge load balancer and an NSX for vSphere app isolation security policy, the dynamically provisioned load balancer is not added to the security group. This prevents the load balancer from communicating with the machines for which it is meant to handle connections. Because edges are excluded from the NSX for vSphere distributed firewall, they cannot be added to security groups. To allow load balancing to function properly, use another security group or security policy that allows the required traffic into the component VMs for load balancing.

The app isolation policy has a lower precedence compared to other security policies in NSX for vSphere. For example, if the provisioned deployment contains a web component machine and an app component machine and the web component machine hosts a web service, then the service must allow inbound traffic on ports 80 and 443. In this case, users must create a web security policy in NSX for vSphere with firewall rules defined to allow incoming traffic to these ports. In vRealize Automation, users must apply the web security policy on the web component of the provisioned machine deployment.

Note If a blueprint contains a load balancers, and app isolation is enabled, load balancer VIPs are added to the app isolation security group as an IPSet. If a blueprint contains an on-demand security group that is associated to a machine tier that is associated to a load balancer, the on-demand security group includes the machine tier, IPSet, and VIPs.

If the web component machine needs access to the app component machine using a load balancer on ports 8080 and 8443, the web security policy should also include firewall rules to allow outbound traffic to these ports in addition to the existing firewall rules that allow inbound traffic to ports 80 and 443.

Considerations for NSX-T

The provisioned components are placed in an NS Group, which is isolated using firewall rules. Enablement requires that the vSphere endpoint is configured to support NSX app isolation.

NSX-T supports creating a two-tier logical router topology: the top-tier logical router is Tier-0 and the bottom-tier logical router is Tier-1. This structure gives both provider administrator and tenant administrators complete control over their services and policies. In NSX-T administrators control and configure Tier-0 routing and services, and tenant administrators control and configure Tier-1.

Configuring Network and Security Component Settings in vRealize Automation

vRealize Automation supports virtualized networks based on the NSX platform. Integrated Containers for vRealize Automation networks are also supported.

To integrate NSX network and security with vRealize Automation, an IaaS administrator must configure vSphere and NSX endpoints. vRealize Automation supports NSX for vSphere and NSX-T.

For information about external preparation, see Checklist for Preparing NSX Network and Security Configuration.

You can create network profiles that specify network settings in reservations and in the blueprint. External network profiles define existing physical networks. On-demand NAT and routed network profiles can build NSX logical switches and appropriate routing settings for a new network path.

The network and security component settings that you add to the blueprint are derived from your NSX for vSphere and NSX-T configuration. For information about configuring NSX, see the *Administration Guide* in NSX for vSphere product documentation or NSX-T product documentation, depending on which application you are using.

For vSphere machine components with associated NSX, use network, security, and load balancing setting in the user interface. For machine components that do not have a **Network** or **Security** tab, you can add network and security custom properties, such as `VirtualMachine.Network0.Name`, to their **Properties** tab in the design canvas. NSX network, security, and load balancer properties are only applicable to vSphere machines.

If you specify a network profile in a reservation and a blueprint, the blueprint values take precedence.

Depending on the compute resource, you can select a transport zone that identifies a vSphere endpoint. A transport zone specifies the hosts and clusters that can be associated with logical switches created within the zone. A transport zone can span multiple vSphere clusters. The blueprint and the reservations used in the provisioning must have the same transport zone setting. Transport zones are defined in the NSX environments.

You can configure security settings by specifying information in a reservation, blueprint, or guest agent script. If machines require a guest agent, add a security rule to the reservation or the blueprint.

You can also add a Containers network component to a blueprint.

For related information about configuring networking and security for NSX-T in vRealize Automation, see VMware blog Application Networking and Security with vRealize Automation and NSX-T.

Controlling Tenant Access for Security Objects in vRealize Automation

You can control the cross-tenancy availability of NSX security objects in vRealize Automation.

When you create an NSX security object, its default availability can be either global, meaning available in all tenants for which the associated endpoint has a reservation, or hidden to all users except the administrator.

The availability of security objects across tenants depends on whether the associated endpoint has a reservation or reservation policy in the tenant.

NSX does not tenant security groups. However, you can control security group availability in vRealize Automation by using the `VMware.Endpoint.NSX.HideDiscoveredSecurityObjects` custom property.

By default, new security objects are available to all tenants for the associated NSX endpoints in which you have a reservation. If the endpoint does not have a reservation in the active tenant, the security objects are not available in the active tenant.

If you have not set the `VMware.Endpoint.NSX.HideDiscoveredSecurityObjects` custom property on NSX endpoints, new security objects are set to global by default. Security objects that existed prior to upgrading to this release of vRealize Automation are set to global regardless of the custom property.

Note When you upgrade to this vRealize Automation release, security groups from the previous release are set to global by default. Existing security groups and security tags are available in all tenants in which the associated endpoint has a reservation.

You can hide new security groups by default by adding the `VMware.Endpoint.NSX.HideDiscoveredSecurityObjects` custom property to the associated NSX endpoint. This setting takes effect the next time the NSX endpoint is data-collected and applied only to new security objects.

For more information about the `VMware.Endpoint.NSX.HideDiscoveredSecurityObjects` custom property, see Custom Properties for Networking and Security.

You can also change the tenancy setting of an existing security object programmatically. For example, if a security group is set to global, you can change the tenant availability of a security object by using the associated NSX endpoint's Tenant ID setting in the vRealize Automation REST API or vRealize CloudClient. The available Tenant ID settings for the NSX endpoint are as follows:

- "`<global>`" - the security object is available to all tenants. This is the default setting for existing security objects after upgrade to this release and for all new security objects that you create.
- "`<unscoped>`" - the security object is not available to any tenants. Only the system administrator can access the security object. This is an ideal setting when defining security objects that are to eventually be assigned to a specific tenant.
- "*`tenant_id_name`*" - the security object is only available to a single, named tenant.

You can use the vRealize Automation REST API or vRealize CloudClient tools to assign the Tenant ID parameter (*tenantId*) of security objects that are associated to a specific endpoint to a named tenant.

For information about vRealize Automation REST API commands, see *vRealize Automation API Reference* in the vRealize Automation API Documentation section for your vRealize Automation 7.x release. For additional information, see *vRealize Automation Programming Guide* in the vRealize Automation API Documentation section for your vRealize Automation 7.x release.

For information about vRealize CloudClient, see https://code.vmware.com/web/dp/tool/cloudclient.

Understanding NSX-T Deployment Topologies for Networking, Security, and Load Balancer Configurations

You can establish and use various deployment topologies based on how you configure your NSX-T network and security and your load balancer components in the vRealize Automation blueprint.

Networking and Security

- Routed networks

  If you attach an NSX-T routed network component to a vSphere machine component in the blueprint, the following topology is provisioned in NSX-T:

  - A Tier-1 router is created.
  - A logical switch is created.
  - The Tier-1 router is down-linked to the logical switch.

  - Specific routed routes are advertised on the Tier 1 router.
- NAT networks (static IP)

  If you attach an NSX-T NAT network to a vSphere machine component in the blueprint, the following topology is provisioned in NSX-T:

  - A Tier-1 router is created.
  - A logical switch is created.
  - The Tier-1 router is connected to the edge cluster.
  - The Tier-1 router is up-linked to a Tier-0 router; the Tier-0 router is selected from the reservation.
  - The Tier-1 router is down-linked to the logical switch.
  - All NAT routes are advertised on the Tier 1 router.
  - One external IP is allocated for each NAT network from the external network profile that supports the on-demand NAT network profile. This IP is used for SNAT and DNAT rules.
- NAT networks (DHCP)

  If you attach an NSX-T NAT network with DHCP to a vSphere machine component in the blueprint, the following topology is provisioned in NSX-T:

  - A Tier-1 router is created.
  - A logical switch is created.
  - The Tier-1 router is connected to the edge cluster.
  - The Tier-1 router is up-linked to a Tier-0 router; the Tier-0 router is selected from the reservation.
  - The Tier-1 router is down-linked to the logical switch.
  - A DHCP server with an IP pool is provisioned.
  - All NAT routes are advertised on the Tier 1 router.
- App Isolation

  If app isolation is required for a blueprint with NSX-T components, the following topology is provisioned in NSX-T:

  Note You configure app isolation for the blueprint on the Blueprint Properties page when you create or edit the blueprint.

  - An NS Group is created.
  - A firewall section, with firewall isolation rules, is created.
  - The machines in the blueprint are added to the app isolation NS Group by using tags.

  - The load balancer VIP and external IP for NAT networks at the IPset are added to the app isolation NS Group.

  To support app isolation NS Groups, you must connect the machines to opaque networks.

- Existing NS Groups

  If you attach an existing NS Group component to a vSphere machine component in the blueprint, the following topology is provisioned in NSX-T:

  - The machines that are attached to the NS Group are added to the NS Group in NSX-T using tags as a membership criteria

  To support existing NS Groups, you must connect the machines to opaque networks.

Load Balancers

The following topologies are supported for load balancers in an NSX-T blueprint deployment:

- One-arm on a NAT on-demand network.
- One-arm on a routed on-demand network.
- One-arm on external (existing) network.
- Two-arm, one on NAT and one on external.
- Two-arm, one on routed and one on external.

If an NSX-T load balancer is added to the blueprint, the following topology - in addition to the network topologies, is provisioned in the deployment:

- For all topologies except where the load balancer is one-armed on an external network:
  - A single load balancer service is created, even if there are multiple load balancer components in the blueprint.
  - The load balancer service is attached to the Tier-1 router, for the deployment. The Tier-1 router is created on-demand.
- For topologies where the load balancer is one-armed on an external network:
  - The external network that is specified in the reservation must be a VC-opaque network (NSX-T logical switch).
  - The Tier-1 router must exist and be attached to the external network (NSX-T logical switch).
  - If the Tier 1 router does not already exist, the load balancer server is created on-demand and attached to the Tier-1 router; otherwise, an already existing load balancer is used.
- The VIP route is advertised, unless the VIP is on a private NAT network.
- One or more virtual servers are created on the load balancer service.

  There are limitations on the number of virtual servers per load balancer service based on the size of the load balancer.

- A virtual server application profile is created for each virtual server.
- A virtual server persistence profile is created for each virtual server that has configured persistence options.
- A membership pool is configured that contains the static IP of each machine in the membership pool.
- A single load balancer service is created regardless of the number of load balancer components in the blueprint.
- A health monitor is created and configured for each member pool.

For virtual servers with HTTPS support, and unlike load balancers in NSX for vSphere, there's no support for SSL passthrough in NSX-T load balancers. vRealize Automation configures the load balancer virtual server to terminate SSL at the load balancer and to use plain HTTP from the load balancer to the pool members. The certificate name and SSL client profile name, bot of which must exist in NSX-T, must be specified when configuring virtual server with HTTPS. You can import certificates into the NSX-T trust manager.

When more than one NSX-T component is present on the blueprint, the Tier-1 logical router is shared among all the components and is configured accordingly. The external Tier 1 logical router ID is displayed in the Details view for each component on the vRealize Automation Deployments page.

Using NSX for vSphere Network Components in a vRealize Automation Blueprint

You can add one or more NSX for vSphere network components to the design canvas and configure their settings for vSphere machine components in a vRealize Automation blueprint.

The network and security component settings that you add to the blueprint are derived from your NSX for vSphere configuration. For information about configuring NSX for vSphere, see the *NSX Administration Guide* in NSX for vSphere product documentation.

Add an Existing Network Component for NSX for vSphere

You can add an existing NSX for vSphere network component to the design canvas to associate its settings to one or more vSphere machine components in the blueprint.

You can use an existing network component to add an NSX for vSphere network to the design canvas and configure its settings for use with vSphere machine components and Software or XaaS components that pertain to vSphere.

When you associate an existing network component or an on-demand network component with a machine component, the NIC information is stored with the machine component. The network profile information that you specify is stored with the network component.

You can add multiple network and security components to the design canvas.

For vSphere machine components with associated NSX, use network, security, and load balancing setting in the user interface. For machine components that do not have a **Network** or **Security** tab, you can add network and security custom properties, such as `VirtualMachine.Network0.Name`, to their **Properties** tab in the design canvas. NSX network, security, and load balancer properties are only applicable to vSphere machines.

Only the network profiles that are applicable to the current tenant are exposed when authoring a blueprint. Specifically, network profiles are made available if there is at least one reservation in the current tenant that has at least one network assigned to the profile.

Prerequisites

- Create and configure network settings for NSX. See Checklist for Preparing NSX Network and Security Configuration and the *NSX for vSphere Administration Guide* in NSX for vSphere product documentation.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Create a network profile. See Creating a Network Profile in vRealize Automation.
- Log in to vRealize Automation as an **infrastructure architect**.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.

Procedure

1. To display the list of available network and security components, click **Network & Security** in the Categories section.
2. Drag an **Existing Network** component onto the design canvas.
3. Click in the **Existing network** text box and select an existing network profile.

   The description, subnet mask, and gateway values are populated based on the selected network profile.
4. (Optional) Click the **DNS/WINS** tab.
5. (Optional) Specify DNS and WINS settings for the network profile.
   - Primary DNS
   - Secondary DNS
   - DNS Suffix
   - Preferred WINS
   - Alternate WINS

   You cannot change the DNS or WINS settings for an existing network.
6. (Optional) Click the **IP Ranges** tab.

   The IP range or ranges specified in the network profile are displayed. You can change the sort order or column display. For NAT networks, you can also change IP range values.
7. To save the blueprint as draft or continue configuring the blueprint, click **Save** or **Finish**.

What to do next

You can add network settings in the **Network** tab of a vSphere machine component.

Add a Private Network Component for NSX for vSphere in vRealize Automation

You can add a private NSX for vSphere network component to the design canvas to associate its settings to one or more vSphere machine components in the vRealize Automation blueprint.

Only the network profiles that are applicable to the current tenant are exposed when you author a blueprint.

This private network option is only available for NSX for vSphere. It is not available for NSX-T.

Prerequisites

- Create and configure network settings for NSX. See Checklist for Preparing NSX Network and Security Configuration and the *NSX for vSphere Administration Guide* in NSX for vSphere product documentation.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Create a network profile. See Creating a Network Profile in vRealize Automation.
- Log in to vRealize Automation as an **infrastructure architect**.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.

Procedure

1. To display the list of available network and security components, click **Network & Security** in the Categories section.
2. Drag an On-Demand Private network component onto the design canvas.
3. To uniquely label the component in the design canvas, enter a component name in the **ID** text box.
4. Select an appropriate existing network profile from the **Parent network profile** drop-down menu.
5. (Optional) Enter a component description in the **Description** text box.
6. (Optional) Click the **DNS/WINS** tab.
7. (Optional) Specify DNS and WINS settings for the network profile.
   - Primary DNS
   - Secondary DNS
   - DNS Suffix
   - Preferred WINS
   - Alternate WINS

   You cannot change the DNS or WINS settings for an existing network.

8 Click the **IP Ranges** tab.

   a Enter a start IP address value in the **IP range start** text box.

   b Enter a start IP address value in the **IP range start** text box.

9 To save the blueprint as draft or continue configuring the blueprint, click **Save** or **Finish**.

Creating and Using NAT Rules for NSX for vSphere

You can add NAT rules to a one-to-many NAT network component in a blueprint when the NAT network component is associated to a non-clustered vSphere machine component or an on-demand NSX for vSphere load balancer component.

You can define NAT rules for any NSX for vSphere-supported protocol. You can map a port or a port range from the external IP address of an Edge to a private IP address in the NAT network component.

- vSphere Machine Component

  You can create NAT rules for a NAT one-to many network component that is associated to a non-clustered vSphere machine component.

  For example, if two machines are associated to a NAT one-to-many network component on the blueprint, you can define a NAT rule that allows port 443 on the external IP to connect to the machines through port 80 on the NAT network using TCP protocol.

- NSX for vSphere Load Balancer Component

  You can create NAT rules for a NAT one-to many network component that is associated to the VIP network of an NSX for vSphere load balancer component.

  For example, if the NAT network component is associated to a load balancer component that is load balancing three machines, you can define a NAT rule that allows port 90 on the external IP to connect to the load balancer VIP through port 80 on the NAT network using UDP protocol.

You can create any number of NAT rules and you can control the order in which the rules are processed.

The following elements are not supported for NAT rules:

- NICs that are not in the current network
- NICs that are configured to get IP addresses by using DHCP
- Machine clusters

To add NAT rules to a NAT network component in a blueprint, see Add an On-Demand NAT or On-Demand Routed Network Component in vRealize Automation.

For related information about using NAT rules, see public articles such as this vmwarelab blog post.

Add an On-Demand NAT or On-Demand Routed Network Component in vRealize Automation

You can add an NSX for vSphere on-demand NAT network component or NSX for vSphere on-demand routed network component to the design canvas in preparation for associating their settings to one or more vSphere machine components in the vRealize Automation blueprint.

When you associate an existing network component or an on-demand network component with a machine component, the NIC information is stored with the machine component. The network profile information that you specify is stored with the network component.

You can add multiple network and security components to the design canvas.

You can have more than one on-demand network component in a single blueprint. However, all of the on-demand network profiles that are used in the blueprint must reference the same external network profile.

For vSphere machine components with associated NSX, use network, security, and load balancing setting in the user interface. For machine components that do not have a **Network** or **Security** tab, you can add network and security custom properties, such as `VirtualMachine.Network0.Name`, to their **Properties** tab in the design canvas. NSX network, security, and load balancer properties are only applicable to vSphere machines.

Only the network profiles that are applicable to the current tenant are exposed when authoring a blueprint. Specifically, network profiles are made available if there is at least one reservation in the current tenant that has at least one network assigned to the profile.

Prerequisites

- Create and configure network settings for NSX for vSphere externally. See Checklist for Preparing NSX Network and Security Configuration and *NSX Administration Guide* at NSX for vSphere product documentation.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Create an on-demand network profile. See Creating a Network Profile in vRealize Automation.

  For example, if you are adding an on-demand NAT network component see Creating a NAT Network Profile For an On-Demand Network.
- Log in to vRealize Automation as an **infrastructure architect**.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.
- If you want to specify NAT rules for a NAT network component, you must use a NAT one-to-many network profile. See Create a NAT Network Profile By Using the Supplied IPAM Endpoint or Create a NAT Network Profile By Using a Third-Party IPAM Endpoint in vRealize Automation. For information about NAT rules, see Creating and Using NAT Rules for NSX for vSphere.

Procedure

1 To display the list of available network and security components, click **Network & Security** in the Categories section.

2 Drag an On-Demand NAT or On-Demand Routed network component onto the design canvas.

3 To uniquely label the component in the design canvas, enter a component name in the **ID** text box.

4 Select an appropriate network profile from the **Parent network profile** drop-down menu. For example, if you want to add a NAT network component, select a NAT network profile that is configured to support your intended network settings.

   If you want to specify NAT rules in a NAT network component, you must use a parent network profile that is configured for NAT one-to-many.

   Depending on the profile type you select, the following network settings are populated based on your network profile selection. Changes to these values must be made in the network profile:

   - External network profile name
   - NAT type (On-Demand NAT)
   - Subnet mask
   - Range subnet mask (On-Demand Routed)
   - Range subnet mask (On-Demand Routed)
   - Base IP address (On-Demand Routed)

5 (Optional) Enter a component description in the **Description** text box.

6 (Optional) Click the **DNS/WINS** tab.

7 (Optional) Specify DNS and WINS settings for the network profile.

   - Primary DNS
   - Secondary DNS
   - DNS Suffix
   - Preferred WINS
   - Alternate WINS

   You cannot change the DNS or WINS settings for an existing network.

8 Click the **IP Ranges** tab.

   The IP range or ranges specified in the network profile are displayed. You can change the sort order or column display. For NAT networks, you can also change IP range values.

   a Enter a start IP address value in the **IP range start** text box.

   b Enter a start IP address value in the **IP range start** text box.

9 If you are using a NAT network that is based on a one-to-many NAT network profile that uses static IP ranges, you can use the **NAT Rules** tab to add rules that enable an external IP to access components in the internal NAT network.

   For a NAT one-to-many network, you can define NAT rules that can be configured when you add a NAT network component to the blueprint. You can change a NAT rule when you edit the NAT network in a deployment.

   The options that are available for selection are based on the vSphere machine or NSX for vSphere load balancer components that you have associated to the NAT network component.

   - **Name** - Enter a unique rule name.
   - **Component** - Select from a list of associated vSphere machine or load balancer components to which the NAT network is associated.

     NAT rules are only supported for non-clustered machines. If you have specified a cluster size of more than 1, no components are listed as the configuration is not supported.
   - **Source port** - Select the ANY option, enter a valid port or port range, or specify a valid property binding.
   - **Destination port** - Select the ANY option, enter a valid port or port range, or specify a valid property binding.
   - **Protocol** - Enter any valid NSX for vSphere-supported protocol or select the TCP, UDP, or ANY option.
   - **Description** - Enter a brief description of the NAT rule.

10 To save the blueprint as draft or continue configuring the blueprint, click **Save** or **Finish**.

What to do next

You can add network settings in the **Network** tab of a vSphere machine component.

Using NSX-T Network Components in a Blueprint

You can add one or more NSX-T network components to the design canvas and configure their settings for vSphere machine components in the blueprint.

The network and security component settings that you add to the blueprint are derived from your NSX-T configuration. For information about configuring NSX-T, see the *NSX-T Administration Guide* in NSX-T product documentation.

When you deploy a blueprint that contains an NSX-T endpoint, the deployment assigns a tag to NSX-T components in the deployment. The tag name and the deployment name match.

For more information about NSX-T-specific deployment and topology considerations, see Understanding NSX-T Deployment Topologies for Networking, Security, and Load Balancer Configurations.

Add an Existing Network Component for NSX-T

You can add an existing NSX-T network component to the design canvas to associate its settings to one or more vSphere machine components in the blueprint.

You can use an existing network component to add an NSX-T network to the design canvas and configure its settings for use with vSphere machine components and Software or XaaS components that pertain to vSphere.

When you associate an existing network component or an on-demand network component with a machine component, the NIC information is stored with the machine component. The network profile information that you specify is stored with the network component.

You can add multiple network and security components to the design canvas.

For vSphere machine components with associated NSX, use network, security, and load balancing setting in the user interface. For machine components that do not have a **Network** or **Security** tab, you can add network and security custom properties, such as `VirtualMachine.Network0.Name`, to their **Properties** tab in the design canvas. NSX network, security, and load balancer properties are only applicable to vSphere machines.

Only the network profiles that are applicable to the current tenant are exposed when authoring a blueprint. Specifically, network profiles are made available if there is at least one reservation in the current tenant that has at least one network assigned to the profile.

Prerequisites

- Create and configure network settings for NSX-T. See Checklist for Preparing NSX Network and Security Configuration and *NSX-T Administration Guide* at NSX-T product documentation.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Create a network profile. See Creating a Network Profile in vRealize Automation.
- Log in to vRealize Automation as an **infrastructure architect**.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.

Procedure

1. To display the list of available network and security components, click **Network & Security** in the Categories section.
2. Drag an **Existing Network** component onto the design canvas.
3. Click in the **Existing network** text box and select an existing network profile.

   The description, subnet mask, and gateway values are populated based on the selected network profile.
4. (Optional) Click the **DNS/WINS** tab.

5 (Optional) Specify DNS and WINS settings for the network profile.

   - Primary DNS
   - Secondary DNS
   - DNS Suffix
   - Preferred WINS
   - Alternate WINS

   You cannot change the DNS or WINS settings for an existing network.

6 (Optional) Click the **IP Ranges** tab.

   The IP range or ranges specified in the network profile are displayed. You can change the sort order or column display. For NAT networks, you can also change IP range values.

7 To save the blueprint as draft or continue configuring the blueprint, click **Save** or **Finish**.

What to do next

You can add network settings in the **Network** tab of a vSphere machine component.

Creating and Using NAT Rules for NSX-T
You can add NAT rules to a one-to-many NAT network component in a blueprint when the NAT network component is associated to a non-clustered vSphere machine component.

You can define NAT rules for any NSX-T-supported protocol. You can map a port or a port range from the external IP address of an Edge to a private IP address in the NAT network component.

You can create NAT rules for a NAT one-to many network component that is associated to a non-clustered vSphere machine component. For example, if two machines are associated to a NAT one-to-many network component on the blueprint, you can define a NAT rule that allows port 443 on the external IP to connect to the machines through port 80 on the NAT network using TCP protocol.

NAT rules are not supported for NSX-T load balancers or for NSX-T version 2.2.

You can create any number of NAT rules and you can control the order in which the rules are processed.

The following elements are not supported for NAT rules:

- NICs that are not in the current network
- NICs that are configured to get IP addresses by using DHCP
- Machine clusters

To add NAT rules to a NAT network component in a blueprint, see Add an NSX-T On-Demand NAT or NSX-T On-Demand Routed Network Component .
Add an NSX-T On-Demand NAT or NSX-T On-Demand Routed Network Component

You can add an NSX-T on-demand NAT network component or NSX-T on-demand routed network component to the design canvas in preparation for associating their settings to one or more vSphere machine components in the blueprint.

When you associate an existing network component or an on-demand network component with a machine component, the NIC information is stored with the machine component. The network profile information that you specify is stored with the network component.

You can add multiple network and security components to the design canvas.

You can have more than one on-demand network component in a single blueprint. However, all of the on-demand network profiles that are used in the blueprint must reference the same external network profile.

For NSX-T, the network ranges that are used by the different networks in your blueprint cannot overlap. This restriction surfaces when you are configuring NSX-T Tier-1 router networks.

For vSphere machine components with associated NSX, use network, security, and load balancing setting in the user interface. For machine components that do not have a **Network** or **Security** tab, you can add network and security custom properties, such as `VirtualMachine.Network0.Name`, to their **Properties** tab in the design canvas. NSX network, security, and load balancer properties are only applicable to vSphere machines.

Only the network profiles that are applicable to the current tenant are exposed when authoring a blueprint. Specifically, network profiles are made available if there is at least one reservation in the current tenant that has at least one network assigned to the profile.

Prerequisites

- Create and configure network settings for NSX for vSphere externally. See Checklist for Preparing NSX Network and Security Configuration and *NSX for vSphere Administration Guide* at NSX-T product documentation.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Create an on-demand network profile. See Creating a Network Profile in vRealize Automation.

  For example, if you are adding an on-demand NAT network component see Creating a NAT Network Profile For an On-Demand Network.
- Log in to vRealize Automation as an **infrastructure architect**.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.
- If you want to specify NAT rules for a NAT network component, you must use a NAT one-to-many network profile. See Create a NAT Network Profile By Using the Supplied IPAM Endpoint or Create a NAT Network Profile By Using a Third-Party IPAM Endpoint in vRealize Automation. For information about NAT rules, see Creating and Using NAT Rules for NSX for vSphere.

Procedure

1 To display the list of available network and security components, click **Network & Security** in the Categories section.

2 Drag and NSX-T On-Demand NAT or NSX-T On-Demand Routed network component onto the design canvas.

3 To uniquely label the component in the design canvas, enter a component name in the **ID** text box.

4 Select an appropriate network profile from the **Parent network profile** drop-down menu. For example, if you want to add a NAT network component, select a NAT network profile that is configured to support your intended network settings.

   If you want to specify NAT rules in a NAT network component, you must use a parent network profile that is configured for NAT one-to-many.

   Depending on the profile type you select, the following network settings are populated based on your network profile selection. Changes to these values must be made in the network profile:

   - External network profile name
   - NAT type (NSX-T On-Demand NAT)
   - Subnet mask
   - Range subnet mask (NSX-T On-Demand Routed)
   - Range subnet mask (NSX-T On-Demand Routed)
   - Base IP address (NSX-T On-Demand Routed)

5 (Optional) Enter a component description in the **Description** text box.

6 (Optional) Click the **DNS/WINS** tab.

7 (Optional) Specify DNS and WINS settings for the network profile.

   - Primary DNS
   - Secondary DNS
   - DNS Suffix
   - Preferred WINS
   - Alternate WINS

   You cannot change the DNS or WINS settings for an existing network.

8 Click the **IP Ranges** tab.

   The IP range or ranges specified in the network profile are displayed. You can change the sort order or column display. For NAT networks, you can also change IP range values.

   a Enter a start IP address value in the **IP range start** text box.

   b Enter a start IP address value in the **IP range start** text box.

9 If you are using a NAT network that is based on a one-to-many NAT network profile that uses static IP ranges, you can use the **NAT Rules** tab to add rules that enable an external IP to access components in the internal NAT network.

   For a NAT one-to-many network, you can define NAT rules that can be configured when you add a NAT network component to the blueprint. You can change a NAT rule when you edit the NAT network in a deployment.

   The options that are available for selection are based on the vSphere machine components that you have associated to the NAT network component.

   - **Name** - Enter a unique rule name.
   - **Component** - Select from a list of associated vSphere machine or load balancer components to which the NAT network is associated.

     NAT rules are only supported for non-clustered machines. If you have specified a cluster size of more than 1, no components are listed as the configuration is not supported.
   - **Source port** - Select the ANY option, enter a valid port or port range, or specify a valid property binding.
   - **Destination port** - Select the ANY option, enter a valid port or port range, or specify a valid property binding.
   - **Protocol** - Enter any valid NSX-T-supported protocol or select the TCP, UDP, or ANY option.
   - **Description** - Enter a brief description of what the NAT rule is designed to do.

10 To save the blueprint as draft or continue configuring the blueprint, click **Save** or **Finish**.

What to do next

You can add network settings in the **Network** tab of a vSphere machine component.

Using NSX for vSphere Load Balancer Components in a Blueprint

You can add one or more on-demand NSX for vSphere load balancer components to the design canvas to configure vSphere machine component settings in the blueprint.

The network and security component settings that you add to the blueprint are derived from your NSX for vSphere and NSX-T configuration. For information about configuring NSX, see the *Administration Guide* in NSX for vSphere product documentation or NSX-T product documentation, depending on which application you are using.

The following rules apply to load balancer pools and VIP network settings in the blueprint.

- If the pool network profile is NAT, the VIP network profile can be part of the NAT network profile.
- If the pool network profile is routed, the VIP network profile can only be on the same routed network.
- If the pool network profile is external, the VIP network profile can only be the same external network profile.

Each load balancer component can have multiple virtual servers, which are also referred to as load balancer services. Each virtual server in the load balancer component has one port and protocol. For example, you can load balance an HTTP service or HTTPS service. A load balancer can have multiple services that it is load balancing.

The NSX Edge is the network device that contains the load balancer virtual servers. While you can have more than one load balancer component in a blueprint, when you provision the deployment, the virtual servers defined in each load balancer component are contained in a single NSX Edge.

If a blueprint contains a load balancers, and app isolation is enabled, load balancer VIPs are added to the app isolation security group as an IPSet. If a blueprint contains an on-demand security group that is associated to a machine tier that is associated to a load balancer, the on-demand security group includes the machine tier, IPSet, and VIPs.

You can reconfigure load balancer settings in an existing deployment to add, edit, or remove virtual servers. For information, see Reconfigure a Load Balancer in a Deployment.
Considerations When Working With Upgraded or Migrated Load Balancer Components
The following considerations are important to understand and act on relative to NSX load balancer components in the target vRealize Automation release.

This information applies to NSX for vSphere load balancer components that were upgraded or migrated to this vRealize Automation release.

- You must run NSX Network and Security Inventory data collection before and after upgrading or migrating to this release to avoid issues when running the Reconfigure Load Balancer action. The Reconfigure Load Balancer action for new deployments is not affected.
  - Run NSX Network and Security Inventory Data Collection Before Upgrade
  - Run NSX Network and Security Inventory Data Collection After Upgrade
  - Run NSX Network and Security Inventory Data Collection Before Migration
  - Run NSX Network and Security Inventory Data Collection After Migration
- You can reconfigure a load balancer. The required catalog entitlement is Reconfigure (Load Balancer).

  For related information, see Reconfigure a Load Balancer in a Deployment.

- For deployments that were upgraded or migrated from vRealize Automation 7.x to this vRealize Automation release, load balancer reconfiguration is limited to deployments that contain a single load balancer.
- The Reconfigure Load Balancer operation is not supported for deployments that were upgraded or migrated from vRealize Automation 6.2.x to this vRealize Automation release.

Add an On-Demand Load Balancer Component
You can drag an NSX on-demand load balancer component onto the design canvas and configure its settings for use with vSphere machine components and container components in the blueprint.

For related information about creating NSX for vSphere application profiles to define the behavior of a particular type of network traffic, see the *NSX Administration Guide* in NSX for vSphere product documentation.

Procedure

1 Define Load Balancer Member Settings

   You can define an on-demand NSX load balancer component to distribute task processing among provisioned vSphere member machines or container machines in a network.

2 Define Virtual Server General Settings

   You can define a single virtual server protocol and port for your load balancer or you can add additional virtual servers to customize additional NSX load balancer options.

3 Define Virtual Server Distribution Settings

   By selecting the **Customize** option on the **General** tab, you can specify information about the pool members such as the port on which the members receive traffic, the protocol type that the NSX load balancer can use for accessing that port, the algorithm used for load balancing, and persistence settings.

4 Define Virtual Server Health Check Settings

   By selecting the **Customize** option on the **General** tab, you can specify how, or if, the NSX load balancer performs health checks on pool members within the virtual server.

5 Define Virtual Server Advanced Settings

   By selecting the **Customize** option on the **General** tab, you can customize the NSX load balancer component to specify settings such as the number of concurrent connections that a single pool member can recognize and the maximum number of concurrent connections that the virtual server can process.

6 Define Load Balancer Logging Options

   You can define the types of load balancer logging actions that are captured and recorded in the load balancer logs.

Define Load Balancer Member Settings
You can define an on-demand NSX load balancer component to distribute task processing among provisioned vSphere member machines or container machines in a network.

When you add a load balancer component to a blueprint in the design canvas, you can choose either a default or custom option when creating or editing your virtual server definitions in the load balancer component. The default option allows you to specify the virtual server protocol, port, and description and use defaults for all other settings. The custom option allows you to define additional levels of detail.

If the load balancer is provisioned with an external network, the VIP (VIP network) and member pool (member network) must be on the same existing network. Provisioning fails if the VIP and member pool are not on the same external network.

Prerequisites

- Create and configure load balancer settings for NSX. See Checklist for Preparing NSX Network and Security Configuration and *NSX Administration Guide*.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Create a network profile. See Creating a Network Profile in vRealize Automation.
- Log in to vRealize Automation as an **infrastructure architect**.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.
- Verify that at least one vSphere machine component or container component exists in the blueprint.

Procedure

1. To display the list of available network and security components, click **Network & Security** in the Categories section.
2. Drag an **On-Demand Load Balancer** component onto the design canvas.
3. To uniquely label the component in the design canvas, enter a component name in the **ID** text box.
4. Select a vSphere machine component or container component name from the **Member** drop-down menu.

   The list contains only the vSphere machine components and container components in the active blueprint.
5. Select the NIC to load balance from the **Member network** drop-down menu.

   The list contains NICs that are defined for the selected vSphere machine member.
6. Select an available virtual IP address network from the **VIP Network** drop-down menu. For example, select an available external or NAT network.

   While you can have multiple NSX load balancer and NSX on-demand network components in a blueprint, they must all be associated to the same VIP network.

7 (Optional) Enter a valid IP address for the NIC in the **IP Address** text box.

   The default setting is the static IP address that is associated with the VIP network. You can specify another IP address or an IP address range. By default, the next available IP address is allocated from the associated VIP network.

   Leave the IP address field empty to allow the IP address to be allocated from the associated VIP network during provisioning.

   If specifying an IP address for any other type of network, only one deployment can be provisioned. Subsequent deployments fail IP allocation since the IP is already in use by the first deployment.

8 To create a virtual server definition, click **New** and see Define Virtual Server General Settings.

   Each load balancer component requires at least one virtual server.

   To specify logging options, see Define Load Balancer Logging Options.

Define Virtual Server General Settings
You can define a single virtual server protocol and port for your load balancer or you can add additional virtual servers to customize additional NSX load balancer options.

For example, you can customize the load balancer component to define settings such as health check protocol and port, algorithm, persistence, and transparency.

Prerequisites

Define Load Balancer Member Settings.

Procedure

1 Click the **General** tab on the **New Virtual Server** page.

2 Select the network traffic protocol in the **Protocol** drop-down menu to use for load balancing the virtual server.

   The protocol options are HTTP, HTTPS, TCP, and UDP.

3 Enter a port value in the **Port** text box.

   The selected protocol determines the default port setting.

Protocol	Default port
HTTP	80
HTTPS	443
TCP	8080
UDP	no default

   The HTTP, HTTPS, and TCP protocols can share a port with UDP. For example, if service 1 uses TCP, HTTP, or HTTPS on port 80, service 2 can use UDP on port 80. If service 1 uses UDP on port 80 though, service 2 cannot use UDP on port 80.

4 (Optional) Enter a description for the virtual server component.

5 Select one of the **Settings** options.

   - **Use default value for all other settings**

     Accept all other default settings. Click **OK** to finish the load balancer component definition and continue working in the blueprint.

     You can display the defaults by clicking **Customize** and examining the additional tab options. If the default settings are acceptable, click **Use default value for all other settings** on the **General** tab.

   - **Customize**

     Configure the load balancer component with additional settings, for example to define a different protocol for health monitoring or a different port for monitoring member traffic.

     Additional tabs appear that allow you to add customized settings.

   If you selected **Use default value for all other settings** and clicked **OK** you are done and can continue to define or edit your blueprint in the design canvas. If you selected **Customize**, continue to the step.

6 Click the **Distribution** tab and proceed to the Define Virtual Server Distribution Settings topic to continue defining the virtual server in the NSX load balancer component.

Define Virtual Server Distribution Settings
By selecting the **Customize** option on the **General** tab, you can specify information about the pool members such as the port on which the members receive traffic, the protocol type that the NSX load balancer can use for accessing that port, the algorithm used for load balancing, and persistence settings.

A pool represents a cluster of machines that are being load balanced. A pool member represents one machine in that cluster.

The default member protocol and member port settings match the protocol and port settings on the **General** page.

The pool of member machines is shown in the **Member** option value in the blueprint load balancer component user interface. The **Member** entry is set to the pool or cluster of machines.

Prerequisites

Define Virtual Server General Settings.

Procedure

1 (Optional) The **Member protocol** setting matches the protocol that you specified on the **General** tab. This setting defines how the pool member is to receive network traffic.

2 (Optional) Enter a port number in the **Member port** text box to specify the port on which the pool member is to receive network traffic.

For example, if the incoming request on the load balancer virtual IP address (VIP) is on port 80, you might want to route the request to another port, for example port 8080, on the pool members.

3 (Optional) Select the algorithm balancing method for this pool.

The algorithm options and the algorithm parameters for the options that require them are described in the following table.

Option	Description and algorithm parameters
ROUND_ROBIN	Each server is used in turn according to the weight assigned to it. If the load balancer was created in vRealize Automation, the weight is the same for all members. This is the smoothest and fairest algorithm when the server's processing time remains equally distributed. Algorithm parameters are deactivated for this option.
IP-HASH	Selects a server based on a hash of the source IP address and the total weight of all the running servers. Algorithm parameters are deactivated for this option.
LEASTCONN	Distributes client requests to multiple servers based on the number of connections already on the server. New connections are sent to the server that has the fewest connections. Algorithm parameters are deactivated for this option.
URI	The left part of the URI (before the question mark) is hashed and divided by the total weight of the running servers. The result designates which server receives the request. This ensures that a URI is always directed to the same server as long as no server goes up or down. The URI algorithm parameter has two options -- `uriLength=<len>` and `uriDepth=<dep>`. Enter the length and depth parameters on separate lines in the **Algorithm parameters** text box. Length and depth parameters are followed by a positive integer number. These options can balance servers based on the beginning of the URI only. The length parameter indicates that the algorithm should only consider the defined characters at the beginning of the URI to compute the hash. The length parameter range should be 1<=len<256. The depth parameter indicates the maximum directory depth to be used to compute the hash. One level is counted for each slash in the request. The depth parameter range should be 1<=dep<10. If both parameters are specified, the evaluation stops when either parameter is reached.

Option	Description and algorithm parameters
HTTPHEADER	The HTTP header name is looked up in each HTTP request. The header name in parenthesis is not case sensitive which is similar to the ACL 'hdr()' function. The HTTPHEADER algorithm parameter has one option `headerName=<name>`. For example, you can use **`host`** as the HTTPHEADER algorithm parameter. If the header is absent or does not contain any value, the round robin algorithm is applied.
URL	The URL parameter specified in the argument is looked up in the query string of each HTTP GET request. The URL algorithm parameter has one option `urlParam=<url>`. If the parameter is followed by an equal sign = and a value, then the value is hashed and divided by the total weight of the running servers. The result designates which server receives the request. This process is used to track user identifiers in requests and ensure that a same user ID is always sent to the same server as long as no server goes up or down. If no value or parameter is found, then a round robin algorithm is applied.

4 (Optional) Select the persistence method for this pool.

Persistence tracks and stores session data, such as the specific pool member that serviced a client request. With persistence, client requests are directed to the same pool member for the life of a session or during subsequent sessions.

Protocol	Persistence method supported
HTTP	None, Cookie, Source IP
HTTPS	None, Source IP and SSL Session ID
TCP	None, Source IP, MSRDP
UDP	None, Source IP

- Select **Cookie** to insert a unique cookie to identify the session the first time a client accesses the site. The cookie is referred in subsequent requests to persist the connection to the appropriate server.
- Select **Source IP** to track sessions based on the source IP address. When a client requests a connection to a virtual server that supports source address affinity persistence, the load balancer checks to see if that client previously connected, and if so, returns the client to the same pool member.
- Select **SSL Session ID** and select the SSL Passthrough HTTPS traffic pattern.
  - SSL Passthrough - Client -> HTTPS-> LB (SSL passthrough) -> HTTPS -> server
  - Client - HTTP-> LB -> HTTP -> servers

  Note vRealize Automation currently supports SSL Passthrough only. The SSL Passthrough method is used regardless of which option you select.

- Select **MSRDP** to maintain persistent sessions between Windows clients and servers that are running the Microsoft Remote Desktop Protocol (RDP) service. The recommended scenario for enabling MSRDP persistence is to create a load balancing pool that consists of members running the supported Windows Server, where all members belong to a Windows cluster and participate in a Windows session directory.
- Select **None** to specify that session actions are not stored for subsequent recall.

5 If you are using a cookie persistence setting, enter the cookie name.

6 (Optional) Select the mode by which the cookie is inserted from the **Mode** drop-down menu.

Option	Description
Insert	The NSX Edge sends a cookie. If the server sends one or more cookies, the client receives an extra cookie (the server cookie(s) + the NSX Edge cookie). If the server does not send a cookie, the client receives the NSX Edge cookie.
Prefix	The server sends a cookie. Use this option if your client does not support more than one cookie. If you have a proprietary application using a proprietary client that supports only one cookie, the Web server sends a cookie but the NSX Edge injects (as a prefix) its cookie information in the server cookie value
App Session	The server does not send a cookie. Instead, it sends the user session information as a URL. For example, http://mysite.com/admin/UpdateUserServlet;jsessionid=X000X0XXX0XXXX, where `jsessionid` is the user session information and is used for persistence.

7 (Optional) Enter the persistence expiration time for the cookie in seconds.

As an example, for L7 load balancing with a TCP source IP, the persistence entry times out if no new TCP connections are made for the specified expiration time, even if the existing connections are still live.

8 (Optional) Click the **Health Check** tab and proceed to the Define Virtual Server Health Check Settings topic to continue defining the virtual server in the NSX load balancer component.

Define Virtual Server Health Check Settings

By selecting the **Customize** option on the **General** tab, you can specify how, or if, the NSX load balancer performs health checks on pool members within the virtual server.

The default health check protocol and health check port settings match the protocol and port settings on the **General** tab.

For related information see *Create a Service Monitor* in NSX Product Documentation at https://www.vmware.com/support/pubs/nsx_pubs.html. Note that the NSX documentation refers to the virtual server member as a pool member.

Prerequisites

Define Virtual Server General Settings.

Procedure

1 (Optional) Select a heath check protocol in the **Health check protocol** drop-down menu to specify how the pool member is accessed when the load balancer listens to determine the health of the pool member.

  The protocol options are **HTTP**, **HTTPS**, **TCP**, **ICMP**, **UDP**, and **None**.

  You can also accept the default protocol as specified on the General tab.

2 (Optional) Enter a value in the **Health check port** box to specify on which port the load balancer listens to monitor the health of the virtual server member or pool member.

  Note that the NSX documentation refers to a virtual server member as a pool member.

  The HTTP, HTTPS, and TCP protocols can share a port with UDP. For example, if service 1 uses TCP, HTTP, or HTTPS on port 80, service 2 can use UDP on port 80. If service 1 uses UDP on port 80 though, service 2 cannot use UDP on port 80.

3 Enter the **Interval** value in seconds at which a server is to be pinged.

4 Enter the maximum **Timeout** value in seconds within which a response from the server must be received.

5 Enter a **Max. retries** value as the number of times the server must be pinged before it is declared down.

6 Specify additional health check settings based on your selected **Health check protocol**.

  a Enter the **Method** to be used for detecting server status. The options are GET, OPTIONS, and POST.

  b Enter the **URL** to be used in the request for detecting server status. This is the URL that is used for by GET and POST ("/" by default) method options.

  c In the **Send** text box, enter the string to be sent to the server after a connection is established.

    In the **Send** text box, enter the string to be sent to the server after a connection is established.

  d In the **Receive** text box, enter the string expected to receive from the server.

    Only when the received string matches this definition is the server is considered as up.

    The string can be a header or in the body of the response.

7 Click the **Advanced** tab and proceed to the Define Virtual Server Advanced Settings topic to continue defining the virtual server in the NSX load balancer component.

  To specify logging options, see Define Load Balancer Logging Options.

Define Virtual Server Advanced Settings

By selecting the **Customize** option on the **General** tab, you can customize the NSX load balancer component to specify settings such as the number of concurrent connections that a single pool member can recognize and the maximum number of concurrent connections that the virtual server can process.

Prerequisites

Define Virtual Server General Settings.

Procedure

1 Enter a value in the **Connection limit** text box to specify the maximum concurrent connections in NSX that the virtual server can process.

   This setting considers the number of all member connections.

   Enter a value of 0 to specify no limit.

2 Enter a value in the **Connection rate limit** text box to specify the maximum number of incoming connection requests in NSX that can be accepted per second.

   This setting considers the number of all member connections.

   Enter a value of 0 to specify no limit.

3 (Optional) Select the **Enable acceleration** check box to specify that each virtual IP (VIP) uses the faster L4 load balancer rather than the L7 load balancer.

4 (Optional) Select the **Transparent** check box to allow the load balancer pool members to view the IP address of the machines that are calling the load balancer.

   If not selected, the members of the load balancer pool view the traffic source IP address as a load balancer internal IP address.

5 Enter a value in the **Max connections** text box to specify the maximum number of concurrent connections that a single pool member can recognize.

   If the number of incoming requests is higher than this value, requests are queued and then processed in the order in which they are received as connections are released.

   Enter a value of 0 to specify no maximum value.

6 Enter a value in the **Min connections** text box to specify the minimum number of concurrent connections that a single pool member must always accept.

   Enter a value of 0 to specify no minimum value.

7 Click **OK** to complete the virtual server definition.

8 To specify logging options, see Define Load Balancer Logging Options, otherwise click **Save** or **Finish**.

Define Load Balancer Logging Options

You can define the types of load balancer logging actions that are captured and recorded in the load balancer logs.

After you define a load balancer component, or while you are defining a load balancer component, you can specify a logging level for collecting load balancer traffic logs. The logging levels that you define for any load balancer component on the blueprint apply to all load balancers that are defined in the blueprint.

Logging levels include debug, info, warning, error, and critical. Debug and info options log user requests while warning, error, and critical options do not log users requests.

For additional information about NSX load balancer logging, see the *NSX Administration Guide*.

Prerequisites

Define Load Balancer Member Settings.

Procedure

1 Select the **Global** tab on the load balancer component in the design canvas.

2 Select one or more logging options from the **Logging level** drop-down menu.

   Select a logging level for collecting load balancer traffic logs. The setting applies to all NSX load balancer components in the blueprint.

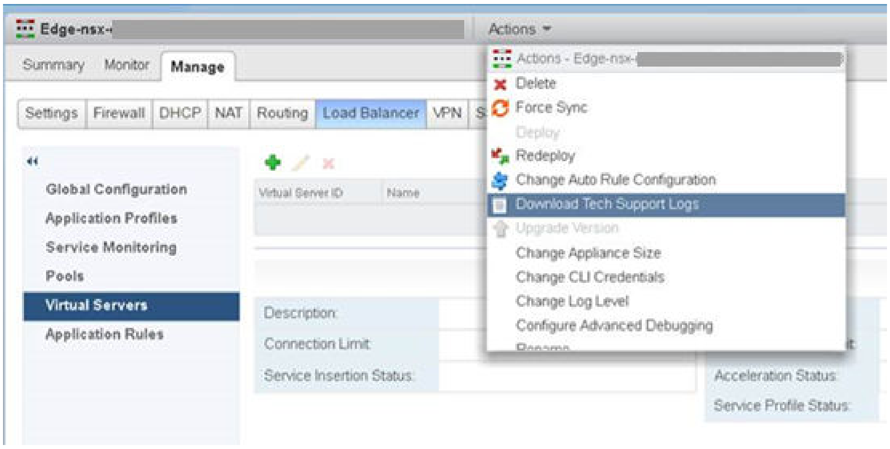
   The logging settings are defined in the vSphere web client.

   - None
   - Info
   - Emergency
   - Alert
   - Critical
   - Error
   - Warning
   - Notice
   - Debug

3 Click **Save**.

Results

You can view and download the logs in the vSphere web client by using the **Actions** menu for the NSX Edge as described in *Download Tech Support Logs for NSX Edge* in NSX Product Documentation at https://www.vmware.com/support/pubs/nsx_pubs.html.



Using NSX-T Load Balancer Components in a Blueprint

You can add one or more on-demand NSX-T load balancer components to the design canvas to configure vSphere machine component settings in the blueprint.

The network and security component settings that you add to the blueprint are derived from your NSX for vSphere and NSX-T configuration. For information about configuring NSX, see the *Administration Guide* in NSX for vSphere product documentation or NSX-T product documentation, depending on which application you are using.

The network and security component settings that you add to the blueprint are derived from your NSX-T configuration. For information about configuring NSX-T, see the *NSX-T Administration Guide* in NSX-T product documentation.

The following rules apply to load balancer pools and VIP network settings in the blueprint.

- If the pool network profile is NAT, the VIP network profile can be part of the NAT network profile.
- If the pool network profile is routed, the VIP network profile can only be on the same routed network or same external network.
- If the pool network profile is external, the VIP network profile can only be the same external network profile.

Each load balancer component can have multiple virtual servers, which are also referred to as load balancer services. Each virtual server in the load balancer component has one port and protocol. For example, you can load balance an HTTP service or HTTPS service. A load balancer can have multiple services that it is load balancing.

The NSX load balancer is the service that contains the load balancer virtual servers.

If a blueprint contains a load balancers, and app isolation is enabled, load balancer VIPs are added to the app isolation security group as an IPSet. If a blueprint contains an on-demand security group that is associated to a machine tier that is associated to a load balancer, the on-demand security group includes the machine tier, IPSet, and VIPs.

For more information about NSX-T-specific deployment and topology considerations, see Understanding NSX-T Deployment Topologies for Networking, Security, and Load Balancer Configurations.

Add an NSX-T On Demand Load Balancer

You can drag an NSX-T on-demand load balancer component onto the design canvas and configure its settings for use with vSphere machine components and container components in the blueprint.

The NSX-T load balancer distributes incoming service requests evenly among multiple servers in such a way that the load distribution is transparent to users. Load balancing helps in achieving optimal resource utilization, maximizing throughput, minimizing response time, and avoiding overload.

You can map a virtual IP address to a set of pool servers for load balancing. The load balancer accepts TCP, UDP, HTTP, or HTTPS requests on the virtual IP address and decides which pool member to use. A load balancer is attached to a Tier-1 logical router.

Depending on your environment needs, you can scale the load balancer performance by increasing the existing virtual servers and pool members to handle heavy network traffic load.

For information about creating NSX-T load balancers to define the behavior of network traffic, see *Logical Load Balancer* and *Configuring Load Balancer Components* in the *NSX-T Administration Guide* in NSX-T product documentation.

Procedure

1. Define NSX-T Load Balancer Member Settings

   You can define an NSX-T on-demand load balancer component to distribute task processing among provisioned vSphere member machines or container machines in a network.

2. Define Virtual Server General Settings for NSX-T

   You can define a single virtual server protocol and port for your load balancer or you can add additional virtual servers to customize additional NSX-T load balancer options.

3. Define Virtual Server Distribution Settings for NSX-T

   By selecting the **Customize** option when you define a virtual server, you can specify pool member information such as the port on which the members receive traffic, the protocol type that the NSX-T load balancer can use for accessing that port, the algorithm used for load balancing, and persistence settings.

4. Define Virtual Server Health Check Settings for NSX-T

   By selecting the **Customize** option on the **General** tab, you can specify how, or if, the NSX-T load balancer performs health checks on pool members within the virtual server.

5. Define Virtual Server Advanced Settings for NSX-T

   By selecting the **Customize** option on the **General** tab, you can customize the NSX-T load balancer component to specify settings such as the number of concurrent connections that a single pool member can recognize and the maximum number of concurrent connections that the virtual server can process.

6 Define NSX-T Load Balancer Logging Options

   You can define the types of load balancer logging actions that are captured and recorded in the load balancer logs.

Define NSX-T Load Balancer Member Settings

You can define an NSX-T on-demand load balancer component to distribute task processing among provisioned vSphere member machines or container machines in a network.

When you add a load balancer component to a blueprint in the design canvas, you can choose either a default or custom option when creating or editing your virtual server definitions in the load balancer component. The default option allows you to specify the virtual server protocol, port, and description and use defaults for all other settings. The custom option allows you to define additional levels of detail.

If the load balancer is provisioned with an external network, the VIP (VIP network) and member pool (member network) must be on the same existing network. Provisioning fails if the VIP and member pool are not on the same external network.

Prerequisites

- Create and configure load balancer settings for NSX. See Checklist for Preparing NSX Network and Security Configuration.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Create a network profile. See Creating a Network Profile in vRealize Automation.
- Log in to vRealize Automation as an **infrastructure architect**.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.
- Verify that at least one vSphere machine component or container component exists in the blueprint.

Procedure

1 To display the list of available network and security components, click **Network & Security** in the Categories section.

2 Drag an **NSX-T On-Demand Load Balancer** component onto the design canvas.

3 To uniquely label the component in the design canvas, enter a component name in the **ID** text box.

4 Select a vSphere machine component or container component name from the **Member** drop-down menu.

   The list contains only the vSphere machine components and container components in the active blueprint.

5 Select the NIC to load balance from the **Member network** drop-down menu.

The list contains NICs that are defined for the selected vSphere machine member.

6 Select an available virtual IP address network from the **VIP Network** drop-down menu. For example, select an available external or NAT network.

While you can have multiple NSX load balancer and NSX on-demand network components in a blueprint, they must all be associated to the same VIP network.

7 (Optional) Enter a valid IP address for the NIC in the **IP Address** text box.

The default setting is the static IP address that is associated with the VIP network. You can specify another IP address or an IP address range. By default, the next available IP address is allocated from the associated VIP network.

Leave the IP address field empty to allow the IP address to be allocated from the associated VIP network during provisioning.

If specifying an IP address for any other type of network, only one deployment can be provisioned. Subsequent deployments fail IP allocation since the IP is already in use by the first deployment.

8 To create a virtual server definition, click **New** and see Define Virtual Server General Settings for NSX-T.

Each load balancer component requires at least one virtual server.

To specify logging options, see Define NSX-T Load Balancer Logging Options.

Define Virtual Server General Settings for NSX-T

You can define a single virtual server protocol and port for your load balancer or you can add additional virtual servers to customize additional NSX-T load balancer options.

For example, you can customize the load balancer component to define settings such as health check protocol and port, algorithm, persistence, and transparency.

Prerequisites

Define NSX-T Load Balancer Member Settings.

Procedure

1 Click the **General** tab on the **Virtual Server** page.

2 Select the network traffic protocol in the **Protocol** drop-down menu to use for load balancing the virtual server.

The protocol options are HTTP, HTTPS, TCP, and UDP.

NSX-T load balances do not support SSL passthough mode but instead use SSL termination mode. If you specify HTTPS, you'll need to provide the following additional information, which must already exist in the NSX-T manager:

- Name of the certificate in the NSX-T certificate inventory. The load balancer presents this certificate to clients.
- Name of the client SSL profile.

3 Enter a port value in the **Port** text box.

The selected protocol determines the default port setting.

Protocol	Default port
HTTP	80
HTTPS	443
TCP	8080
UDP	no default

The HTTP, HTTPS, and TCP protocols can share a port with UDP. For example, if service 1 uses TCP, HTTP, or HTTPS on port 80, service 2 can use UDP on port 80. If service 1 uses UDP on port 80 though, service 2 cannot use UDP on port 80.

4 (Optional) Enter a description for the virtual server component.

5 Click the **Distribution** tab and proceed to the Define Virtual Server Distribution Settings for NSX-T topic to continue defining the virtual server in the NSX-T load balancer component.

Define Virtual Server Distribution Settings for NSX-T

By selecting the **Customize** option when you define a virtual server, you can specify pool member information such as the port on which the members receive traffic, the protocol type that the NSX-T load balancer can use for accessing that port, the algorithm used for load balancing, and persistence settings.

A pool represents a cluster of machines that are being load balanced. A pool member represents one machine in that cluster.

The default member protocol and member port settings match the protocol and port settings on the **General** page.

The pool of member machines is shown in the **Member** option value in the blueprint load balancer component user interface. The **Member** entry is set to the pool or cluster of machines.

Prerequisites

Define NSX-T Load Balancer Member Settings.

Procedure

1 (Optional) The **Member protocol** setting matches the protocol that you specified on the **General** tab. This setting defines how the pool member is to receive network traffic.

2 (Optional) Enter a port number in the **Member port** text box to specify the port on which the pool member is to receive network traffic.

   For example, if the incoming request on the load balancer virtual IP address (VIP) is on port 80, you might want to route the request to another port, for example port 8080, on the pool members.

3 (Optional) Select the algorithm balancing method for this pool.

   The algorithm options and the algorithm parameters for the options that require them are described in the following table.

   For related information, see *Add a Server Pool for Load Balancing* in NSX-T product documentation.

Option	Description and algorithm parameters
ROUND_ROBIN	Incoming client requests are cycled through a list of available servers capable of handling the request. Ignores the server pool member weights even if they are configured.
WEIGHTED ROUND ROBIN	Each server is assigned a weight value that signifies how that server performs relative to other servers in the pool. The value determines how many client requests are sent to a server compared to other servers in the pool. This load balancing algorithm focuses on fairly distributing the load among the available server resources.
IP-HASH	Selects a server based on a hash of the source IP address and the total weight of all the running servers.
LEASTCONN	Distributes client requests to multiple servers based on the number of connections already on the server. New connections are sent to the server with the fewest connections. Ignores the server pool member weights even if they are configured.
WEIGHTED LEASTCONN	Each server is assigned a weight value that signifies how that server performs relative to other servers in the pool. The value determines how many client requests are sent to a server compared to other servers in the pool. This load balancing algorithm focuses on using the weight value to fairly distribute the load among the available server resources. By default, the weight value is 1 if the value is not configured and slow start is enabled.

4 (Optional) Select the persistence method for this pool.

   Persistence tracks and stores session data, such as the specific pool member that serviced a client request. With persistence, client requests are directed to the same pool member for the life of a session or during subsequent sessions. For more information about the persistence methods, see *Configure Persistent Profiles* in NSX-T product documentation.

   - Select **None** to specify that session actions are not stored for subsequent recall.
   - Select **Cookie** to insert a unique cookie to identify the session the first time a client accesses the site. The cookie is referred in subsequent requests to persist the connection to the appropriate server.

- Select **Source IP** to track sessions based on the source IP address. When a client requests a connection to a virtual server that supports source address affinity persistence, the load balancer checks to see if that client previously connected, and if so, returns the client to the same pool member.

5 If you are using cookie persistence, enter the cookie name.

6 (Optional) Select the mode by which the cookie is inserted from the **Mode** drop-down menu.

Option	Description
Insert	Create a unique cookie to identify the session.
Prefix	Adds to the existing cookie.
Rewrite	Overwrites the existing cookie.

7 (Optional) Enter the persistence expiration time for the cookie in seconds.

As an example, for L7 load balancing with a TCP source IP, the persistence entry times out if no new TCP connections are made for the specified expiration time, even if the existing connections are still live.

8 (Optional) Click the **Health Check** tab and proceed to the Define Virtual Server Health Check Settings for NSX-T topic to continue defining the virtual server in the NSX-T load balancer component.

Define Virtual Server Health Check Settings for NSX-T

By selecting the **Customize** option on the **General** tab, you can specify how, or if, the NSX-T load balancer performs health checks on pool members within the virtual server.

The default health check protocol and health check port settings match the protocol and port settings on the **General** tab.

For related information see NSX-T product documentation. Note that the NSX-T documentation refers to the virtual server member as a pool member.

Prerequisites

Define Virtual Server Distribution Settings for NSX-T.

Procedure

1 (Optional) Select a heath check protocol in the **Health check protocol** drop-down menu to specify how the pool member is accessed when the load balancer listens to determine the health of the pool member.

The protocol options are **None**, **HTTP**, **HTTPS**, **TCP**, **ICMP**, and **UDP**.

You can also accept the default protocol as specified on the General tab.

2 (Optional) Enter a value in the **Health check port** box to specify on which port the load balancer listens to monitor the health of the virtual server member or pool member.

Note that the NSX documentation refers to a virtual server member as a pool member.

The HTTP, HTTPS, and TCP protocols can share a port with UDP. For example, if service 1 uses TCP, HTTP, or HTTPS on port 80, service 2 can use UDP on port 80. If service 1 uses UDP on port 80 though, service 2 cannot use UDP on port 80.

3. Enter the **Interval** value in seconds at which a server is to be pinged.
4. Enter the maximum **Timeout** value in seconds within which a response from the server must be received.
5. Enter a **Max. retries** value as the number of times the server must be pinged before it is declared down.
6. If you specified an HTTP or HTTPS protocol, enter the **Method** to be used for detecting server status.
7. If available, enter the **URL** to be used in the request for detecting server status. This is the URL that is used for by GET and POST ("/" by default) method options.
8. If available, enter sending and receiving strings in the **Send** and **Receive** text boxes.

   In the **Send** text box, enter the string to be sent to the server after a connection is established.

   In the **Receive** text box, enter the string expected to receive from the server. Only when the received string matches this definition is the server is considered as up.
9. Click the **Advanced** tab and proceed to the Define Virtual Server Advanced Settings for NSX-Ttopic to continue defining the virtual server in the NSX-T load balancer component.

   To specify logging options, see Define NSX-T Load Balancer Logging Options.

Define Virtual Server Advanced Settings for NSX-T
By selecting the **Customize** option on the **General** tab, you can customize the NSX-T load balancer component to specify settings such as the number of concurrent connections that a single pool member can recognize and the maximum number of concurrent connections that the virtual server can process.

Prerequisites

Define Virtual Server General Settings for NSX-T.

Procedure

1. Enter a value in the **Connection limit** text box to specify the maximum concurrent connections in NSX-T that the virtual server can process.

   This setting considers the number of all member connections.

   Enter a value of 0 to specify no limit.
2. Enter a value in the **Connection rate limit** text box to specify the maximum number of incoming connection requests in NSX-T that can be accepted per second.

   This setting considers the number of all member connections.

Enter a value of 0 to specify no limit.

3 (Optional) Select the **Transparent** check box to allow the load balancer pool members to view the IP address of the machines that are calling the load balancer.

   If not selected, the members of the load balancer pool view the traffic source IP address as a load balancer internal IP address.

4 Enter a value in the **Max connections** text box to specify the maximum number of concurrent connections that a single pool member can recognize.

   If the number of incoming requests is higher than this value, requests are queued and then processed in the order in which they are received as connections are released.

   Enter a value of 0 to specify no maximum value.

5 Click **OK** to complete the virtual server definition.

6 To specify logging options, see Define NSX-T Load Balancer Logging Options, otherwise click **Save** or **Finish**.

Define NSX-T Load Balancer Logging Options
You can define the types of load balancer logging actions that are captured and recorded in the load balancer logs.

You can specify a logging level for collecting load balancer traffic logs. The logging levels that you define for any NSX-T load balancer component on the blueprint apply to all load balancers in the blueprint.

Logging levels include debug, info, warning, error, and critical. Debug and info options log user requests while warning, error, and critical options do not log users requests.

For additional information about NSX-T load balancer logging, see the *NSX-T Administration Guide* in NSX-T product documentation.

Prerequisites

Define NSX-T Load Balancer Member Settings

Procedure

1 Select the **Global** tab on the load balancer component in the design canvas.

2 Select one or more logging options from the **Logging level** drop-down menu.

   The logging settings are defined in the vSphere web client.

   - None
   - Emergency
   - Alert
   - Critical
   - Error

- Warning
- Info
- Debug

3 Select a small, medium, or large load balancer size.

4 Click **Save** and then click **Finish**.

Using NSX for vSphere Security Components in a Blueprint

You can add NSX for vSphere security components to the design canvas to make their configured settings available to one or more vSphere machine components in the blueprint.

Security groups, tags, and policies are configured outside of vRealize Automation in the NSX application.

The network and security component settings that you add to the blueprint are derived from your NSX for vSphere and NSX-T configuration. For information about configuring NSX, see the *Administration Guide* in NSX for vSphere product documentation or NSX-T product documentation, depending on which application you are using.

You can add security controls to blueprints by configuring security groups, tags, and policies for the vSphere compute resource in NSX. After you run data collection, the security configurations are available for selection in vRealize Automation.

For a sample NSX for vSphere security strategy, see this vRealize and NSX blog post.

Existing and On-Demand Security Groups for NSX for vSphere

A security group is a collection of assets or grouping objects from the vSphere inventory that is mapped to a set of security policies, for example distributed firewall rules and third party security service integrations such as anti-virus and intrusion detection. The grouping feature enables you to create custom containers to which you can assign resources, such as virtual machines and network adapters, for distributed firewall protection. After a group is defined, you can add the group as source or destination to a firewall rule for protection.

You can add vSphere existing or on-demand security groups to a blueprint, in addition to the security groups specified in the reservation.

You can create one or more on-demand security groups. You can select one or more security policies to configure on a security group.

A security policy is a set of endpoint, firewall, and network introspection services that can be applied to a security group. You can add security policies to a vSphere virtual machine by using an on-demand security group in a blueprint. You cannot add a security policy directly to a reservation. After data collection, security policies that were defined in NSX for vSphere for a compute resource are available for selection in a blueprint.

Security groups are managed in the source resource. For information about managing security groups for various resource types, see the NSX for vSphere documentation.

Note When App isolation is enabled, a separate security policy is created. App isolation uses a logical firewall to block all inbound and outbound traffic to the applications in the blueprint. Component machines that are provisioned by a blueprint that contains an app isolation policy can communicate with each other but cannot connect outside the firewall unless other security groups are added to the blueprint with security policies that allow access.

If a blueprint contains a load balancers, and app isolation is enabled, load balancer VIPs are added to the app isolation security group as an IPSet. If a blueprint contains an on-demand security group that is associated to a machine tier that is associated to a load balancer, the on-demand security group includes the machine tier, IPSet, and VIPs.

Existing Security Tags for NSX for vSphere

You can add exiting security tag components for NSX for vSphere. A security tag is a qualifier object or categorizing entry that you can use as a grouping mechanism. You define the criteria that an object must meet to be added to the security group you are creating. This gives you the ability to include machines by defining a filter criteria with a number of parameters supported to match the search criteria. For example, you can add all of the machines tagged with a specified security tag to a security group.

Add an Existing Security Group Component for NSX for vSphere

You can add an existing NSX for vSphere security group component to the design canvas in preparation for associating its settings to one or more vSphere machine components in the blueprint.

You can use an existing security group component to add an NSX security group to the design canvas and configure its settings for use with vSphere machine components and Software or XaaS components that pertain to vSphere.

By default, security groups that are applicable to the current tenant are exposed when authoring a blueprint. Specifically, security groups are made available if the associated endpoint has a reservation in the current tenant. For additional information about controlling tenancy access, see Controlling Tenant Access for Security Objects in vRealize Automation.

Prerequisites

- Create and configure security groups for NSX. See Checklist for Preparing NSX Network and Security Configuration and the *NSX for vSphere Administration Guide* in NSX for vSphere product documentation.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Review security component concepts. See Using NSX for vSphere Security Components in a Blueprint.
- Log in to vRealize Automation as an **infrastructure architect**.

- Open a new or existing blueprint in the design canvas by using the **Design** tab.

Procedure

1. To display the list of available network and security components, click **Network & Security** in the Categories section.
2. Drag an **Existing Security Group** component onto the design canvas.
3. Select an existing security group from the **Security Group** drop-down menu.
4. Click **OK**.
5. To save the blueprint as draft or continue configuring the blueprint, click **Save** or **Finish**.

Results

You can add security settings in the **Security** tab of a vSphere machine component.
Add an Existing Security Tag Component for NSX for vSphere
You can add an NSX for vSphere existing security tag component to the blueprint design canvas in preparation for associating its settings to one or more vSphere components in the blueprint.

You can use a security tag component to add an vSphere existing security tag to the design canvas and configure its settings for use with vSphere machine components and Software components that pertain to vSphere.

By default, security tags that are applicable to the current tenant are exposed when authoring a blueprint. Specifically, security tags are made available if the associated endpoint has a reservation in the current tenant. For additional information about controlling tenancy access, see Controlling Tenant Access for Security Objects in vRealize Automation.

You can add multiple network and security components to the design canvas.

For more information, see Using NSX for vSphere Security Components in a Blueprint.

Prerequisites

- Create and configure security tags for NSX. See Checklist for Preparing NSX Network and Security Configuration and the *NSX for vSphere Administration Guide* in NSX for vSphere product documentation.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Log in to vRealize Automation as an **infrastructure architect**.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.

Procedure

1. To display the list of available network and security components, click **Network & Security** in the Categories section.
2. Drag an **Existing Security Tag** component onto the design canvas.

3 Click in the **Security tag** text box and select an existing security tag.

4 Click **OK**.

5 To save the blueprint as draft or continue configuring the blueprint, click **Save** or **Finish**.

Results

You can add security settings in the **Security** tab of a vSphere machine component.
Add an On-Demand Security Group Component
You can add an on-demand NSX security group component to the design canvas in preparation for associating its settings to one or more vSphere machine components or other available component types in the blueprint.

When you create an on-demand security group you add security policies to create the group. The security policies can be globally exposed or hidden by default. Policies are only exposed in tenants for which the associated NSX endpoint has a reservation in that tenant.

By default, security groups that are applicable to the current tenant are exposed when authoring a blueprint. Specifically, security groups are made available if the associated endpoint has a reservation in the current tenant. For additional information about controlling tenancy access, see Controlling Tenant Access for Security Objects in vRealize Automation.

Prerequisites

- Create and configure a security policy in NSX. See *NSX Administration Guide*.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Log in to vRealize Automation as an **infrastructure architect**.
- Review security component concepts. See Using NSX for vSphere Security Components in a Blueprint.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.

Procedure

1 To display the list of available network and security components, click **Network & Security** in the Categories section.

2 Drag an **On-Demand Security Group** component onto the design canvas.

3 Enter a name and, optionally, a description.

4 Add one or more security policies by clicking the **Add** icon in the **Security policies** area and selecting available security policies.

5 Click **OK**.

6 To save the blueprint as draft or continue configuring the blueprint, click **Save** or **Finish**.

Results

You can add security settings in the **Security** tab of a vSphere machine component.

Using NSX-T Security Components in a Blueprint

You can add an NSX-T network security component to the design canvas to make its configured settings available to one or more associated vSphere machine components in the blueprint.

An NSX-T Existing NS Group enables you to assign resources, such as virtual machines and network adapters, for distributed firewall protection.

You can add security controls to blueprints by configuring NS Groups for the vSphere compute resource in NSX-T. After you run data collection, the security configurations are available for selection in vRealize Automation. You can add an NSX-T Existing NS Group component to the blueprint as a source or destination to a firewall rule.

NSX-T NS security groups are managed outside of vRealize Automation in the NSX-T application. For information about managing NS Groups, see the NSX-T product documentation.

The network and security component settings that you add to the blueprint are derived from your NSX for vSphere and NSX-T configuration. For information about configuring NSX, see the *Administration Guide* in NSX for vSphere product documentation or NSX-T product documentation, depending on which application you are using.

When you deploy a blueprint that contains an NSX-T endpoint, the deployment assigns a tag to NSX-T components in the deployment. The tag name and the deployment name match.

When app isolation is enabled, a new firewall section with rules is created for a deployment. App isolation uses a logical firewall to block all inbound and outbound traffic to the applications in the blueprint. Component machines that are provisioned by a blueprint that contains an app isolation policy can communicate with each other but cannot connect outside the firewall unless other NS Groups are added to the blueprint with security rules that allow access.

If a blueprint contains a load balancers, and app isolation is enabled, load balancer VIPs are added to the app isolation security group as an IPSet. If a blueprint contains an on-demand security group that is associated to a machine tier that is associated to a load balancer, the on-demand security group includes the machine tier, IPSet, and VIPs.

For NSX-T, app isolation is the only on-demand NS Group created. It contains an IP set that includes load balancer VIPs and NAT one-to-many network external IPs.

For more information about NSX-T-specific deployment and topology considerations, see Understanding NSX-T Deployment Topologies for Networking, Security, and Load Balancer Configurations.

Add an NSX-T NSGroup Component

You can add an NSX-T Existing NS Group component to the design canvas and configure its settings for use with vSphere machine components and their other associated components, such as software and network components.

An NSX-T NS Group can contain a combination of IP sets, MAC sets, logical ports, logical switches, and other NSGroups. You can specify NSGroups as sources and destinations in firewall rules. For more information about NSGroup characteristics, see *Create an NSGroup* in the *NSX-T Administration Guide* in NSX-T product documentation.

Note The NSGroup security is applied to VMs that are connected to opaque networks that are managed by NSX-T. If a VM is connected to a vSphere dvPortGroup, micro-segmentation is not available for that network.

By default, NSGroups that apply to the current tenant are exposed when you create or edit a blueprint. Security groups are made available if the associated endpoint has a reservation in the current tenant. For additional information about controlling tenancy access, see Controlling Tenant Access for Security Objects in vRealize Automation.

Prerequisites

- Create and configure an NS Group in NSX-T. See Checklist for Preparing NSX Network and Security Configuration.
- Verify that the NSX inventory has run successfully for your cluster.

  To use NSX configurations in vRealize Automation, you must run data collection.
- Review security component concepts. See Using NSX-T Security Components in a Blueprint.
- Log in to vRealize Automation as an **infrastructure architect**.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.

Procedure

1 To display the list of available network and security components, click **Network & Security** in the Categories section.

2 Drag an **NSX-T NSGroup** component onto the design canvas.

3 Select an NSGroup from the drop-down menu.

4 If prompted, enter an associated endpoint.

5 To save the blueprint as draft or continue configuring the blueprint, click **Save** or **Finish**.

Results

You can add security settings in the **Security** tab of a vSphere machine component.

Associating Network and Security Components

You can drag network and security components onto the design canvas to make their settings available for machine component configuration in the blueprint. After you have defined network and security settings for the machine, you can optionally associate settings from a load balancer component.

After you add an NSX network or security component to the design canvas and define its available settings, you can open the network and security tabs of a vSphere machine component in the canvas and configure its settings.

You can drag an on-demand NAT network component onto the design canvas and associate it with a vSphere machine component or NSX load balancer component in the blueprint.

The network and security component settings that you add to the blueprint are derived from your NSX for vSphere and NSX-T configuration. For information about configuring NSX, see the *Administration Guide* in NSX for vSphere product documentation or NSX-T product documentation, depending on which application you are using.

Note If a blueprint contains a load balancers, and app isolation is enabled, load balancer VIPs are added to the app isolation security group as an IPSet. If a blueprint contains an on-demand security group that is associated to a machine tier that is associated to a load balancer, the on-demand security group includes the machine tier, IPSet, and VIPs.

For information about using NAT rules to allow a TCP or UDP port to map from the external IP address of an Edge (source port) to a private IP address in the NAT network component (target port), see Creating and Using NAT Rules for NSX for vSphere or Creating and Using NAT Rules for NSX-T.

For more information about NSX-T-specific deployment and topology considerations, see Understanding NSX-T Deployment Topologies for Networking, Security, and Load Balancer Configurations.

Configuring a Blueprint to Provision from an OVF

You can use an OVF to define vSphere machine properties and hardware settings that are ordinarily defined on blueprint configuration pages in vRealize Automation or programmatically by using vRealize Automation REST APIs or vRealize CloudClient.

You can also import settings from an OVF to define a value set for an image component profile. Parameterized blueprints use the image and size component profile types.

OVF is an open-source standard for packaging and distributing software applications for virtual machines.

OVF provisioning is similar to cloning, except that the source machine is an OVF template that's hosted on a server or a Web site, instead of a virtual machine template that's hosted in vCenter.

An OVF file is typically used to describe a single virtual machine or virtual appliance. It can contain information about the format of a virtual disk image file and a description of the virtual hardware that should be emulated to run the OS or application contained on the disk image. An OVA file is a virtual appliance package that contains files used to describe a virtual machine, including an OVF descriptor file, optional manifest and certificate files, and other related files.

The `ImportOvfWorkflow` provisioning option is available on a vSphere machine component when you define a blueprint. It's also available when you define a value set for an image component profile in the property dictionary.

You can add blueprint configuration settings to an OVF to describe the following types of information:

- Minimum CPU, memory, and storage allocations.
- User-configurable custom properties.
- Component profile settings for blueprint parameterization.

OVF and OVA with multiple machines is not supported.

Essential considerations include the following statements:

- OVF files and OVA packages are supported.
- Basic user name and password authentication for the HTTP server on which the hosted OVF or OVA resides is supported. The specified URL is validated in the blueprint.
- OVFs and OVAs are not data-collected from the vCenter Server.
- EBS subscriptions are supported.
- You can define custom properties when you import user-configurable OVF settings into the blueprint.
- You can add, change, or remove settings obtained from an OVF import when requesting vSphere machine provisioning.
- You can add, change, or remove, settings during machine reconfiguration.

Define Blueprint Settings for a vSphere Component By Using an OVF

You can import settings from an OVF to simplify the process of configuring vSphere machine component settings in a vRealize Automation blueprint.

This procedure assumes that you have a basic familiarity with the vRealize Automation blueprint creation process.

Prerequisites

- Log in to vRealize Automation as an **infrastructure architect**.
- Meet the remaining prerequisites specified in Configure a Machine Blueprint.

Procedure

1 Select **Design > Blueprints**.

2 Click the **New** icon (✚).

3 Enter a blueprint name and description and click **OK**.

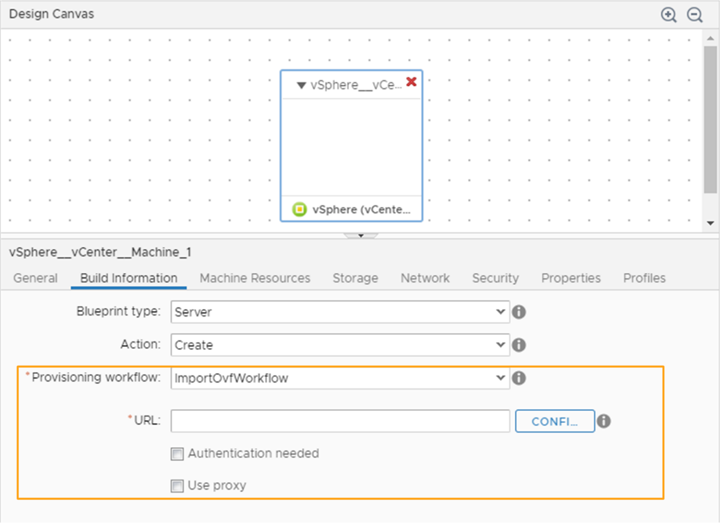
4 Click **Machine Types** in the Categories area and drag a **vSphere (vCenter) Machine** component onto the design canvas.

5 Click the **Build Information** tab and specify the following options:

   - **Blueprint type**: Server

- **Action**: Create
- **Provisioning workflow**: ImportOvfWorkflow

  The `ImportOvfWorkflow` setting allows the **URL** option to become available.



Blueprint type: Server

Action: Create

*Provisioning workflow: ImportOvfWorkflow

*URL:

□ Authentication needed

□ Use proxy

6 Specify the location of the OVF.

- Enter the path to the OVF URL using the format `https://`*`server/folder/name`*`.ovf` or *`name`*`.ova`.

  If you enable authentication with the server that is hosting the OVF, enter the credentials for the authenticating user.

- If the OVF is hosted on a Web site, and you have created a Proxy endpoint to use in accessing the Web site, select **Use proxy** and select the available Proxy endpoint.

7 Click **Configure**.

Note If you receive an authentication error message, the server on which the OVF is hosted requires authentication credentials. If this happens, check the **Authentication needed** check box, enter the **Username** and **Password** credentials required to authenticate with the HTTP server on which the OVF resides, and click **Configure** again.

The Configure option opens a wizard, displaying all the user-configurable properties and values to import from the OVF as custom properties. If there are no configurable properties to import, the pane is empty.

a Use the wizard to either accept the default values to be imported or change those values for the blueprint prior to import.

b Click **OK** to import the properties and values.

All user-configurable properties in the OVF template are imported into the blueprint as editable vRealize Automation custom properties, prefaced with `VMware.Ovf`, while others are imported as hidden properties that are not meant to be edited after import.

8 Click the **Machine Resources** tab to display the results of the OVF import reflected in the minimum value entries for the **CPUs**, **Memory(MB)**, and **Storage(GB)** options.

   You can change any of these values after import.

9 Click the **Storage** tab to display the results of the OVF import.

10 Click the **Properties > Custom Properties** tab sequence to display the results of the OVF import.

   For more information, see Custom Properties for OVF Import.

11 Click **Save**.

What to do next

Continue defining blueprint settings or click **Finish**.

Define an Image Value Set for a Component Profile By Using an OVF

You can import settings from an OVF to create one or more value sets for an image component profile to be used in a parameterized vRealize Automation blueprint.

After you import value set definitions for the `Image` component profile, you can add one or more value sets to the component profile for a vSphere machine component in a blueprint. When a user requests a catalog item, they can select an available `Image` and deploy using parameters that are defined in the image's value set.

When you import the OVF, user-configurable properties and values in the OVF are not imported as custom properties in the value set. If you want to use new custom properties from the imported OVF in relation to the image value set, you must manually define the new custom properties in the vSphere machine component or overall blueprint. The custom properties created in the parameterized blueprint should be applicable to the value set for each component profile image.

Note The OVF custom properties for vRealize Automation are not applicable to OVF custom properties for vSphere. Consider creating one image value set for vRealize Automation and one image value set for vSphere.

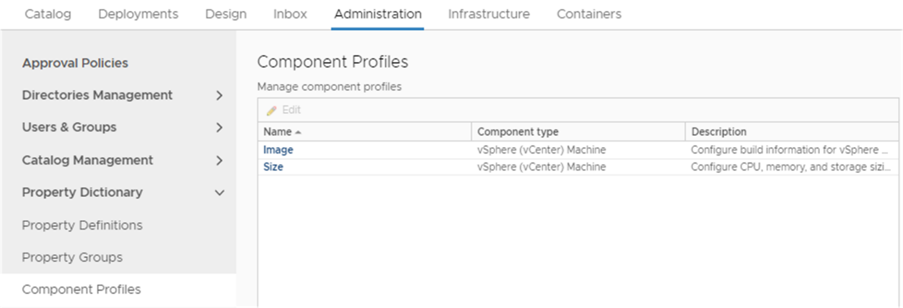
For more information about using component profiles for blueprint parameterization, see Understanding and Using Blueprint Parameterization.

Prerequisites

- Log in to vRealize Automation as an administrator with **tenant administrator** and **IaaS administrator** access rights.

Procedure

1. Select **Administration > Property Dictionary > Component Profiles**.



Component Profiles

Manage component profiles

Name	Component type	Description
Image	vSphere (vCenter) Machine	Configure build information for vSphere ...
Size	vSphere (vCenter) Machine	Configure CPU, memory, and storage sizi...

2. Click **Image** in the Name column.

   Information about the supplied image component property is displayed.

3. Click the **Value Sets** tab.

4. To define a new value set click **New** and configure the `Image` settings.

   a. Enter a value in the **Display name** field to append to the ValueSet delimeter, for example **`ProdOVF`**.

   b. Accept the default value shown in the **Name** text box or enter a custom name.

   c. Enter a description such as **`Build settings for cloning scenario A`** in the **Description** text box.

   d. Select **Active** or **Inactive** in the **Status** drop down menu.

      Select **Active** to allow the value set to be visible in the catalog provisioning request form.

   e. Select the **Create** build action.

   f. Select **Server** or **Desktop** as the blueprint type.

   g. Select the **ImportOvfWorkflow** provisioning workflow.

   h. Enter the path to the OVF URL using the format `https://`*`server/folder/name`*`.ovf` or *`name`*`.ova`.

   i. If you enable authentication with the server that is hosting the OVF, enter the credentials for the authenticating user.

   j. If the OVF is hosted on a Web site, and you have created a Proxy endpoint to use in accessing the Web site, select **Use proxy** and select the available Proxy endpoint.

5. Click **Save**.

6. When you are satisfied with your settings, click **Finish**.

What to do next

After you have created the image and imported the OVF to define the image value set, you can add the image to a vSphere machine component in a blueprint.

Using Container Components in Blueprints

You can configure and use container components in the blueprint.

After a container administrator has created container definitions in Containers for vRealize Automation, a container architect can add and configure container components for vRealize Automation blueprints in the design canvas.

Container Component Settings

You can configure blueprint settings and options for a Containers for vRealize Automation container component in the vRealize Automation design canvas.

General Tab

Configure general settings for the blueprint container component in the design canvas.

Table 5-33. **General** Tab Settings

Setting	Description
Name	Enter a name for your container component in the blueprint.
Description	Summarize your container component for the benefit of other architects.
Image	Enter the full name of an image in a managed registry such as a private registry or Docker Hub registry, for example registry.hub.docker.com/library/python.
Commands	Enter a command that applies to the specified image, such as python app.py. The command is run when the container provisioning process is started.
Links	Links provide another way to connect containers on a single host or across hosts. Enter one or more services to which this container is to be linked, such as redis or datadog.

Network Tab

Configure network settings for the blueprint container component in the design canvas.

You can attach a container to a network. The network is represented as a container network component on the design canvas. Information about available networks is specified in Network page of the container component form.

Table 5-34. **Network** Tab Settings

Setting	Description
Networks	Specify the existing networks that are defined for the selected image. You can also create a new network. When you add a network container component to the design form, the networks that you specify here are listed as available options for selection.
Port bindings	Specify the port bindings for the selected network. Point bindings consist of protocol host, host port, and container port.

Table 5-34. **Network** Tab Settings (continued)

Setting	Description
Publish All Ports	Select the check mark box to expose the ports that are used in the container image to all users.
Host name	Specify the container host name. If no name is specified, the value defaults to the name of the container component in the blueprint.
Network mode	Specify the networking stack of the container. If no value is specified, the container is configured in Bridge network mode.

Storage Tab

Configure storage settings for the blueprint container component in the design canvas.

Table 5-35. **Storage** Tab Settings

Settings	Description
Volumes	Specify the storage volumes that are mapped from the host to be used by the container.
Volumes from	Specify the storage volumes to be inherited from another container.
Working directory	Specify the directory from which to run commands.

Policy Tab

Configure policy settings such as deployment policy and affinity constraints for the blueprint container component in the design canvas.

Table 5-36. **Policy** Tab Settings

Settings	Description
Deployment policy	Specify a deployment policy to set preferences for which set of hosts to use for deploying this container. You can associate deployment policies to hosts, policies, and container definitions to set a preference for hosts, policies, and quotas when deploying a container. You can add a deployment policy by using the **Containers** tab in vRealize Automation.
Cluster size	Specify the number of instances to generate as a cluster from this container.
Restart policy	Specify a restart policy for how a container is restarted on exit.
Max restart	If you selected on-failure as a restart policy, you can specify the maximum number of restarts.
CPU shares	Specify the number of CPU shares allocated for the provisioned resource.

Table 5-36. Policy Tab Settings (continued)

Settings	Description
Memory limit	Specify a number between 0 and the memory available in the placement zone. This is the total memory available for resources in this placement. 0 means no limit.
Memory swap	Total memory limit.
Affinity constraints	Defines rules for provisioning of containers on the same or different hosts. ■ Affinity type For anti-affinity, the containers are placed on different hosts, otherwise they are placed on the same host . ■ Service The service name that is available from the drop-down menu matches the container component name specified in the **Name** field on the **General** tab. ■ Constraint A hard constraint specifies that if the constraint cannot be satisfied, provisioning should fail. A soft constraint specifies that if the constraint cannot be satisfied, provisioning should continue.

Environment Tab

Configure environment settings such as property bindings for the blueprint container component in the design canvas.

Table 5-37. Environment Tab Settings

Setting	Description
Name	The variable name.
Binding	Bind the variable to another property, that is a part of the template. When you select binding, you must input a value in the `_resource~`*`TemplateComponent`*`~`*`TemplateComponentProperty`* syntax.
Value	The value of the environment variable or if you selected binding, the value of the property you want to bind.

Properties Tab

Configure individual and groups of custom properties for the blueprint container component in the design canvas.

For information about the custom properties and property groups that are supplied with the Containers application, see Using Container Properties and Property Groups in a Blueprint.

If you select the **Property Groups** tab and click **Add**, the following options are available:

- Container host properties with certificate authentication

- Container host properties with user/password authentication

If additional property groups have been defined, they are also listed.

If you select the **Custom Properties** tab and click **Add** you can add individual custom properties to the container component.

Table 5-38. **Properties** Tab Settings for Custom Properties

Setting	Description
Name	Enter the name of a custom property or select an available custom property from the drop-down menu.
Value	Enter or edit a value to associate with the custom property name.
Encrypted	You can choose to encrypt the property value, for example, if the value is a password.
Overridable	You can specify that the property value can be overridden by the next or subsequent person who uses the property. Typically, this is another architect, but if you select Show in request, your business users are able to see and edit property values when they request catalog items.
Show in Request	If you want to display the property name and value to your end users, you can select to display the property on the request form when requesting machine provisioning. You must also select **Overridable** if you want users to provide a value.

Health Config Tab

Specify a health configuration mode for the blueprint container component in the design canvas.

Table 5-39. **Health Config** Tab Settings

Mode setting	Description
None	Default. No health checks are configured.
HTTP	If you select **HTTP**, you must provide an API to access and an HTTP method and version to use. The API is relative and you do not need to enter the address of the container. You can also specify a timeout period for the operation and set health thresholds. For example, a healthy threshold of 2 means that two consecutive successful calls must occur for the container to be considered healthy and in the RUNNING status. An unhealthy threshold of 2 means that two unsuccessful calls must occur for the container to be considered unhealthy and in the ERROR status. For all the states in between the healthy and unhealthy thresholds, the container status is DEGRADED.

Table 5-39. **Health Config** Tab Settings (continued)

Mode setting	Description
TCP connection	If you select **TCP connection**, you must only enter a port for the container. The health check attempts to establish a TCP connection with the container on the provided port. You can also specify a timeout value for the operation and set healthy or unhealthy thresholds as with HTTP.
Command	If you select **Command**, you must enter a command to be run on the container. The success of the health check is determined by the exit status of the command.
Ignore health check on provision	Uncheck this option to force health check on provision. By forcing it, a container is not considered provisioned until one successful health check passes.
Autodeploy	Automatic redeployment of containers when they are in ERROR state.

Log Config Tab

Specify a logging mode, and optional logging options, for the blueprint container component in the design canvas.

Table 5-40. **Log Config** Tab Settings

Setting	Description
Driver	Select a logging format from the drop-down menu.
Options	Enter driver options using a name and value format that adheres to the logging format.

Using Container Properties and Property Groups in a Blueprint

You can add predefined property groups to a containers component in a vRealize Automation blueprint. When machines are provisioned by using a blueprint that contain these properties, the provisioned machine is registered as a Docker Container host machine.

Containers for vRealize Automation supplied the following two property groups of container-specific custom properties. When you add a container component to a blueprint you can add these property groups to the container to register provisioned machines as container hosts.

- Container host properties with certificate authentication
- Container host properties with user/password authentication

These property groups are visible in vRealize Automation when you select **Administration > Property Dictionary > Property Groups**.

Because property groups are shared by all tenants, if you are working in a multi-tenant environment, consider cloning and customizing your properties. By uniquely naming property groups and properties in the groups, you can edit them to define custom values for use in a specific tenant.

The most commonly used properties are `Container.Auth.PublicKey` and `Container.Auth.PrivateKey` in which the container administrator provides the client certificate for authenticating with the container host.

Table 5-41. Containers Custom Properties

Property	Description
`containers.ipam.driver`	For use with containers only. Specifies the IPAM driver to be used when adding a Containers network component to a blueprint. The supported values depend on the drivers installed in the container host environment in which they are used. For example, a supported value might be `infoblox` or `calico` depending on the IPAM plug-ins that are installed on the container host.
`containers.network.driver`	For use with containers only. Specifies the network driver to be used when adding a Containers network component to a blueprint. The supported values depend on the drivers installed in the container host environment in which they are used. By default, Docker-supplied network drivers include bridge, overlay, and macvlan, while Virtual Container Host (VCH)-supplied network drivers include the bridge driver. Third-party network drivers such as `weave` and `calico` might also be available, depending on what network plug-ins are installed on the container host.
`Container`	For use with containers only. The default value is `App.Docker` and is required. Do not modify this property.
`Container.Auth.User`	For use with containers only. Specifies the user name for connecting to the Containers host.
`Container.Auth.Password`	For use with containers only. Specifies either the password for the user name or the public or private key password to be used. Encrypted property value is supported.
`Container.Auth.PublicKey`	For use with containers only. Specifies the public key for connecting to the Containers host.
`Container.Auth.PrivateKey`	For use with containers only. Specifies private key for connecting to the Containers host. Encrypted property value is supported.
`Container.Connection.Protocol`	For use with containers only. Specifies the communication protocol. The default value is `API` and is required. Do not modify this property.
`Container.Connection.Scheme`	For use with containers only. Specifies the communication scheme. The default is `https`.
`Container.Connection.Port`	For use with containers only. Specifies the Containers connection port. The default is 2376.

Table 5-41. Containers Custom Properties (continued)

Property	Description
`Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.MachineActivated`	For use with containers only. Specifies the event broker property to expose all Containers properties and is used for registering a provisioned host. The default value is `Container*` and is required. Do not modify this property.
`Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.Disposing`	For use with containers only. Specifies the event broker property to expose all Containersproperties above and is used for unregistering a provisioned host. The default value is `Container*` and is required. Do not modify this property.

Using Containers Network Components in the Design Canvas

You can add one or more Containers network components to the design canvas and configure their settings for vSphere machine components in the blueprint.

You can add the `containers.ipam.driver` and `containers.network.driver` to the component when you add it to the blueprint.

Add a Container Network Component

You can add container network information to a vRealize Automation blueprint that contains container components.

You can configure containers in Containers for vRealize Automation by using the vRealize Automation **Containers** tab. You can add those containers and their network settings as components in a blueprint by using options on the vRealize Automation **Design** tab.

For related information, see Configuring Network Resources for Containers and Using Container Properties and Property Groups in a Blueprint.

Prerequisites

- Log in to vRealize Automation as a **container architect**.
- Open a new or existing blueprint in the design canvas by using the **Design** tab.

Procedure

1. To display the list of available network and security components, click **Network & Security** in the Categories section.
2. Drag a **Container Network** component onto the design canvas.
3. To uniquely label the component in the design canvas, enter a name in the **Name** text box.
4. (Optional) Enter a component description in the **Description** text box.
5. (Optional) Select the **External** check box if you do not want to specify external IPAM settings.

   If you select the **External** check box, the **IPAM Configuration** tab is removed.

6 Click the **IPAM Configuration** tab to specify a new or edit an existing subnet, IP range, and gateway for the network specified in a container component in the blueprint.

IPAM configuration applies to new networks that are created by vRealize Automation as opposed to those that have been previously created in Docker or other supported container application. These settings are not validated and provisioning fails if the settings overlap with other networks. For example, the subnet and gateway must be unique within the container host.

7 Click the **Properties** tab to specify custom properties for the component.

If you select the **Property Groups** tab and click **Add**, the following options are available:

- Container host properties with certificate authentication
- Container host properties with user/password authentication

If additional property groups have been defined, they are also listed.

If you select the **Custom Properties** tab and click **Add** you can add individual custom properties to the container component.

Table 5-42. **Properties** Tab Settings for Custom Properties

Setting	Description
Name	Enter the name of a custom property or select an available custom property from the drop-down menu.
Value	Enter or edit a value to associate with the custom property name.
Encrypted	You can choose to encrypt the property value, for example, if the value is a password.
Overridable	You can specify that the property value can be overridden by the next or subsequent person who uses the property. Typically, this is another architect, but if you select Show in request, your business users are able to see and edit property values when they request catalog items.
Show in Request	If you want to display the property name and value to your end users, you can select to display the property on the request form when requesting machine provisioning. You must also select **Overridable** if you want users to provide a value.

8 To save the blueprint as draft or continue configuring the blueprint, click **Save** or **Finish**.

What to do next

You can add container network settings in the **Network** tab of a container component.

Pushing Container Templates for Use in Blueprints

You can make a container template available for use in a vRealize Automation blueprint.

A container template can include multiple containers. When you push a multi-container template to vRealize Automation, the template is created as a multi-component blueprint in vRealize Automation.

The container-specific properties that you add to the container template are recognized in the vRealize Automation blueprint. See Using Container Properties and Property Groups in a Blueprint.

When you request to provision a blueprint published in the vRealize Automation catalog, you provision the source container application for that blueprint.

You can add other components to the vRealize Automation blueprint, including the following component types:

- Machine types
- Software components
- Other blueprints
- NSX network and security components
- XaaS components
- Custom components

You can push a template from Containers to vRealize Automation. Changes that you make to the vRealize Automation blueprint have no affect on the Containers template.

You can make subsequent changes in the Containers template and push again to overwrite the blueprint in vRealize Automation. Pushing the template to vRealize Automation overwrites the blueprint, and any changes made to the blueprint in vRealize Automation between pushes are lost. To avoid losing blueprint changes, use vRealize CloudClient to clone a new blueprint or to export the blueprint.

Provisioning a Docker Container or Host from a Blueprint

You can create and use vRealize Automation blueprints to provision machines as registered Docker Container hosts.

For a provisioned machine to be registered as a container host, it must meet the following requirements:

- The machine is provisioned by a blueprint that contains Containers-specific custom properties.

  The required container-specific custom properties are supplied in two property groups. See Using Container Properties and Property Groups in a Blueprint.

  For information about using custom properties and property groups in vRealize Automation, see Chapter 8 Custom Properties and the Property Dictionary .
- The machine is accessible over the network.

  For example, the machine must have a valid IP address and be powered on.

You can define a vRealize Automation blueprint to contain specific custom properties that designate a machine as a container host when provisioned using the blueprint.

When a machine with the required blueprint properties is successfully provisioned, it is registered in the Containers and receives events and actions from vRealize Automation.

Creating Microsoft Azure Blueprints and Incorporating Resource Actions

As a cloud or fabric administrator, you can create Microsoft Azure virtual machine blueprints that business group administrators employ as a building block to create customized provisioned machines for consumers. DevOps administrators can also create Azure machine blueprints, or they can use existing Azure machine blueprints when creating composite blueprints.

- Create a Blueprint for Microsoft Azure

  You can create Microsoft Azure virtual machine blueprints that provide access to Azure virtual machine resources.

- Create Azure Custom Resource Actions

  You can create and use custom resource actions to control Azure virtual machines.

Create a Blueprint for Microsoft Azure

You can create Microsoft Azure virtual machine blueprints that provide access to Azure virtual machine resources.

A default Azure Machine template appears in the **Machine Types** category on the vRealize Automation Edit Blueprint page. You can use this virtual machine template as the basis of an Azure blueprint as described in the following procedure. After you create an Azure blueprint, you can publish and deploy it as designed, or you can use it in conjunction with custom Azure resources or with other blueprints to create a composite blueprint.

After creating and publishing the blueprint, users with appropriate privileges can request and provision an Azure instance through the vRealize Automation Service Catalog.

Note that Azure blueprints define virtual machine requirements. vRealize Automation uses these requirements to select the most appropriate reservation for the deployment.

For information about the NSX Settings and Properties tab on the New Blueprint dialog box, see Blueprint Properties Settings.

If you want to create two virtual machines from a single deployment simultaneously, you must create two network interface names and two virtual machine names.

Note Avoid provisioning a deployment to both Azure and vSphere using the same naming prefix, as this can result in duplicate names in Azure and vSphere that may cause problems for some users.

Prerequisites

- Obtain a valid Azure subscription ID and related information including resource group, storage account, and virtual network information that you may need to create a blueprint.

- Configure an Azure endpoint to create a connection to Azure for use with your vRealize Automation deployment.
- Configure Azure reservations as appropriate for your business groups.

Procedure

1 Select **Design > Blueprints**.

2 Click the **New** icon (✚).

3 Enter a blueprint name in the **Name** text box.

The name you enter also populates the **ID** text box. For most cases, you can ignore the **NSX Settings** and **Properties** tabs.

4 Click **OK**.

5 Click **Machine Types** in the Categories menu.

6 Drag the **Azure Machine** virtual machine template to the Design canvas.

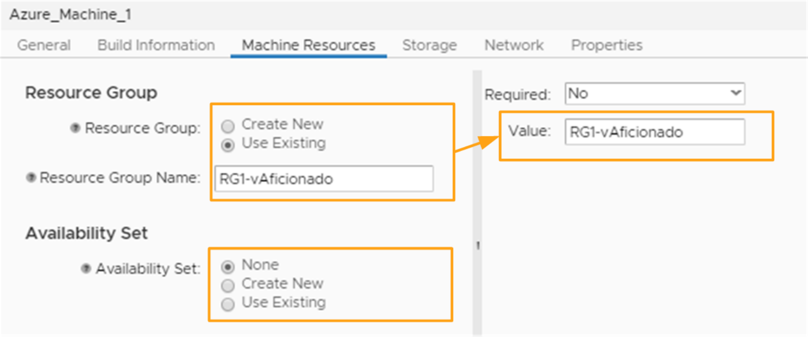
If you created a custom Azure resource for use as the basis of a blueprint, you can select that resource from the assigned category in the Categories list.

7 Enter the required information for the Azure virtual machine in the text boxes on the tabbed pages located on the bottom half of the Design Canvas that appear when you drag the Azure Machine template to the Design Canvas.

Available selections for text boxes and other parameters on all of these tabs are determined primarily by the Azure endpoint that was configured as a basis for blueprints.

For most parameters, when you can click the text box beside the parameter name, a new pane opens on the right side of the page. In this pane, you can enter parameter values in the **Value** text box and indicate whether or not it is **Required**. Note that in some cases you can also enter a **Minimum value** and a **Maximum value**. Click **Apply** within the right pane to populate the initial text box.

Figure 5-1. Azure blueprint right side menu



Resource Group

Resource Group: ○ Create New ● Use Existing

Resource Group Name: RG1-vAficionado

Availability Set

Availability Set: ● None ○ Create New ○ Use Existing

Required: No

Value: RG1-vAficionado

Most parameters also have an **Advanced Options** button. These options enable you to specify parameter lengths and even hide parameters from end users.

Note You must populate required parameters on each tab in order to proceed with the blueprint configuration. If you want to leave a field empty, you can go back and delete the entry before saving.

Tab	Description	Important Parameters
General	Select basic connection information for the Azure virtual machine such as the endpoint to be used.	**ID** - Identifies the Azure virtual machine you are creating. If you change this name, the Azure virtual machine image on the Design Canvas is also updated automatically. **Description** - Identifies the virtual machine you are creating and whether or not it is required. **Instances** - This selection enables you to create a scalable virtual machine. Use the **Minimum** and **Maximum** fields to identity the number of Azure instances that can be spawned from this machine. **Use password authentication**: Select Yes to use password authentication or No to use SSH. **Admin username** - Leave this blank and it can be assigned by the user provisioning the machine. **Admin password** - Leave this field blank, and the individual who provisions the machine can supply the appropriate password,
Build Information	Enables you to configure information about the virtual machine being created.	**Location** - Select the geographical location where this virtual machine will be deployed. **Machine Prefix** - Select the appropriate radio button to indicate whether you want to use the machine prefix from the associated business group or to create a custom prefix. If you want to use a custom prefix, enter it in the **Custom Machine Prefix** text box. **Virtual machine image type** - Choose the appropriate radio button for a **Custom** or **Stock** virtual machine image. A custom virtual machine is created from the Azure classic deployment and offers more configuration options regarding cloud services, storage accounts, and availability sets, **Virtual Machine Image** - Identify the Azure virtual machine image that the blueprint will be based upon. ■ For a stock virtual machine image, the machine image URN should match the following format: (publisher):(offer):(sku):(version). ■ For a managed disk, the machine image URN should match the following format: (ResourceGroupName):(CustomImageName) ■ For a custom virtual machine image, the machine image URN should match the following format: `https://`*`storageaccount`*`.blob.core.windows.net/`*`container`*`/`*`image`*`.vhd` Also you must complete OS Image Type (Windows or Linux) text box for Custom images. **Admin User** - Type the name of the designated admin user configured for virtual machines based on this blueprint. Alternatively, it can be left blank here entered on the request form. **Authentication** - Select the appropriate radio button to indicate whether virtual machines based on this blueprint will require password or SSH authentication.

Tab	Description	Important Parameters
		Admin Password - The administrator password for the virtual machine instance. **Series** - Defines the general size of a virtual machine instance. See the Azure documentation at https://azure.microsoft.com/en-us/documentation/articles/virtual-machines-windows-sizes/ for series information. **Size** - Defines the specific virtual machine instance size within a series. Size is related to the selected Series. If you have a valid connection to an Azure instance, the available sizes fare populated dynamically based on the subscription and selected location and series. See the Azure documentation for size information. **Instance Size Details** - Optional information about the virtual machine instance series and size.
Machine Resources	Organize virtual machine resources into buckets. A resource group is an organizational construct that groups virtual machine resources such as Web sites, accounts, databases and networks. An Availability Set is a mechanism for managing two or more virtual machines to support redundancy. See https://azure.microsoft.com/en-us/documentation/articles/virtual-machines-windows-manage-availability/ for more information about Azure Availability Sets. **Note** If you configure a blueprint with the maximum number of Azure instances set to a value greater than 1, then you should use the existing resource group and availability set rather than create new ones. Using new resource groups or new availability sets on more than one instance in the same deployment will cause errors and other problems if associated with load balancers.	**Create or reuse Resource group:** - Select the appropriate radio button to indicate whether you want to use the existing Azure resource group or create a new one. You can find this name of the existing resource group on the Resource Groups page in the Azure portal. If you choose to create a new resource group, an appropriate name for the new group appears automatically in the **Resource Group** text box. **Create or reuse Availability set:** Select the appropriate radio button depending on what you want to do. If you select Create New, the appropriate information for the new Availability set info appears in the text box.

Tab	Description	Important Parameters
Storage	Enables you to choose either an Azure managed disk or a storage account for this blueprint. With managed disk, Azure handles most of the storage related configuration and maintenance. A storage account provides access to the different types of Azure storage, such as Azure Blob, Queue Table, and File storage. For most blueprints, you can accept the defaults.	**Storage Type** - Select whether you want to provide a managed disk or a manually managed storage account. ■ If you selected Managed Disk, select whether you want to use a premium disk or a standard disk in the **VM Disk Type** box. You can ignore the remaining selection boxes. ■ If you selected Storage Account, enter the storage account name for the virtual machine in the **OS Disk Storage Account** box. The Azure virtual machine operating system disk is deployed to this storage account. You can find storage group information in the Azure portal. You may have one or more storage accounts. **Note** Storage account names with underscores or other special characters may cause errors. **Enable Boot Diagnostics** - Select this check box if you use diagnostic data with your Azure instance. **Number of Data Disks** - Select the appropriate number of data storage disks as used with your virtual machine. You can specify up to four disks. These disks are in addition to the operating system disk as specified in the **Storage account** text box. Storage Disk # ■ **Disk Name** - Identifying name assigned to the disk. ■ **Disk Type** - Storage device type. ■ **Disk Size** - Storage size. ■ **Replication** - Redundancy method used for disk back up. ■ **Host Caching** - Indicates whether read/writes are cached to increase performance.

Tab	Description	Important Parameters
Network	Enables you to select networking for the virtual machine blueprint. For most blueprints, you can accept the defaults and the consumer will enter the appropriate network information during deployment. **Note** You can create only one virtual machine per interface, but each virtual machine can have up to four interfaces.	Click the table to open a dialog to the right that contains another editable table with the following fields. - **Load Balancer Name** - The load balancer used with the Azure instance. - **Number of Network Interfaces** - Select the number of network interfaces used with the Azure instance. The number of network interfaces must be supported by the virtual machine size as selected on the Storage tab. - **Network interface** - Select the appropriate network interface for the virtual machine blueprint. If you enter an existing network, you can ignore all other network tabs. If you enter a network interface name that does not exist, a new interface with that name is created, and you can use the other Network tabs to configure the interface. - **NIC Name Prefix** - The prefix for the network interface card. - **IP Address Type** - Indicate whether the virtual machine uses a static or dynamic IP address. - **Networking Configuration** - Enter the appropriate networking configuration. Network profiles are supported. There are two options, **Specify Azure Networks** and **Use Network Profile**, and the subsequent fields change depending on which option you select. - The following options are available if you select **Specify Azure Networks**. If you leave these text boxes empty, then default network constructs are used based on information specified in the applicable reservation. - **vNet Name** - Name of the virtual network - **subNet Name** - The domain name of the Azure subnet. **Note** You can set the public IP address for Azure during day 2 operations. - If you select **Use Network Profile**, the network configuration is detached from underlying Azure constructs and is instead coupled with the vRealize Automation networking profile. - If you leave the **Network Profile** text box empty, the default Azure vNet and subnet pair are resolved based on applicable reservations which have a network profile specified. - If you enter a network profile, then the Azure vNet and subnet are resolved based on the matching reservation.
Properties	Enables you to add custom properties to your blueprint. Custom properties applied here can be overridden by properties assigned later in the precedence chain. For	There are two options for adding custom properties as represented by two tabs on the Properties dialog. - Property Groups: These are reusable groups that simplify the process of adding custom properties. There are four options for selecting property groups:

Tab	Description	Important Parameters
	more information about order of precedence for custom properties, see Understanding Custom Properties Precedence.	■ **Add** - Enables you to add an available property group to the blueprint. ■ **Move up/Move down** - Enables you to control the precedence of property groups. The first group has the highest priority, and its custom properties take first precedence. ■ **View properties** - Enables you to view the custom properties within the selected group. ■ **View merged properties** - If a custom property is included in more than one property group, the value in the property group with the highest priority takes precedence. Viewing these merged properties can assist you in prioritizing property groups. ■ Custom Properties: Use this tab to add individual custom properties. ■ **New** - Enables you to add an individual custom property to the blueprint. ■ **Name** - Enter a name to identify the property. For a list of custom property names and descriptions, see Chapter 8 Custom Properties and the Property Dictionary . ■ **Value** - Enter a value for the custom property. ■ **Encrypted** - You can encrypt the property. ■ **Overridable** - You can specify that the property value can be overridden by the next or subsequent user. Typically, this is another architect, but if you select Show in request, business users can see and edit property values when they request catalog items. ■ Show in request - If you want to display the property name and value to end users, you can select to display the property on the request form when requesting machine provisioning. You must also select overridable if you want users to provide a value.

8 Click **Finish** to save the blueprint configuration and return to the main Blueprints page.

What to do next

If you have configured custom properties in your Azure reservation to support a VPN tunnel, you can add software components to Azure blueprints.

1 Select **Software Components** on the Categories menu. Software components that you have configured Azure blueprints appear in the pane below.

2 Select Azure Virtual Machine in the container drop-down values.

3 Select the desired software component and drag it to the Azure virtual machine on the Design Canvas.

4 If there are properties required for the software component, enter them in the appropriate parameter text boxes below the Design Canvas.

5 Click **Save**.

If you want to publish the blueprint, select it on the main Blueprints page and click **Publish**. A published blueprint is available on Catalog Items page. Also, a business group manager or equivalent can use this published blueprint as the basis of a composite blueprint.

Create Azure Custom Resource Actions

You can create and use custom resource actions to control Azure virtual machines.

The vRealize Automation Azure implementation is supplied with two custom resource actions out of the box:

- Start virtual machine
- Stop virtual machine

In addition, you can create custom resource actions using workflows that are accessible through vRealize Orchestrator library available from the vRealize Automation interface.

You can work with Azure resource actions just as with any other XaaS resource actions in vRealize Automation. See Designing XaaS Blueprints and Resource Actions and vRealize Orchestrator Integration in vRealize Automation for more information about XaaS resource actions.

Prerequisites

Configure a valid Azure Endpoint for your vRealize Automation deployment.

Procedure

1 Select **Design > XaaS > Resource Actions**

2 Click **New**.

3 Navigate to **Orchestrator > Library > Azure** in thevRealize Orchestrator workflow library.

4 Select the desired folder and workflow.

5 Configure the action for your needs as you would any other XaaS resource action.

Adding Configuration Management Capabilities to vSphere Blueprints

You can add configuration management components to vSphere blueprints to support configuration management of vSphere virtual machines.

vRealize Automation supports the addition of Puppet and Ansible configuration management functionality into vSphere blueprints.

Puppet-based configuration management typically uses roles and environments to define and manage software configuration based on the Puppet Enterprise application. Be aware that the meaning of role and environment in Puppet differs for the more IT generic meaning.

Ansible-based configuration management is based on job templates as defined on an Ansible Tower implementation. You can choose and reorder multiple templates. You can run these templates after a machine is deployed and before it is destroyed from vRealize Automation.

An endpoint establishes a connection with an existing Puppet or Ansible enterprise deployment. When the endpoint is created, vRealize Automation retrieves the appropriate information from the specified deployments. You can specify either early binding or late binding scenarios when configuring a Puppet or Ansible enabled virtual machine blueprint.

Note Ansible and Puppet components are currently supported only on vSphere blueprints and virtual machines.

Add a Puppet Component to a vSphere Blueprint

You can add a Puppet configuration management component to a vSphere blueprint to facilitate enforced management of vSphere virtual machines using a Puppet Master.

Adding a Puppet component to a vSphere blueprint adds a Puppet agent to virtual machines created from that blueprint.

When creating Puppet-enabled vSphere blueprints, you must choose whether to create an early binding or late binding configuration.

With early binding, users define the Puppet role and environment settings for all virtual machines based on a particular blueprint when the Puppet component is added to the blueprint. These settings remain static during the life of the blueprint. For late binding, you have several options.

- Leave the **Puppet environment** and**Puppet role** text boxes empty in the blueprint, and users provide these settings at request time.
- Specify a **Puppet environment** and leave the **Puppet role** box empty. Users must specify the role at request time.

Prerequisites

Create an appropriate vSphere blueprint. See vSphere Machine Component Settings in vRealize Automation for more information.

Procedure

1 Select **Design > Blueprints**.

2 Select **Configuration Management** from the Categories menu on the Design page for blueprints.

3 Select the Puppet component and drag it to the vSphere component on the Design Canvas.

4 Enter an **ID** and **Description** for the Puppet component on the General tab at the bottom of the page.

   The ID and description are arbitrary.

5 Click the Server tab.

6 Click the drop-down and select the appropriate Puppet Master for the blueprint.

7 Select the appropriate **Puppet environment** and **Puppet role** if you want to use early binding for this component.

   To configure early binding, select a Puppet environment and role. If you want to create a component with late binding, select a **Puppet environment**, or leave the **Puppet environment** and **Puppet role** text boxes empty and select the **Set in Request form** check boxes.

   Note The **Set in Request form** check boxes are tied together. If you select one, the other is selected automatically.

8 Click **Finish** to save the Puppet component configuration and return to the main blueprint Design page.

Add an Ansible Component to a vSphere Blueprint

You can add an Ansible configuration management component to a vSphere blueprint to facilitate enforced management of vSphere virtual machines using an Ansible Tower.

Adding an Ansible component to a vSphere blueprint enables the Ansible tower to communicate with deployed resources to run commands.

Prerequisites

Create an appropriate vSphere blueprint. See vSphere Machine Component Settings in vRealize Automation for more information.

Procedure

1 Select **Design > Blueprints**.

2 Select **Configuration Management** from the Categories menu on the Design page for blueprints.

3 Select the Ansible component and drag it to the vSphere component on the Design Canvas.

4 Enter an **ID** and **Description** for the Ansible component on the General tab at the bottom of the page.

   The ID and description are arbitrary.

5 Click the Details tab and enter the appropriate information about the Ansible Tower, project, and template.

   a Select an appropriate **Ansible Tower** and **Organization** that will use this component.

   b Configure either early or late binding for the Ansible component.

      - If you want to use early binding for this component, select the appropriate **Project** and **Job Template**. Select an appropriate template to run when the machine is destroyed in the **Deprovision Job Template** text box. Leave the **Set in Request form** check boxes blank. Also select an appropriate Ansible environment and role.

      - If you want to create a component with late binding, you can choose the **Set in Request form** check boxes in lieu of setting values for the **Project**, **Job Template**, and **Deprovision Job Template** boxes.

        Note The **Set in Request form** check boxes are tied together. If you select one, those below are selected automatically. This functionality occurs because the **Project** field acts as a filter for the Job Templates. If you specify a project, the list of job templates is automatically filtered by project. Therefore, if you choose **Set in Request form** for a project, the following two fields are automatically selected.

6 Click **Finish** to save the Ansible component configuration and return to the main blueprint Design page.

Add RDP Connection Support to Your Windows Machine Blueprints

To allow catalog administrators to entitle users to the Connect using RDP action for Windows blueprints, add RDP custom properties to the blueprint and reference the RDP file that the system administrator prepared.

Note If your fabric administrator creates a property group that contains the required custom properties and you include it in your blueprint, you do not need to individually add the custom properties to the blueprint.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **business group manager**.
- Obtain the name of the custom RDP file that your system administrator created for you. See Create a Custom RDP File to Support RDP Connections for Provisioned Machines .
- Create at least one Windows machine blueprint.

Procedure

1 Select **Design > Blueprints**.

2 Point to the blueprint to update and click **Edit**.

3 Select the machine component on your canvas to edit the details.

4 Click the **Properties** tab.

5 Click the **Custom Properties** tab.

6 Configure RDP settings.

   a Click **New Property**.

   b Enter the RDP custom property names in the **Name** text box and the corresponding values in the **Value** text box.

Option	Description and Value
VirtualMachine.Rdp.File	Specifies an RDP file from which to obtain settings, for example My_RDP_Settings.rdp. The file must reside in the Website\Rdp subdirectory of the vRealize Automation installation directory. For related information, see VirtualMachine.Rdp.File and VirtualMachine.RDP.SettingN in Custom Properties V.
VirtualMachine.Rdp.SettingN	Specifies the RDP settings to be used when opening an RDP link to the machine. *N* is a unique number used to distinguish one RDP setting from another. For example, to specify the RDP authentication level so that no authentication requirement is specified, define the custom property VirtualMachine.Rdp.Setting1 and set the value to authentication level:i:3. For information about available RDP settings, and their correct syntax, see Microsoft Windows RDP documentation such as RDP Settings for Remote Desktop Services in Windows Server. For related information, see VirtualMachine.Rdp.File and VirtualMachine.Rdp.SettingN in Custom Properties R.
VirtualMachine.Admin.NameCompletion	Specifies the domain name to include in the fully qualified domain name of the machine that the RDP or SSH files generate for the user interface options **Connect Using RDP** or **Connect Using SSH** option. For example, set the value to myCompany.com to generate the fully qualified domain name *my-machine-name*.myCompany.com in the RDP or SSH file.

   c Click **Save**.

7 Select the blueprint row and click **Publish**.

Results

Your catalog administrators can entitle users to the **Connect Using RDP** action for machines provisioned from your blueprint. If users are not entitled to the action, they are not able to connect by using RDP.

Add Active Directory Cleanup to Your CentOS Blueprint

As an IaaS architect, you want to configure vRealize Automation to clean up your Active Directory environment whenever provisioned machines are removed from your hypervisors. So you edit your blueprint to configure the Active Directory cleanup plugin.

Using the Active Directory Cleanup Plugin, you can specify the following Active Directory account actions to occur when a machine is deleted from a hypervisor:

- Delete the AD account
- Deactivate the AD account

- Rename AD account
- Move the AD account to another AD organizational unit (OU)

Prerequisites

Note This information does not apply to Amazon Web Services.

- Log in to vRealize Automation as an **infrastructure architect**.
- Gather the following information about your Active Directory environment:
  - An Active Directory account user name and password with sufficient rights to delete, deactivate, rename, or move AD accounts. The user name must be in domain\username format.
  - (Optional) The name of the OU to which to move destroyed machines.
  - (Optional) The prefix to attach to destroyed machines.
- Create a machine blueprint. See Configure a Machine Blueprint.

Procedure

1 Select **Design > Blueprints**.

2 Point to your blueprint and click **Edit**.

3 Select the machine component on your canvas to display the Details tab.

4 Click the **Properties** tab.

5 Click the **Custom properties** tab to configure the Active Directory Cleanup Plugin.

   a Click **New Property**.

   b Type `Plugin.AdMachineCleanup.Execute` in the **Name** text box.

   c Type **`true`** in the **Value** text box.

   d Click the **Save** icon (✓).

6 Configure the Active Directory Cleanup Plugin by adding custom properties.

Option	Description and Value
`Plugin.AdMachineCleanup.UserName`	Enter the Active Directory account user name in the **Value** text box. This user must have sufficient privileges to delete, deactivate, move, and rename Active Directory accounts. The user name must be in the format domain \username.
`Plugin.AdMachineCleanup.Password`	Enter the password for the Active Directory account user name in the **Value** text box.
`Plugin.AdMachineCleanup.Delete`	Set to True to delete the accounts of destroyed machines, instead of disabling them.

Option	Description and Value
`Plugin.AdMachineCleanup.MoveToOu`	Moves the account of destroyed machines to a new Active Directory organizational unit. The value is the organization unit to which you are moving the account. This value must be in *ou=OU, dc=dc* format, for example ou=trash,cn=computers,dc=lab,dc=local.
`Plugin.AdMachineCleanup.RenamePrefix`	Renames the accounts of destroyed machines by adding a prefix. The value is the prefix string to prepend, for example destroyed_.

7 Click **OK**.

Results

Whenever machines provisioned from your blueprint are deleted from your hypervisor, your Active Directory environment is updated.

Allow Requesters to Specify Machine Host Name

As a blueprint architect, you want to allow your users to choose their own machine names when they request your blueprints. So you edit your blueprint to add the Hostname custom property and configure it to prompt users for a value during their requests.

Note If your fabric administrator creates a property group that contains the required custom properties and you include it in your blueprint, you do not need to individually add the custom properties to the blueprint.

Prerequisites

- Log in to vRealize Automation as an **infrastructure architect**.
- Create a machine blueprint. See Configure a Machine Blueprint.

Procedure

1 Select **Design > Blueprints**.

2 Point to your blueprint and click **Edit**.

3 Select the machine component on your canvas to bring up the details tab.

4 Click the **Properties** tab.

5 Click **New Property**.

6 Enter `Hostname` in the **Name** text box.

7 Leave the **Value** text box blank.

8 Configure vRealize Automation to prompt users for a hostname value during request.

   a Select **Overridable**.

   b Select **Show in Request**.

   Because host names must be unique, users can only request one machine at a time from this blueprint.

9 Click the **Save** icon (✓).

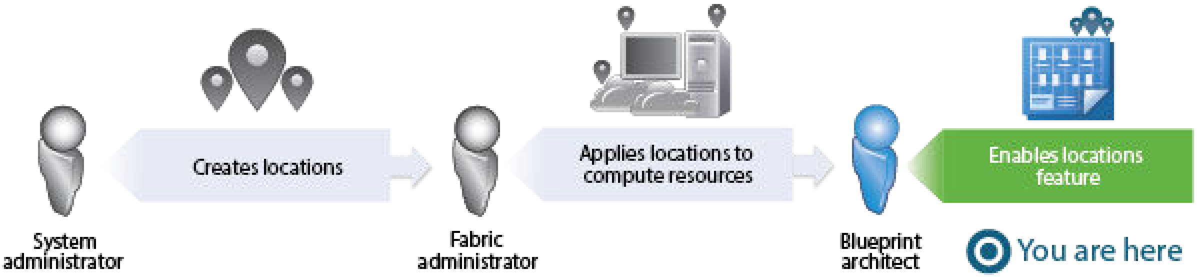
10 Click **OK**.

Results

Users who request a machine from your blueprint are required to specify a host name for their machine. vRealize Automation validates that the specified host name is unique.

Enable Users to Select Datacenter Locations for Cross Region Deployments

As a blueprint architect, you want to allow your users to choose whether to provision machines on your Boston or London infrastructure, so you edit your blueprint to enable the locations feature.



You have a datacenter in London, and a datacenter in Boston, and you don't want users in Boston provisioning machines on your London infrastructure or vice versa. To ensure that Boston users provision on your Boston infrastructure, and London users provision on your London infrastructure, you want to allow users to select an appropriate location for provisioning when they request machines.

Prerequisites

- Log in to vRealize Automation as an **infrastructure architect**.
- As a system administrator, define the datacenter locations. See Scenario: Add Datacenter Locations for Cross Region Deployments .
- As a fabric administrator, apply the appropriate locations to your compute resources. See Scenario: Apply a Location to a Compute Resource for Cross Region Deployments .
- Create a machine blueprint. See Configure a Machine Blueprint.

Procedure

1 Select **Design > Blueprints**.

2 Point to your blueprint and click **Edit**.

3 Select the machine component on your canvas to bring up the **General** details tab.

4 Select the **Display location on request** check box.

5 Click **Finish**.

6 Point to your blueprint and click **Publish**.

Results

Business group users are now prompted to select a datacenter location when they request a machine to be provisioned from your blueprint.

Designing Software Components

As the software architect, you create reusable software components, standardizing configuration properties and using action scripts to specify exactly how components are installed, configured, uninstalled, or updated during deployment scale operations. You can rewrite these action scripts at any time and publish live to push changes to provisioned software components.

You can design your action scripts to be generic and reusable by defining and consuming name and value pairs called software properties and passing them as parameters to your action scripts. If your software properties have values that are unknown or need to be defined in the future, you can either require or allow other blueprint architects or end users to provide the values. If you need a value from another component in a blueprint, for example the IP address of a machine, you can bind your software property to that machine's IP address property. Using software properties to parameterize your action scripts makes them generic and reusable so you can deploy software components on different environments without modifying your scripts.

Table 5-43. Life Cycle Actions

Life Cycle Actions	Description
Install	Install your software. For example, you might download Tomcat server installation bits and install a Tomcat service. Scripts you write for the Install life cycle action run when software is first provisioned, either during an initial deployment request or as part of a scale out.
Configure	Configure your software. For the Tomcat example, you might set the JAVA_OPTS and CATALINA_OPTS. Configuration scripts run after the install action completes.
Start	Start your software. For example, you might start the Tomcat service using the start command in the Tomcat server. Start scripts run after the configure action completes.
Update	If you are designing your software component to support scalable blueprints, handle any updates that are required after a scale in or scale out operation. For example, you might change the cluster size for a scaled deployment and manage the clustered nodes using a load balancer. Design your update scripts to run multiple times (idempotent) and to handle both the scale in and the scale out cases. When a scale operation is performed, update scripts run on all dependent software components.
Uninstall	Uninstall your software. For example, you might perform specific actions in the application before a deployment is destroyed. Uninstall scripts run whenever software components are destroyed.

You can download predefined Software components for a variety of middleware services and applications from the VMware Solution Exchange. Using either the vRealize CloudClient or vRealize Automation REST API, you can programmatically import predefined Software components into your vRealize Automation instance.

- To visit the VMware Solution Exchange, see https://solutionexchange.vmware.com/store/category_groups/cloud-management.
- For information about the vRealize Automation REST API, see *Programming Guide* and vRealize Automation Content Service API at https://code.vmware.com.
- For information about vRealize CloudClient, see https://developercenter.vmware.com/tool/cloudclient.

Property Types and Setting Options

You can design your action scripts to be generic and reusable by defining and consuming name and value pairs called software properties and passing them as parameters to your action scripts. You can create software properties that expect string, array, content, boolean, or integer values. You can supply the value yourself, require someone else to supply the value, or retrieve the value from another blueprint component by creating a binding.

Property Options

You can compute the value of any string property by selecting the computed check box, and you can make any property encrypted, overridable, or required by selecting the appropriate check boxes when you configure Software properties. Combine these options with your values to achieve different purposes. For example, you want to require blueprint architects to supply a value for a password and encrypt that value when they use your software component in a blueprint. Create the password property, but leave the value text box blank. Select Overridable, Required, and Encrypted. If the password you are expecting belongs to your end user, the blueprint architect can select **Show in Request** to require your users to enter the password when they fill out the request form.

Option	Description
Encrypted	Mark properties as encrypted to mask the value and display as asterisks in vRealize Automation. If you change a property from encrypted to unencrypted, vRealize Automation resets the property value. For security, you must set a new value for the property.
Overridable	Allow architects to edit the value of this property when they are assembling an application blueprint. If you enter a value, it displays as a default.

Option	Description
Required	Require architects to provide a value for this property, or to accept the default value you supply.
Computed	Values for computed properties are assigned by the INSTALL, CONFIGURE, START, or UPDATE life cycle scripts. The assigned value is propagated to the subsequent available life cycle stages and to components that bind to these properties in a blueprint. If you select Computed for a property that is not a string property, the property type is changed to string.

If you select the computed property option, leave the value for your custom property blank. Design your scripts for the computed values.

Table 5-44. Scripting Examples for the Computed Property Option

Sample String Property	Script Sytax	Sample Usage
my_unique_id = ""	Bash - `$my_unique_id`	`export my_unique_id="0123456789"`
	Windows CMD - `%my_unique_id%`	`set my_unique_id=0123456789`
	Windows PowerShell - `$my_unique_id`	`$my_unique_id = "0123456789"`

String Property

String properties expect string values. You can supply the string yourself, require someone else to supply the value, or retrieve the value from another blueprint component by creating a binding to another string property. String values can contain any ASCII characters. To create a property binding, use the **Properties** tab on the design canvas to select the appropriate property for binding. The property value is then passed to the action scripts as raw string data. When you bind to a blueprint string property, make sure the blueprint component you bind to is not clusterable. If the component is clustered, the string value becomes an array and you do not retrieve the value you expect.

Sample String Property	Script Syntax	Sample Usage
admin_email = "admin@email987.com"	Bash - `$admin_email`	`echo $admin_email`
	Windows CMD - `%admin_email%`	`echo %admin_email%`
	Windows PowerShell - `$admin_email`	`write-output  $admin_email`

Array Property

Array properties expect an array of string, integer, decimal, or boolean values defined as *["value1", "value2", "value3"...]*. You can supply the values yourself, require someone else to supply the values, or retrieve the values from another blueprint component by creating a property binding.

When you create a software property of type Array, where the data type is integer or decimal, you must use a semicolon as an array element separator, regardless of the locale. Do not use a comma (,) or a dot (.). For some locales, you can use a comma (,) as the decimal separator. For example:

- A valid array for French resembles: [1,11;2,22;3,33]
- A valid array for English resembles: [1.11,2.22,3.33]

When you pass large numbers into an array, do not use the grouping format. For example: do not use **`4444 444.000`** (French), **`4.444.444,000`** (Italian), or **`4,444,444.000`** (English), because data files that contain locale-specific formats might be misinterpreted when they are transferred to a machine that has a different locale. The grouping format is not allowed, because a number such as **`4,444,444.000`** would be considered as three separate numbers. Instead, just enter **`4444444.000`**.

When you define values for an array property you must enclose the array in square brackets. For an array of strings, the value in the array elements can contain any ASCII characters. To properly encode a backslash character in an Array property value, add an extra backslash, for example, `["c:\\test1\\test2"]`. For a bound property, use the **Properties** tab in the design canvas to select the appropriate property for binding. If you bind to an array, you must design your software components so they don't expect a value array in any specific order.

For example, consider a load balancer virtual machine that is balancing the load for a cluster of application server virtual machines. In such a case, an array property is defined for the load balancer service and set to the array of IP addresses of the application server virtual machines.

These load balancer service configure scripts use the array property to configure the appropriate load balancing scheme on the Red Hat, Windows, and Ubuntu operating systems.

Sample Array Property	Script Syntax	Sample Usage

operating_systems = ["Red Hat","Windows","Ubuntu"]	Bash - `${operating_systems[@]}`
	for the entire array of strings
	`${operating_systems[N]}`
	for the individual array element
		`for (( i = 0 ; i < $`
		`{#operating_systems[@]}; i++ )); do`
		`   echo ${operating_systems[$i]}`
		`done`

	Windows CMD - `%operating_systems_N%`
	where *N* represents the position of the element in the array
		`for /F "delims== tokens=2" %%A in`
		`('set operating_systems_') do (`
		`    echo %%A`
		`)`

Sample Array Property	Script Syntax	Sample Usage

	Windows PowerShell - `$operating_systems`
	for the entire array of strings
	`$operating_systems[N]`
	for the individual array element
		`foreach ($os in $operating_systems){`
		`   write-output  $os`
		`}`

Content Property

The content property value is a URL to a file to download content. Software agent downloads the content from the URL to the virtual machine and passes the location of the local file in the virtual machine to the script.

Content properties must be defined as a valid URL with the HTTP or HTTPS protocol. For example, the JBOSS Application Server Software component in the Dukes Bank sample application specifies a content property cheetah_tgz_url. The artifacts are hosted in the Software appliance and the URL points to that location in the appliance. The Software agent downloads the artifacts from the specified location into the deployed virtual machine.

For information about `software.http.proxy` settings that you can use with content properties, see Custom Properties S.

Sample String Property	Script Syntax	Sample Usage
cheetah_tgz_url = "http://*app_content_server_ip:port*/artifacts/software/jboss/cheetah-2.4.4.tar.gz"	Bash - `$cheetah_tgz_url`	`tar -zxvf $cheetah_tgz_url`
	Windows CMD - `%cheetah_tgz_url%`	`start /wait c:\unzip.exe %cheetah_tgz_url%`
	Windows PowerShell - `$cheetah_tgz_url`	`& c:\unzip.exe $cheetah_tgz_url`

Boolean Property

Use the boolean property type to provide True and False choices in the Value drop-down menu.

Integer Property

Use the integer property type for zeros, and positive or negative integers.

Decimal Property

Use the decimal property type for values representing non-repeating decimal fractions.

When Your Software Component Needs Information from Another Component

In several deployment scenarios, a component needs the property value of another component to customize itself. You can do this with vRealize Automation by creating property bindings. You can design your Software action scripts for property bindings, but the actual bindings are configured by the architect that assembles the blueprint.

In addition to setting a property to a hard-coded value, a software architect, IaaS architect, or application architect can bind Software component properties to other properties in the blueprint, such as an IP address or an installation location. When you bind a Software property to another property, you can customize a script based on the value of another component property or virtual machine property. For example, a WAR component might need the installation location of the Apache Tomcat server. In your scripts, you can configure the WAR component to set the server_home property value to the Apache Tomcat server install_path property value in your script. As long as the architect who assembles the blueprint binds the server_home property to the Apache Tomcat server install_path property, then the server_home property value is set correctly.

Your action scripts can only use properties that you define in those scripts, and you can only create property bindings with string and array values. Blueprint property arrays are not returned in any specific order, so binding to clusterable or scalable components might not produce the values you expect. For example, your software component requires each of the machine IDs of a cluster of machines, and you allow your users to request a cluster from 1-10, and to scale the deployment from 1-10 machines. If you configure your software property as a string type, you get a single randomly selected machine ID from the cluster. If you configure your software property as an array type, you get an array of all the machine IDs in the cluster, but in no particular order. If your users scale the deployment, the order of values could be different for each operation. To make sure you never lose values for clustered components, you can use the array type for any software properties. However, you must design your software components so they don't expect a value array in any specific order.

See the Examples of String Property Bindings table for examples of a string property value when binding to different types of properties.

Table 5-45. Examples of String Property Bindings

Sample Property Type	Property Type to Bind	Binding Outcome (A binds to B)
String (property A)	String (property B="Hi")	A="Hi"
String (property A)	Content (property B="http://my.com/content")	A="http://my.com/content"
String (property A)	Array (property B=["1","2"])	A="["1","2"]"
String (property A)	Computed (property B="Hello")	A="Hello"

See the Examples of Array Property Bindings table for examples of an array property value when binding to different types of properties.

Table 5-46. Examples of Array Property Bindings

Sample Property Type	Property Type to Bind	Binding Outcome (A binds to B)
Array (property A)	String (property B="Hi")	A="Hi"
Array (property A)	Content (property B="http://my.com/content")	A="http://my.com/content"
Array (property A)	Computed (property B="Hello")	A="Hello"

For a detailed explanation of supported property types, see Property Types and Setting Options.

Passing Property Values Between Life Cycle Stages

You can modify and pass property values between life cycle stages by using the action scripts.

For a computed property, you can modify the value of a property and pass the value to the next life cycle stage of the action script. For example, if component A has the progress_status value defined as staged, in the INSTALL and CONFIGURE life cycle stage you change the value to progress_status=installed in the respective action scripts. If component B is bound to component A, the property values of progress_status in the life cycle stages of the action script are the same as component A.

Define in the software component that component B depends on A. This dependency defines the passing of correct property values between components whether they are in the same node or across different nodes.

For example, you can update a property value in an action script by using the supported scripts.

- Bash `progress_status="completed"`
- Windows CMD `set progress_status=completed`
- Windows PowerShell `$progress_status="completed"`

Note Array and content property do not support passing modified property values between action scripts of life cycle stages.

Best Practices for Developing Components

To familiarize yourself with best practices for defining properties and action scripts, you can download and import Software components and application blueprints from the VMware Solution Exchange.

Follow these best practices when developing Software components.

- For a script to run without any interruptions, the return value must be set to zero (0). This setting allows the agent to capture all of the properties and send them to the Software server.
- Some installers might need access to the tty console. Redirect the input from `/dev/console`. For example, a RabbitMQ Software component might use the `./rabbitmq_rhel.py --setup-rabbitmq < /dev/console` command in its install script.

- When a component uses multiple life cycle stages, the property value can be changed in the INSTALL life cycle stage. The new value is sent to the next life cycle stage. Action scripts can compute the value of a property during deployment to supply the value to other dependent scripts. For example, in the Clustered Dukes Bank sample application, JBossAppServer service computes the JVM_ROUTE property during the install life cycle stage. This property is used by the JBossAppServer service to configure the life cycle. Apache load balancer service then binds its JVM_ROUTE property to the all (appserver:JbossAppServer:JVM_ROUTE) property to get the final computed value of node0 and node1. If a component requires a property value from another component to complete an application deployment successfully, you must state explicit dependencies in the application blueprint.

  Note You cannot change the content property value for a component that uses multiple life cycle stages.

Create a Software Component

Configure and publish a Software component that other software architects, IaaS architects, and application architects can use to assemble application blueprints.

Prerequisites

Log in to vRealize Automation as a **software architect**.

Procedure

1. Select **Design > Software Components**.
2. Click the **Add** icon (✚).
3. Enter a name and, optionally, a description.

   Using the name you specified for your Software component, vRealize Automation creates an ID for the Software component that is unique within your tenant. You can edit this field now, but after you save the blueprint you can never change it. Because IDs are permanent and unique within your tenant, you can use them to programmatically interact with blueprints and to create property bindings.

4 (Optional) If you want to control how your Software component is included in blueprints, select a container type from the **Container** drop-down menu.

Option	Description
Machines	Your Software component must be placed directly on a machine.
One of your published Software components	If you are designing a Software component specifically to install on top of another Software component that you created, select that Software component from the list. For example, if you are designing an EAR component to install on top of your previously created JBOSS component, select your JBOSS component from the list.
Software components	If you are designing a Software component that should not be installed directly on a machine, but can be installed on several different Software components, then select the software components option. For example, if you are designing a WAR component and you want it to be installed on your Tomcat Server Software component, and your Tcserver Software component, select the software components container type.

5 Click **Next**.

6 Define any properties you intend to use in your action scripts.

a Click the **Add** icon (✚).

b Enter a name for the property.

c Enter a description for the property.

This description displays to architects who use your Software component in blueprints.

d Select the expected type for the value of your property.

e Define the value for your property.

Option	Description
Use the value you supply now	■ Enter a value. ■ Deselect **Overridable**. ■ Select **Required**.
Require architects to supply a value	■ To provide a default, enter a value. ■ Select **Overridable**. ■ Select **Required**.
Allow architects to supply a value if they choose	■ To provide a default, enter a value. ■ Select **Overridable**. ■ Deselect **Required**.

Architects can configure your Software properties to show to users in the request form. Architects can use the Show in Request option to require or request that users fill in values for properties that you mark as overridable.

7 Follow the prompts to provide a script for at least one of the software life cycle actions.

Table 5-47. Life Cycle Actions

Life Cycle Actions	Description
Install	Install your software. For example, you might download Tomcat server installation bits and install a Tomcat service. Scripts you write for the Install life cycle action run when software is first provisioned, either during an initial deployment request or as part of a scale out.
Configure	Configure your software. For the Tomcat example, you might set the JAVA_OPTS and CATALINA_OPTS. Configuration scripts run after the install action completes.
Start	Start your software. For example, you might start the Tomcat service using the start command in the Tomcat server. Start scripts run after the configure action completes.
Update	If you are designing your software component to support scalable blueprints, handle any updates that are required after a scale in or scale out operation. For example, you might change the cluster size for a scaled deployment and manage the clustered nodes using a load balancer. Design your update scripts to run multiple times (idempotent) and to handle both the scale in and the scale out cases. When a scale operation is performed, update scripts run on all dependent software components.
Uninstall	Uninstall your software. For example, you might perform specific actions in the application before a deployment is destroyed. Uninstall scripts run whenever software components are destroyed.

Include exit and status codes in your action scripts. Each supported script type has unique exit and status code requirements.

Script Type	Success Status	Error Status	Unsupported Commands
Bash	■ `return 0` ■ `exit 0`	■ `return non-zero` ■ `exit non-zero`	None
Windows CMD	`exit /b 0`	`exit /b non-zero`	Do not use `exit 0` or `exit non-zero` codes.
PowerShell	`exit 0`	`exit non-zero;`	Do not use `warning`, `verbose`, `debug`, or `host` calls.

8 Select the **Reboot** checkbox for any script that requires you to reboot the machine.

After the script runs, the machine reboots before starting the next life cycle script.

9 Click **Finish**.

10 Select your Software component and click **Publish**.

Results

You configured and published a Software component. Other software architects, IaaS architects, and application architects can use this Software component to add software to application blueprints.

What to do next

Add your published Software component to an application blueprint. See Assembling Composite Blueprints.

Software Component Settings

Configure general settings, create properties, and write custom action scripts to install, configure, update, or uninstall your Software component on provisioned machines.

As a software architect, click **Design > Software components** and click the **Add** icon to create a new Software component.

New Software General Settings

Apply general settings to your Software component.

Table 5-48. New Software General Settings

Setting	Description
Name	Enter a name for your Software component.
ID	Using the name you specified for your Software component, vRealize Automation creates an ID for the Software component that is unique within your tenant. You can edit this field now, but after you save the blueprint you can never change it. Because IDs are permanent and unique within your tenant, you can use them to programmatically interact with blueprints and to create property bindings.
Description	Summarize your Software component for the benefit of other architects.
Container	On the design canvas, blueprint architects can only place your Software component inside the container type you select. ■ Select **Machines** to require architects to place your Software component directly on a machine component in the design canvas. ■ Select **Software components** if you are designing a Software component that should never be placed directly on a machine component, but can be nested inside one of several different Software components. ■ Select a specific published Software component if you are designing a Software component specifically to nest inside another Software component that you created. ■ Select **Azure Virtual Machine** if you are designing a Software component specifically for an Azure blueprint.

New Software Properties

Software component properties are used to parameterize scripts to pass defined properties as environment variables to scripts running in a machine. Before running your scripts, the Software agent in the provisioned machine communicates with vRealize Automation to resolve the properties. The agent then creates script-specific variables from these properties and passes them to the scripts.

Table 5-49. New Software Properties

Setting	Description
Name	Enter a name for your Software property. Property names are case-sensitive and can contain only alphabetic, numeric, hyphen (-), or underscore (_) characters.
Description	For the benefit of other users, summarize your property and any requirements for the value.
Type	Software supports string, array, content, boolean, and integer types. For a detailed explanation of supported property types, see Property Types and Setting Options. For information about property bindings, see When Your Software Component Needs Information from Another Component and Creating Property Bindings Between Blueprint Components.
Value	■ To use the value you supply: ■ Enter a **Value**. ■ Select **Required**. ■ Deselect **Overridable**. ■ To require architects to supply a value: ■ (Optional) Enter a **Value** to provide a default. ■ Select **Overridable**. ■ Select **Required**. ■ Allow architects to supply a value or leave the value blank: ■ (Optional) Enter a **Value** to provide a default. ■ Select **Overridable**. ■ Deselect **Required**.
Encrypted	Mark properties as encrypted to mask the value and display as asterisks in vRealize Automation. If you change a property from encrypted to unencrypted, vRealize Automation resets the property value. For security, you must set a new value for the property. **Important** If secured properties are printed in the script using the echo command or other similar commands, these values appear in plain text in the log files. The values in the log files are not masked.
Overridable	Allow architects to edit the value of this property when they are assembling an application blueprint. If you enter a value, it displays as a default.

Table 5-49. New Software Properties (continued)

Setting	Description
Required	Require architects to provide a value for this property, or to accept the default value you supply.
Computed	Values for computed properties are assigned by the INSTALL, CONFIGURE, START, or UPDATE life cycle scripts. The assigned value is propagated to the subsequent available life cycle stages and to components that bind to these properties in a blueprint. If you select Computed for a property that is not a string property, the property type is changed to string.

New Software Actions

You create Bash, Windows CMD, or PowerShell action scripts to specify exactly how components are installed, configured, uninstalled, or updated during deployment scale operations.

Table 5-50. Life Cycle Actions

Life Cycle Actions	Description
Install	Install your software. For example, you might download Tomcat server installation bits and install a Tomcat service. Scripts you write for the Install life cycle action run when software is first provisioned, either during an initial deployment request or as part of a scale out.
Configure	Configure your software. For the Tomcat example, you might set the JAVA_OPTS and CATALINA_OPTS. Configuration scripts run after the install action completes.
Start	Start your software. For example, you might start the Tomcat service using the start command in the Tomcat server. Start scripts run after the configure action completes.
Update	If you are designing your software component to support scalable blueprints, handle any updates that are required after a scale in or scale out operation. For example, you might change the cluster size for a scaled deployment and manage the clustered nodes using a load balancer. Design your update scripts to run multiple times (idempotent) and to handle both the scale in and the scale out cases. When a scale operation is performed, update scripts run on all dependent software components.
Uninstall	Uninstall your software. For example, you might perform specific actions in the application before a deployment is destroyed. Uninstall scripts run whenever software components are destroyed.

Select the **Reboot** checkbox for any script that requires you to reboot the machine. After the script runs, the machine reboots before starting the next life cycle script. Verify that no processes are prompting for user interaction when the action script is running. Interruptions pause the script, causing it to remain in an idle state indefinitely, eventually failing. Additionally, your scripts must include proper exit codes that are applicable to the application deployment. If the script lacks exit and return codes, the last command that ran in the script becomes the exit status. Exit and return codes vary between the supported script types, Bash, Windows CMD, PowerShell.

Script Type	Success Status	Error Status	Unsupported Commands
Bash	■ `return 0` ■ `exit 0`	■ `return non-zero` ■ `exit non-zero`	None
Windows CMD	`exit /b 0`	`exit /b non-zero`	Do not use `exit 0` or `exit non-zero` codes.
PowerShell	`exit 0`	`exit non-zero;`	Do not use `warning`, `verbose`, `debug`, or `host` calls.

Designing XaaS Blueprints and Resource Actions

The XaaS blueprints can be published as catalog items or used in the blueprint design canvas. The resource actions are actions that you run on deployed items.

XaaS uses vRealize Orchestrator to run workflows that provision items or run actions. For example, you can configure the workflows to create vSphere virtual machines, Active Directory users in groups, or run PowerShell scripts. If you create a custom vRealize Orchestrator workflow, you can provide that workflow as an item in the service catalog so that the entitled users can run the workflow.

You can use an XaaS blueprint as a component in a blueprint that you create in the design canvas, or you can publish it directly to the service catalog.

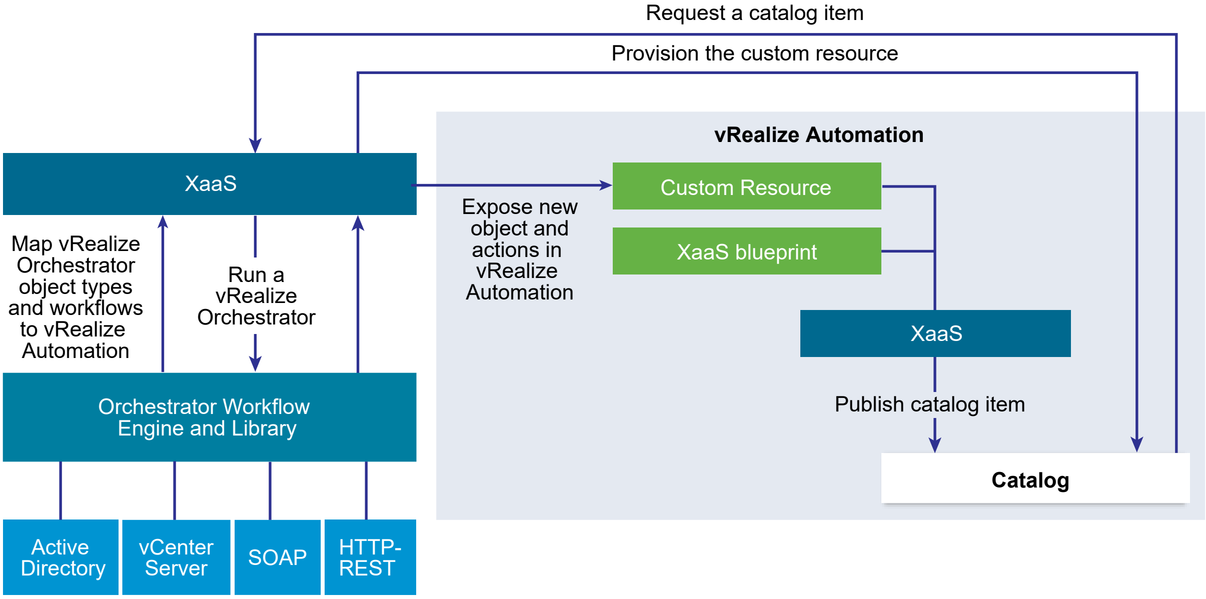
If you use a blueprint as a component in another blueprint, you can configure it to scale when the deployed blueprint is scaled in or out.

vRealize Orchestrator Integration in vRealize Automation

vRealize Orchestrator is the workflow engine integrated in vRealize Automation.

The vRealize Orchestrator server distributed with vRealize Automation is preconfigured, and therefore when your system administrator deploys the vRealize Automation Appliance, the vRealize Orchestrator server is up and running.

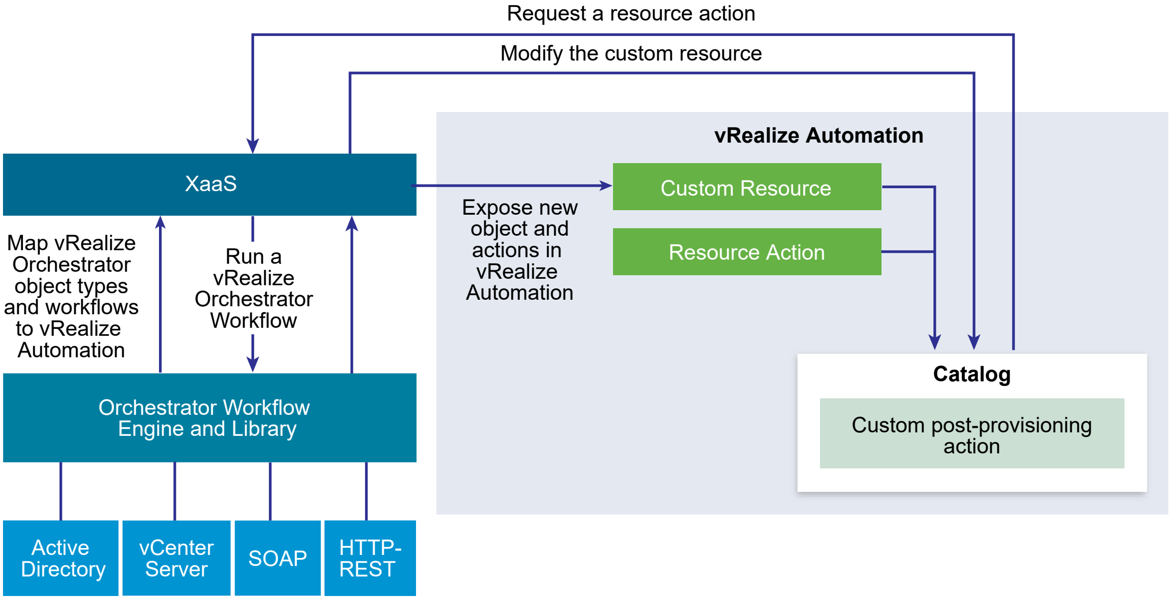
Figure 5-2. Create and Request Catalog Items Included in an XaaS to Provision a Custom Resource


Request a catalog item
Provision the custom resource
vRealize Automation
XaaS
Custom Resource
XaaS blueprint
Expose new object and actions in vRealize Automation
XaaS
Map vRealize Orchestrator object types and workflows to vRealize Automation
Run a vRealize Orchestrator
Publish catalog item
Orchestrator Workflow Engine and Library
Catalog
Active Directory
vCenter Server
SOAP
HTTP-REST

XaaS architects add custom resources related to the supported endpoints and provided workflows, and then create XaaS blueprints and actions based on those resources. Tenant administrators and business group managers can add the XaaS blueprints and actions to the service catalog. The XaaS blueprint can also be used in the blueprint designer.

When the service catalog user requests an item, vRealize Automation runs a vRealize Orchestrator workflow to provision the custom resource.

Figure 5-3. Create and Request Custom Resource Actions to Modify a Custom Resource


Request a resource action
Modify the custom resource
vRealize Automation
XaaS
Custom Resource
Resource Action
Expose new object and actions in vRealize Automation
Map vRealize Orchestrator object types and workflows to vRealize Automation
Run a vRealize Orchestrator Workflow
Catalog
Custom post-provisioning action
Orchestrator Workflow Engine and Library
Active Directory
vCenter Server
SOAP
HTTP-REST

XaaS architects can also add vRealize Orchestrator workflows as resource actions to extend vRealize Automation capabilities. After the service catalog users provision a custom resource, they can run post-provisioning action. This way, the consumers run a vRealize Orchestrator workflow and modify the provisioned custom resource.

When a service catalog user requests an XaaS blueprint or resource action as a catalog item, the XaaS service runs the corresponding vRealize Orchestrator workflow passing the following data as global parameters to the workflow:

Table 5-51. XaaS Global Parameters

Parameter	Description
`__asd_tenantRef`	The tenant of the user requesting the workflow.
`__asd_subtenantRef`	The business group of the user requesting the workflow.
`__asd_catalogRequestId`	The request id from the catalog for this workflow run.
`__asd_requestedFor`	The target user of the request. If the request is on behalf of a user, then this is the user on behalf of whom the workflow is requested, otherwise it is the user requesting the workflow.
`__asd_requestedBy`	The user requesting the workflow.

If an XaaS blueprint or resource action uses a vRealize Orchestrator workflow that contains a User Interaction schema element, when a consumer requests the service, the workflow suspends its run and waits for the user to provide the required data. To answer to a waiting user interaction, the user must navigate to **Inbox > Manual User Action**.

The default vRealize Orchestrator server inventory is shared across all tenants and cannot be used per tenant. For example, if a service architect creates a service blueprint for creating a cluster compute resource, the consumers from different tenants have to browse through the inventory items of all vCenter Server instances although they might belong to a different tenant.

System administrators can install vRealize Orchestrator or deploy the vRealize Orchestrator Applianceseparately to set up an external vRealize Orchestrator instance and configure vRealize Automation to work with that external vRealize Orchestrator instance.

System administrators can also configure vRealize Orchestrator workflow categories per tenant and define which workflows are available to each tenant.

In addition, tenant administrators can also configure an external vRealize Orchestrator instance but only for their own tenants.

For information about configuring an external vRealize Orchestrator instance and vRealize Orchestrator workflow categories, see *Configuring vCenter Orchestrator and Plug-Ins*.

List of vRealize Orchestrator Plug-Ins

With plug-ins you can use vRealize Orchestrator to access and control external technologies and applications. By exposing an external technology in a vRealize Orchestrator plug-in, you can incorporate objects and functions in workflows that access the objects and functions of the external technology.

The external technologies that you can access by using plug-ins can include virtualization management tools, email systems, databases, directory services, remote control interfaces, and so on.

You can use the standard set of vRealize Orchestrator plug-ins to incorporate external technologies such as the vCenter Server API and email capabilities into workflows. In addition, you can use the vRealize Orchestrator open plug-in architecture to develop plug-ins to access other applications.

Table 5-52. Plug-Ins Included by Default in vRealize Orchestrator

Plug-In	Purpose
vCenter Server	Provides access to the vCenter Server API so that you can incorporate all of the vCenter Server objects and functions into the management processes that you automate by using vRealize Orchestrator.
Configuration	Provides workflows for configuring the vRealize Orchestrator authentication, database connection, SSL certificates, and so on.
vCO Library	Provides workflows that act as basic building blocks for customization and automation of client processes. The workflow library includes templates for life cycle management, provisioning, disaster recovery, hot backup, and other standard processes. You can copy and edit the templates to modify them according to your needs.
SQL	Provides the Java Database Connectivity (JDBC) API, which is the industry standard for database-independent connectivity between the Java programming language and a wide range of databases. The databases include SQL databases and other tabular data sources, such as spreadsheets or flat files. The JDBC API provides a call-level API for SQL-based database access from workflows.
SSH	Provides an implementation of the Secure Shell v2 (SSH-2) protocol. Allows remote command and file transfer sessions with password and public key-based authentication in workflows. Supports keyboard-interactive authentication. Optionally, the SSH plug-in can provide remote file system browsing directly in the vRealize Orchestrator client inventory.
XML	A complete Document Object Model (DOM) XML parser that you can implement in workflows. Alternatively, you can use the ECMAScript for XML (E4X) implementation in the vRealize Orchestrator JavaScript API.
Mail	Uses Simple Mail Transfer Protocol (SMTP) to send email from workflows.
Net	Wraps the Jakarta Apache Commons Net Library. Provides implementations of Telnet, FTP, POP3, and IMAP. The POP3 and IMAP part is used for reading email. In combination with the Mail plug-in, the Net plug-in provides complete email send and receive capabilities in workflows.
Enumeration	Provides common enumerated types that can be used in workflows by other plug-ins.
Workflow documentation	Provides workflows that let you generate information in PDF format about a workflow or a workflow category.

Table 5-52. Plug-Ins Included by Default in vRealize Orchestrator (continued)

Plug-In	Purpose
HTTP-REST	Lets you manage REST Web services by providing interaction between vCenter Orchestrator and REST hosts.
SOAP	Lets you manage SOAP Web services by providing interaction between vCenter Orchestrator and SOAP hosts.
AMQP	Lets you interact with Advanced Message Queuing Protocol (AMQP) servers also known as brokers.
SNMP	Enables vCenter Orchestrator to connect and receive information from SNMP-enabled systems and devices.
Active Directory	Provides interaction between vCenter Orchestrator and Microsoft Active Directory.
vCO WebOperator	A Web view that lets you to access the workflows in the vRealize Orchestrator library and interact with them across a network by using a Web browser.
Dynamic Types	Lets you define dynamic types and create and use objects of these dynamic types.
PowerShell	Lets you manage PowerShell hosts and run custom PowerShell operations.
Multi-Node	Contains workflows for hierarchical orchestration, management of Orchestrator instances, and scale-out of Orchestrator activities.
vRealize Automation	Lets you create and run workflows for interaction between vRealize Orchestrator and vRealize Automation.

For more information about the vRealize Orchestrator plug-ins that VMware develops and distributes, see the VMware vRealize ™ Orchestrator ™ Documentation landing page.

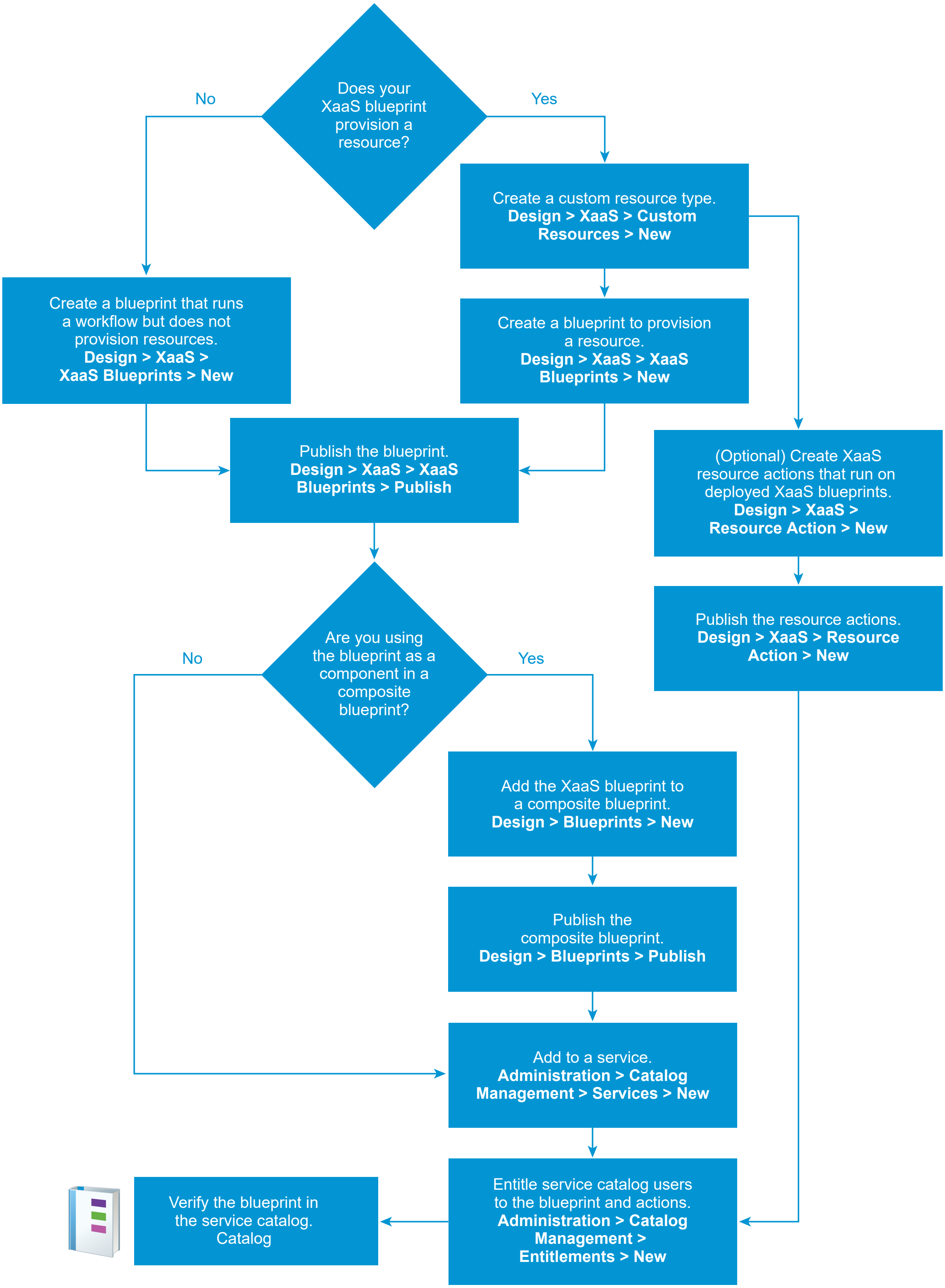
Creating XaaS Blueprints and Resource Actions

The XaaS blueprints can be entitled to users as catalog items, or they can be assembled into a composite blueprints using the design canvas. The resource actions run on the provisioned items to manage the items after they are provisioned.

For example, you can use an XaaS blueprint to create Active Directory users in a group . You can then use a resource action to require that the user change the password.

XaaS Blueprint Workflow

The workflow that you follow to create an XaaS blueprint and any optional resource actions varies depending on how you intend to use the blueprint. The following workflow provides the basic process.


Does your XaaS blueprint provision a resource?
No
Yes
Create a custom resource type.
Design > XaaS > Custom Resources > New
Create a blueprint that runs a workflow but does not provision resources.
Design > XaaS > XaaS Blueprints > New
Create a blueprint to provision a resource.
Design > XaaS > XaaS Blueprints > New
Publish the blueprint.
Design > XaaS > XaaS Blueprints > Publish
(Optional) Create XaaS resource actions that run on deployed XaaS blueprints.
Design > XaaS > Resource Action > New
Publish the resource actions.
Design > XaaS > Resource Action > New
Are you using the blueprint as a component in a composite blueprint?
No
Yes
Add the XaaS blueprint to a composite blueprint.
Design > Blueprints > New
Publish the composite blueprint.
Design > Blueprints > Publish
Add to a service.
Administration > Catalog Management > Services > New
Entitle service catalog users to the blueprint and actions.
Administration > Catalog Management > Entitlements > New
Verify the blueprint in the service catalog.
Catalog

XaaS Blueprint Terminology

XaaS blueprints are vRealize Orchestrator workflows that can provision resources, make changes to provisioned resources, or behave as a service that performs a task in your environment. The blueprints and the resource actions have several nuances that you must understand when you design blueprints for your service catalog users.

The following definitions help you understand the terms used when working with XaaS blueprints.

Custom resource

A vRealize Orchestrator object type that is exposed as a resource through the API of a vRealize Orchestrator plug-in. You create a custom resource to define the output parameter of an XaaS provisioning blueprint and to define an input parameter of a resource action.

XaaS blueprint component

A provisioning or non-provisioning blueprint that you can use in the blueprint design canvas. This blueprint might also be a standalone XaaS blueprint.

Standalone XaaS blueprint

A provisioning or non-provisioning blueprint that is published and entitled directly to the service catalog.

Provisioning blueprint

A provisioning blueprint that runs a vRealize Orchestrator workflow to provision resources on the target endpoint using the vRealize Orchestrator plug-in API for the endpoint. For example, add virtual NICs to a network device in vSphere. To create a provisioning blueprint, you must have a custom resource that defines the vRealize Orchestrator resource type.

When a service catalog user requests this type of catalog items, the workflow provisions the item and the deployed item is stored on the **Deployments** tab. You can define post-provisioning operations for this type of provisioned resources. You can also make the blueprints scalable by adding or removing instance when needed.

Non-provisioning blueprint

A non-provisioning blueprint runs a vRealize Orchestrator workflow to perform a task that does not require the API to make changes to an endpoint. For example, the workflow that runs builds a report and then emails it or posts it to a target communication system.

When a service catalog user requests this type of catalog item, the workflow runs to perform the scripted task, but the item is not added on the **Deployments** tab. You cannot perform post-provisioning operations on this type of blueprint. You can use non-provisioning blueprints as supporting workflows in scalable blueprints. For example, you can create a blueprint to update a high availability load balancer.

Composite blueprint

A blueprint that was created using the design canvas. The composite blueprint uses one or more components. For example, a machine component, a software component, or an XaaS component. When you add it to a service, it is listed as a Deployment. When you add it to an entitlement to make it available to the service catalog users, it is listed as a Composite Blueprint. A composite blueprint can have one blueprint component, or it can include an entire application with multiple machines, software, and networking.

Resource action

A workflow that you can run on a deployed provisioning blueprint. The deployed blueprint can be an XaaS blueprint or blueprint component, or it can be a machine type that you mapped to a vRealize Orchestrator resource type.

XaaS Blueprint Design Considerations

Before you create an XaaS blueprint, you must understand the intent of your blueprint so that you can create one that correctly provisions your resources.

You can create and use XaaS blueprints as a blueprint component in the design canvas or as a standalone blueprint. The blueprint can be a provisioning blueprint or a non-provisioning blueprint.

Table 5-53. XaaS Blueprint Types and Outcomes

XaaS Blueprint Type	Is a custom resource required?	Is the blueprint scalable in a deployment?	Can I run a resource action on the deployed blueprint?
Blueprint component that provisions resources	Yes	Yes. If it is configured to scale, it will scale when the deployment is scaled.	Yes. It scales when the deployment is scaled, and you can run other resource actions on the deployed component. The blueprint component appears on your Deployments tab.
Blueprint component that runs a workflow but does not provision resources	No. The blueprint uses the vRealize Orchestrator server configuration, but it does not require an XaaS custom resource.	No. It does not provision resources, but it can run as part of a scale operation. For example, update a load balancer with the new configuration based on the scale operation.	No. You cannot run a resource action on a non-provisioning component.

Table 5-53. XaaS Blueprint Types and Outcomes (continued)

XaaS Blueprint Type	Is a custom resource required?	Is the blueprint scalable in a deployment?	Can I run a resource action on the deployed blueprint?
Standalone blueprint that provisions resources	Yes	No. You must create resource actions to add or destroy instances.	Yes. You can run resource actions on the deployed resource, including any actions that you created to support scaling. The blueprint appears on your Deployments tab.
Standalone blueprint that runs a workflow but does not provision resources	No. The blueprint uses the vRealize Orchestrator server configuration, but it does not require an XaaS custom resource.	No. It does not provision resources, but it can run as part of a resource action.	No. You cannot run a resource action on a non-provisioning component.

Add an XaaS Custom Resource

You create a custom resource to define the XaaS item for provisioning. Before you can create an XaaS blueprint or action, you must have a custom resource that is compatible with the object type of the blueprint or action workflow.

By creating a custom resource, you map an object type exposed through the API of a vRealize Orchestrator plug-in as a resource. The custom resource defines the output parameter of an XaaS blueprint for provisioning and to define an input parameter of a resource action.

If a blueprint or resource action workflow does not provision a resource or run on a deployed blueprint, you do not need to create a custom resource. For example, you do not need a custom resource if your workflow updates a database value or sends an email message after a provisioning operation.

As you create a custom resource, you can specify the fields of the read-only form on the details of a provisioned item. See Designing a Custom Resource Form.

Prerequisites

- Log in to vRealize Automation as an **XaaS architect**.
- Use the detailed options information to configure the custom resource. See XaaS Custom Resource Wizard Options.

Procedure

1 Select **Design > XaaS > Custom Resources**.

2 Click the **New** icon (✚).

3 Configure the values on the **Resource type** tab.

   a Enter or select the vRealize Orchestrator object type in the **Orchestrator Type** text box.

      For example, enter **v** to see the types that contain the letter v. To see all types, enter a space.

   b Enter a name and, optionally, a description.

   c Enter a version.

      The supported format extends to major.minor.micro-revision.

   d Click **Next**.

4 Edit the **Details Form** tab as needed.

   You can edit the custom resource form by deleting, editing, and rearranging elements. You can also add a form and form pages and drag elements to the new form and form page.

5 Click **Finish**.

Results

You created a custom resource and you can see it on the Custom Resources page. You can create XaaS blueprints or actions based on this custom resource.

What to do next

- Create an XaaS blueprint. See Add an XaaS Blueprint.
- Create an XaaS resource action. See Create an XaaS Resource Action.

XaaS Custom Resource Wizard Options

You use these custom resource options to create or modify a custom resource so that you can run XaaS blueprint and resource action workflows that provision resources or modify provisioned resources.

You can create only one custom resource for an object type. You can use the custom resource for multiple blueprints and resource actions.

To create a custom resource action, select **Design > XaaS > Custom Resources**

Resource Type

The list of possible object types that appears on the **Resource type** tab based on the installed plug-ins in the configured vRealize Orchestrator instance. vRealize Automation collects the values from the configured vRealize Orchestrator instance.

Table 5-54. Resource Type Options

Option	Description
Orchestrator type	Enter or select the type that supports the workflow that you are using to provision. The type is composed of the plug-in name as it appears in the scripting API, for example, VC for vCenter, and the object type, for example, VirtualMachine. In this example, the API uses the value `VC:VirtualMachine`. This type can be the blueprint workflow output parameter or the resource action workflow input parameter.
Name	Enter an informative name for the custom resource so that you can identify it when you create XaaS blueprints or resource actions.
Description	Enter a verbose description.
Version	The supported form extends to major.minor.micro-revision.

Details Form

These form fields appear as read-only values when your service catalog users provision an item that uses this custom resource. You can modify the existing fields and add new externally defined fields.

For more information about configuring the forms, see Designing a Custom Resource Form.

Where Used

Because you can create only one custom resource per object type, you can use this page of the wizard to understand how the custom resource is used.

This tab is available for saved custom resources, not when you create the resource.

Table 5-55. Where Used Options

Option	Description
XaaS Blueprints	A list of the blueprints that are configured to use this custom resource. From this page you can perform the following actions: ■ **Edit**. Opens the blueprint so that you can see how it is configured or to modify it. ■ **Publish/Unpublish**. Change the state of the blueprint by making it available to use in a composite blueprint or to add to a service. If you unpublish a blueprint, you can potentially make it unavailable for use in composite blueprints, to add to a service, or make it unavailable in the service catalog. ■ **Delete**. Remove this blueprint from the system.
Resource Actions	A list of the resource actions that are configured to use this custom resource. From this page you can perform the following actions: ■ **Edit**. Opens the resource action so that you can see how it is configured or modify it. ■ **Publish/Unpublish**. Change the state of the resource action by making it available in an entitlement. If you unpublish a resource action, you can potentially make it unavailable to add to a service, or make it unavailable to run on deployed blueprints. ■ **Delete**. Remove this resource action from the system.

Create an XaaS Blueprint

An XaaS blueprint is a provisioning or non-provisioning blueprint. Some of the provided vRealize Orchestrator provisioning workflows include creating virtual machines, adding users to Active Directory, or taking virtual machine snapshots. Some of the non-provisioning workflows that you might create include updating your load balancer or to building a report and sending it to recipients.

You can create XaaS blueprints based on workflows provided in vRealize Orchestrator or you can use workflows that you create to accomplish goals specific to your environment.

Procedure

1 Add an XaaS Blueprint

An XaaS blueprint is a specification to run a vRealize Orchestrator workflow that makes a change to a target system in your environment. The blueprint includes the workflow, and it can include the input parameters, submission and read-only forms, sequence of actions, and the provisioning or non-provisioning operation.

2 Add an XaaS Blueprint to a Composite Blueprint

You add an XaaS blueprint as a component of a composite blueprint similar to how you add other blueprint components in the design canvas.

Add an XaaS Blueprint

An XaaS blueprint is a specification to run a vRealize Orchestrator workflow that makes a change to a target system in your environment. The blueprint includes the workflow, and it can include the input parameters, submission and read-only forms, sequence of actions, and the provisioning or non-provisioning operation.

You can create XaaS blueprints that you use in one or more of the following ways:

- Create an XaaS blueprint component. A component blueprint is a provisioning or non-provisioning blueprint that you can use in the blueprint design canvas as part of a composite blueprint. If you are using it as a component, you must configure the component life cycle options that support scale-in and scale-out operations on the deployed composite blueprint.

  This blueprint type might also be published as a standalone blueprint.

- Create a standalone XaaS blueprint. A standalone blueprint is a provisioning or non-provisioning blueprint that is published and entitled directly to the service catalog.

For an example of how to create Active Directory users using an XaaS blueprint, see Create an XaaS Blueprint and Action for Creating and Modifying a User .

Prerequisites

- Log in to vRealize Automation as an **XaaS architect**.
- If the blueprint must provision resources, create a custom resource corresponding to the output parameter of the service blueprint. See Add an XaaS Custom Resource. If it does not use a vRealize Orchestrator plug-in API, you do not need to configure a custom resource.
- By creating an XaaS blueprint, you publish a vRealize Orchestrator workflow as a potential component blueprint or catalog item. The blueprint includes a form that you might edit. See Designing an XaaS Blueprint Form.
- Use the detailed options information to configure the blueprint. See XaaS Blueprint New or Edit Wizard Options.

Procedure

1. Select **Design > XaaS > XaaS Blueprints**.
2. Click the **New** icon (✚).
3. On the **Workflow** tab, select the workflow that runs when the blueprint provisions the resource.

   This tab is not available if you are editing a blueprint.

   a. Navigate through the vRealize Orchestrator workflow library and select a workflow relevant to your custom resource.

   b. Review the input and output parameters to ensure that you can later provide the correct values.

   c. Click **Next**.

4 On the **General** tab, configure the options and click **Next**.

   a In the **Name** text box, enter a name that differentiates this blueprint from similar blueprints.

   b If you do not want to use this blueprint as a component in a composite blueprint, deselect the **Make available as a component in the design canvas** check box.

5 On the **Blueprint Form** tab, edit the form as needed and click **Next**.

6 On the **Provisioned Resource** page, select a value and click **Next**.

Option	Description
No provisioning	If the workflow does not provision resources, you can select this option or leave the field empty.
<A custom resource that you previously created>	Select the custom resource that supports this provisioning workflow.

7 On the **Component Lifecycle** tab, define how this blueprint behaves during scale-in, scale-out, and destroy operations.

   These workflows run on a deployed composite blueprint where this blueprint is a component. The availability of the different options depends on blueprint. Not all blueprint workflows support or require all the options.

8 Click **Finish**.

9 Select the row for you blueprint and click **Publish**.

Results

You created and published an XaaS blueprint.

What to do next

- To add this blueprint directly to the service catalog as a standalone blueprint, add a service and add the blueprint to a service. See Add a Service.
- To use this blueprint as a component in a composite blueprint, see Add an XaaS Blueprint to a Composite Blueprint.

XaaS Blueprint New or Edit Wizard Options

You use these options to create an XaaS blueprint that runs a vRealize Orchestrator workflow when the blueprint is deployed. The workflow changes a target system in your environment.

For the steps that you follow to create the blueprint, see Add an XaaS Blueprint.

To use this wizard, select **Design > XaaS > XaaS Blueprints**.

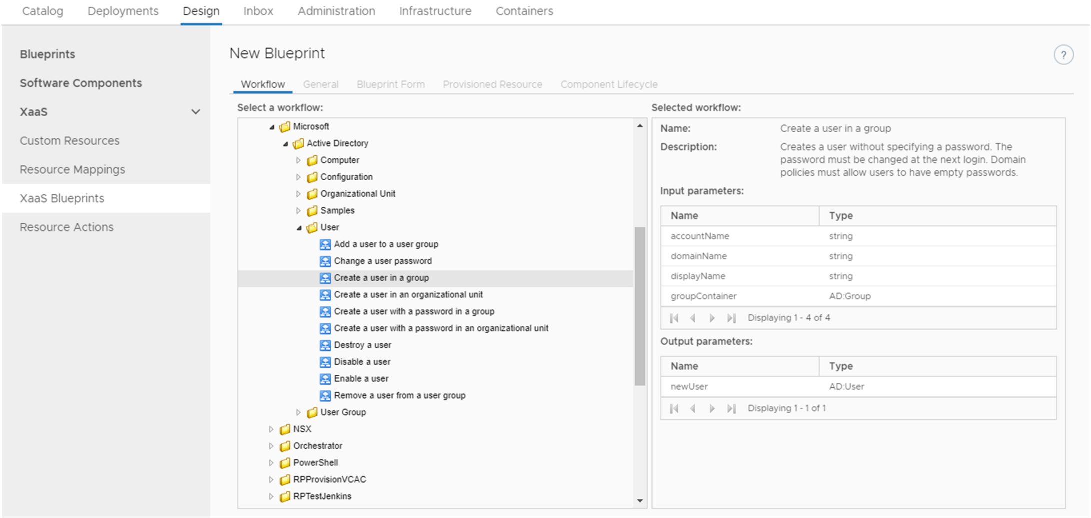
Workflow Tab

Select the workflow that runs when the blueprint provisions the resource.

This tab is not available if you are editing a blueprint.

In the following figure, the workflow tree is on the left and the parameters are on the right.

Figure 5-4. Workflow Tab in the XaaS Blueprint Wizard



New Blueprint

Select a workflow:

- Microsoft
  - Active Directory
    - Computer
    - Configuration
    - Organizational Unit
    - Samples
    - User
      - Add a user to a user group
      - Change a user password
      - Create a user in a group
      - Create a user in an organizational unit
      - Create a user with a password in a group
      - Create a user with a password in an organizational unit
      - Destroy a user
      - Disable a user
      - Enable a user
      - Remove a user from a user group
    - User Group
- NSX
- Orchestrator
- PowerShell
- RPProvisionVCAC
- RPTestJenkins

Selected workflow:

Name: Create a user in a group

Description: Creates a user without specifying a password. The password must be changed at the next login. Domain policies must allow users to have empty passwords.

Input parameters:

Name	Type
accountName	string
domainName	string
displayName	string
groupContainer	AD:Group

Displaying 1 - 4 of 4

Output parameters:

Name	Type
newUser	AD:User

Displaying 1 - 1 of 1

Review the input and output parameters to ensure that you or your service catalog users can provide the correct values under the following circumstances:

- If you customize the blueprint form in this wizard or in the blueprint design canvas.
- If you leave all the input parameters blank, the service catalog users can set the values.

General Tab

Configure the metadata about and the behavior of the blueprint.

Table 5-56. General Tab Options

Option	Description
Name	The name of the blueprint as you want it to appear in the following locations: ■ Design canvas. If you select Make available as a component in the design canvas, this value is the name that appears in the categories list. ■ Services. If you use this blueprint as a standalone blueprint, this value is the name that you see when you add catalog items to service. ■ Entitlements. If you entitle the blueprint as an individual item, this value is the name that you see in the Add Items list.
Description	Provide a verbose description that helps you differentiate between similar items.
Hide catalog request information page	Select the check box when you do not want to require the service catalog consumers to provide a description and reason when they request the item. The check box is selected by default.

Table 5-56. General Tab Options (continued)

Option	Description
Version	The supported format extends to major.minor.micro-revision.
Make available as a component in the design canvas	If you plan to use the blueprint as a component in a design canvas blueprint, select this option. When it is published, the blueprint is available in the category you selected when you configured the custom resource. If you do not select this option, the blueprint does not appear in the design canvas. However, you can still add it to a service and entitle users to deploy it as a standalone blueprint.

Blueprint Form Tab

The fields that appear on this page of the wizard are the workflow input parameters. You can make one or more of the following changes:

- Add fields to the form.
- Modify existing fields by deleting or rearranging the fields.
- Provide default values as the input parameters.

Any changes affect the form that is presented to:

- The application architect working in the design canvas when this XaaS blueprint is used as a blueprint component.
- The service catalog user if this blueprint is published as a standalone blueprint.

For more information about configuring the forms, see Designing an XaaS Blueprint Form.

Provisioned Resource

The provisioned resource links the blueprint to a relevant XaaS custom resource that you configured on the Custom Resource page at **Design > XaaS > Custom Resource** .

Table 5-57. Provisioned Resource Options

Option	Description
A custom resource that you previously created	Select the custom resource that defines the vRealize Orchestrator resource type required to run the provisioning blueprint. A provisioning blueprint runs a vRealize Orchestrator workflow to provision resources on the target endpoint using the vRealize Orchestrator plug-in API for the endpoint. For example, add virtual NICs to a network device in vSphere. You can define post-provisioning operations for this type of provisioned resources. You can also make the blueprint scalable, by adding or removing instances when needed. Results ■ The blueprint is eligible to for scaling. ■ The blueprint appears in the design canvas in the category specified for the selected custom resource. ■ The blueprint is displayed on the **Deployments** tab when you deploy a blueprint that includes it, and you can run any actions on the item after deployment.
No provisioning	A non-provisioning blueprint runs a vRealize Orchestrator workflow to perform a task that does not require the API to make changes to an endpoint. For example, build a report and email or post it to a target communication system. Results ■ The blueprint is not eligible for scaling. You can use non-provisioning blueprints as supporting workflows in scalable blueprints. For example, you can create a blueprint to update a high availability load balancer. ■ The blueprint appears in the XaaS category in the design canvas. ■ The blueprint is not displayed on the **Deployments** tab when you deploy a blueprint that includes it, nor can you run any actions on the item after deployment.

Component Lifecycle Tab

The Component Lifecycle tab is available if you selected **Make available as a component in the design canvas** on the **General** tab.

You use these options to define how this blueprint behaves post-deployment during scale-in and scale-out operations when it is used as a component in a composite blueprint.

The availability of the different options depends on the blueprint. Not all blueprint workflows support or require all the options. Because your XaaS might be used in a composite blueprint, you should configure the update and destroy options, as well as allocate and decallocate, if they are available for the blueprint so that the blueprint scales correctly.

Table 5-58. Component Lifecycle Options

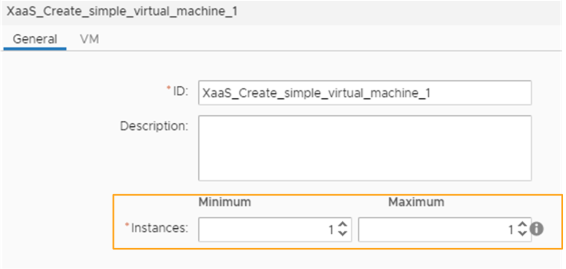
Option	Description
Scalable	Select the option to allow the service catalog user to change the number of instances of this blueprint component after it is deployed as part of a scale-in or scale-out operation. This option is available if you selected a custom resource on the Provisioned Resource tab. It is not available if you selected the No provisioning option. If you make this blueprint scalable, the Instances option is added to the General tab in the design canvas. See the example below. If you do not select Scalable, the Instances option is not available in the design canvas.  XaaS_Create_simple_virtual_machine_1 *ID: XaaS_Create_simple_virtual_machine_1 Description: Minimum 1 Maximum 1 *Instances: 1 1
Provisioning workflow	The workflow that runs during a provisioning or scale-out operation. This workflow was selected when you created this blueprint, and you cannot edit the value.
Allocation workflow	Select the workflow that runs before any initial provisioning or scale-out operation. This life cycle workflow type is available for Azure allocations. If you create an allocation workflow for a scale operation, it must include the following values: ■ Input parameters ■ Parameter name is `requestData` and the parameter type is `Properties`. ■ Parameter name is `subtenant` and the parameter type is `Properties`. ■ `reservations` and the parameter type is `Arrays/Properties`. ■ Output parameter ■ Must include a parameter where the parameter type is `Properties`.

Table 5-58. Component Lifecycle Options (continued)

Option	Description
Update workflow	Select the workflow that runs during update operations, including scale-in or scale out where a component is not scalable, but it can be updated. For example, a load balancer is updated with the new configuration created with the scale-in or scale-out operation for any of the components in the composite blueprint. The update workflow might apply to a component that is bound to the scaled component, but which is not itself scalable. This update workflow can change the non-scalable component based on an update operation. If you create an update workflow for a scale operation, it must include the following values: - Input parameters. - Must include a parameter, regardless of the parameter name, that matches the output parameter type of the provisioning workflow. - Parameter name is `data` and the parameter type is `Properties`.
Destroy workflow	Select the workflow that runs during a scale-in or destroy operation. If you create a destroy workflow for a scale operation, it must include the following value: - Input parameter. - Must include a parameter, regardless of the parameter name, that matches the output parameter type of the provisioning workflow. For example, if the Create simple virtual machine provisioning workflow includes the output parameter VC:VirtualMachine, the destroy workflow must include an input parameter where the type is VC:VirtualMachine.

Table 5-58. Component Lifecycle Options (continued)

Option	Description
Deallocation workflow	Select the workflow that runs after any destroy or scale-in operation. If the deallocation fails during the operation, the destroy workflow still runs as expected. Deallocation is the final process when you scale-in or destroy a composite blueprint. It runs after to the destroy operation, releasing resources. This life cycle workflow type is available for Azure allocations. If you create an deallocation workflow for a scale operation, it must include the following value: ■ Input parameter. ■ Parameter name is `data` and the parameter type is `Properties`.
Category	To specify where the XaaS blueprint appears in the design canvas, select a value in the **Design canvas category** drop-down menu. If you do not select a category, the blueprint is added to the XaaS category when it is published.

Add an XaaS Blueprint to a Composite Blueprint

You add an XaaS blueprint as a component of a composite blueprint similar to how you add other blueprint components in the design canvas.

Use this method to add an XaaS to a composite blueprint. This blueprint can be the only blueprint component or it can be one of several components that make up an application blueprint.

If the XaaS blueprint is all that you want to provide to your users, you can add it to a service and entitle users to it without adding it to a composite blueprint.

If you run a scale-in or scale-out operation on a deployed application blueprint, the XaaS blueprint scales based on how you configured the blueprint life cycle options.

Prerequisites

- Log in to vRealize Automation as an **infrastructure architect**.
- Create and publish an XaaS blueprint. See Create an XaaS Blueprint. When you created the blueprint, you specified the category where the blueprint is located in the design canvas.
- Review how to customize the XaaS blueprint forms in the composite blueprint. See Designing Forms for XaaS Blueprints and Actions.

Procedure

1. Select **Design > Blueprints**.
2. Select the name of the blueprint to which you are adding the XaaS.

   The design canvas appears. It contains the current application component blueprints and other components.
3. In the Categories list, locate the blueprint.

4 Drag your blueprint to the canvas.

5 Configure the default values on the General and Create tabs.

   These are the default values that appear in the service catalog form when a user requests the item.

6 Click **Finish**.

7 Select the blueprint and click **Publish**.

Results

The XaaS blueprint is now part of the composite blueprint.

What to do next

Add the composite blueprint to a service. See Managing the Service Catalog.

Create an XaaS Resource Action

You create a resource action so that you can manage provisioned items using vRealize Orchestrator workflows.

Prerequisites

- Log in to vRealize Automation as an **XaaS architect**.
- Verify that you have a custom resource that support the action. See Add an XaaS Custom Resource.
- If you are creating actions to run on items not provisioned as XaaS catalog items, verify that you mapped the target resources. See Mapping Other Resources to Work with XaaS Resource Actions.

Procedure

1 Create a Resource Action

   A resource action is an XaaS workflow that service catalog users can run on provisioned catalog items. As an XaaS architect, you can create resource actions to define the operations that consumers can perform on the provisioned items.

2 Publish a Resource Action

   The newly created resource action is in draft state, and you must publish the resource action.

3 Assign an Icon to an XaaS Resource Action

   After you create and publish a resource action, you can edit it and assign an icon to the action.

Create a Resource Action

A resource action is an XaaS workflow that service catalog users can run on provisioned catalog items. As an XaaS architect, you can create resource actions to define the operations that consumers can perform on the provisioned items.

By creating a resource action, you associate a vRealize Orchestrator workflow as a post-provisioning operation. During this process, you can edit the default submission and read-only forms. See Designing a Resource Action Form.

Prerequisites

- Log in to vRealize Automation as an **XaaS architect**.
- Create a custom resource corresponding to the input parameter of the resource action.

Procedure

1 Select **Design > XaaS > Resource Actions**.

2 Click the **New** icon (✚).

3 Navigate through the vRealize Orchestrator workflow library and select a workflow relevant to your custom resource.

   You can see the name and description of the selected workflow, and the input and output parameters as they are defined in vRealize Orchestrator.

4 Click **Next**.

5 Select the custom resource that you previously created from the **Resource type** drop-down menu.

6 Select the input parameter for the resource action from the **Input parameter** drop-down menu.

7 Click **Next**.

8 Enter a name and, optionally, a description.

   The **Name** and **Description** text boxes are prepopulated with the name and description of the workflow as they are defined in vRealize Orchestrator.

9 (Optional) If you do not want to prompt consumers to enter a description and reason for requesting this resource action, select the **Hide catalog request information page** check box.

10 Enter a version.

   The supported format extends to major.minor.micro-revision.

11 (Optional) Select the type of the action.

Option	Description
Disposal	The input parameter of the resource action workflow is disposed and the item is removed from the **Deployments** tab. For example, the resource action is for deleting a provisioned machine.
Provisioning	The resource action is for provisioning. For example, the resource action is for copying a catalog item. From the drop-down menu, select an output parameter. You can select a custom resource that you previously created so that when the consumers request this resource action, the provisioned items are added on the **Deployments** tab. If you have only the **No provisioning** option, either the resource action is not for provisioning, or you did not create a proper custom resource for the output parameter, and you cannot proceed.
Provision as child	You can provision a resource as a child of the parent resource. When you are deleting a parent resource or scaling in and scaling out, you have to take care of the child resources first.

Depending on the action workflow, you can select one, both, or none of the options.

12 Select the conditions under which the resource action is available to users, and click **Next**.

13 (Optional) Edit the form of the resource action on the **Form** tab.

The form of the resource action maps the vRealize Orchestrator workflow presentation. You can change the form by deleting, editing, and rearranging the elements. You can also add a new form and form pages and drag the necessary elements to the new form and form page.

Option	Action
Add a form	Click the **New Form** icon (✚) next to the form name, provide the required information, and click **Submit**.
Edit a form	Click the **Edit** icon (✏) next to the form name, make the necessary changes, and click **Submit**.
Regenerate the workflow presentation	Click the **Rebuild** icon (🔄) next to the form name and click **OK**.
Delete a form	Click the **Delete** icon (✖) next to the form name, and in the confirmation dialog box click **OK**.
Add a form page	Click the **New Page** icon (✚) next to the form page name, provide the required information, and click **Submit**.
Edit a form page	Click the **Edit** icon (✏) next to the form page name, make the necessary changes, and click **Submit**.
Delete a form page	Click the **Delete** icon (✖) next to the form name, and in the confirmation dialog box click **OK**.
Add an element to the form page	Drag an element from the New Fields pane on the left to the pane on the right. You can then provide the required information and click **Submit**.

Option	Action
Edit an element	Click the **Edit** icon (✏) next to the element to edit, make the necessary changes, and click **Submit**.
Delete an element	Click the **Delete** icon (✖) next to the element to delete, and in the confirmation dialog box click **OK**.

14 Click **Finish**.

Results

You created a resource action and you can see it listed on the Resource Actions page.

What to do next

Publish the resource action. See Publish a Resource Action.

Publish a Resource Action

The newly created resource action is in draft state, and you must publish the resource action.

Prerequisites

Log in to vRealize Automation as an **XaaS architect**.

Procedure

1 Select **Design > XaaS > Resource Actions**.

2 Select the row of the resource action to publish, and click **Publish**.

Results

The status of the resource action changes to Published.

What to do next

Assign an icon to the resource action. See Assign an Icon to an XaaS Resource Action. Business group managers and tenant administrators can then use the action when they create an entitlement.

Assign an Icon to an XaaS Resource Action

After you create and publish a resource action, you can edit it and assign an icon to the action.

Prerequisites

Log in to vRealize Automation as an **XaaS architect**.

Procedure

1 Select **Administration > Catalog Management > Actions**.

2 Select the resource action that you created.

3 Click **Configure**.

4 Click **Browse** and select the icon to add.

5 Click **Open**.

6 Click **Update**.

Results

You assigned an icon to the resource action. Business group managers and tenant administrators can use the resource action in an entitlement.

Mapping Other Resources to Work with XaaS Resource Actions

You map items that were not provisioned using XaaS so that you can run resource actions to run on those items.

Resource Mapping Script Actions and Workflows

You can use the provided resource mappings for vSphere, vCloud Director, or vCloud Air virtual machines or you can create custom vRealize Orchestrator script actions or workflows to map other vRealize Automation catalog resource types to vRealize Orchestrator inventory types.

Resource Mappings Provided With vRealize Automation

vRealize Automation includes resource mappings for IaaS vSphere virtual machines, IaaS vCloud Director, and deployments.

vRealize Automation includes vRealize Orchestrator resource mapping script actions for each of the provided XaaS resource mappings. Script actions for the provided resource mappings are located in the `com.vmware.vcac.asd.mappings` package of the embedded vRealize Orchestrator server.

When you create a resource action that runs on a deployed composite blueprint that uses a vRealize Orchestrator workflow with `vCACAFE:CatalogResource` as an input parameter, the Deployment mapping is applied as the input resource type. The Deployment mapping is applied only if the selected workflow includes `vCACAFE:CatalogResource` as an input parameter. For example, if you create an action to request a resource action on behalf of a user, the resource type on the Input Resource tab is Deployment because this workflow uses `vCACAFE:CatalogResource`.

The IaaS vCD VM and IaaS VC VirtualMachine resource mappings are used by an action to map the virtual machines that match the IaaS resource to the vRealize Orchestrator vSphere or vCloud Director virtual machine.

Developing Resource Mappings

Depending on your version of vRealize Orchestrator, you can create either a vRealize Orchestrator workflow or a script action to map resources between vRealize Orchestrator and vRealize Automation.

To develop the resource mapping, you use an input parameter of type `Properties`, which contains a key-value pair defining the provisioned resource, and an output parameter of a vRealize Orchestrator inventory type expected by the corresponding vRealize Orchestrator plug-in. The properties available for the mapping depend on the type of resource. For example, the

`EXTERNAL_REFERENCE_ID` property is a common key parameter that defines individual virtual machines, and you can use this property to query a catalog resource. If you are creating a mapping for a resource that does not use an `EXTERNAL_REFERENCE_ID`, you can use one of the other properties that are passed for the individual virtual machines. For example, name, description, and so on.

For more information about developing workflows and script actions, see *Developing with VMware vCenter Orchestrator*.

Create a Resource Mapping

vRealize Automation provides resource mappings for vSphere, vCloud Director, and vCloud Air machines. You can create additional resource mappings for other types of catalog resources.

Prerequisites

- Log in to vRealize Automation as an **XaaS architect**.
- Verify that the mapping script or workflow is available in vRealize Orchestrator. See Resource Mapping Script Actions and Workflows

Procedure

1. Select **Design > XaaS > Resource Mappings**.
2. Click the **New** icon (✚).
3. Enter a name and, optionally, a description.
4. Enter a version.

   The supported format extends to major.minor.micro-revision.
5. Enter the type of the catalog resource in the **Catalog Resource Type** text box and press enter.

   The type of catalog resource appears on the details view of the provisioned item.
6. Enter the vRealize Orchestrator object type in the **Orchestrator Type** text box and press enter.

   This is the output parameter of the resource mapping workflow.

7 (Optional) Add target criteria to restrict the availability of resource actions created by using this resource mapping.

   Resource actions are also subject to restrictions based on approvals and entitlements.

   a Select **Available based on conditions**.

   b Select the type of condition.

Option	Description
All of the following	If all of the clauses you define are satisfied, resource actions created by using this resource mapping are available to the user.
Any of the following	If any one of the clauses you define are satisfied, resource actions created by using this resource mapping are available to the user.
Not the following	If the clause you define exists, resource actions created by using this resource mapping are not available.

   c Follow the prompts to build your clauses and complete the condition.

8 Select your resource mapping script action or workflow from the vRealize Orchestrator library.

9 Click **OK**.

Designing Forms for XaaS Blueprints and Actions

The XaaS includes a form designer that you can use to design submission and details forms for blueprints and resources actions. Based on the presentation of the workflows, the form designer dynamically generates default forms and fields you can use to modify the default forms.

You can create interactive forms that the users can complete for submission of catalog items and resource actions. You can also create read-only forms that define what information the users can see on the details view for a catalog item or a provisioned resource.

As you create XaaS custom resources, XaaS blueprints, and resource actions, forms are generated for common use cases.

Table 5-59. XaaS Object Types and Associated Forms

Object Type	Default Form	Additional Forms
Custom resource	Resource details form based on the attributes of the vRealize Orchestrator plug-in inventory type (read-only).	■ None
XaaS blueprint	Request submission form based on the presentation of the selected workflow.	■ Catalog item details (read-only) ■ Submitted request details (read-only)
Resource action	Action submission form based on the presentation of the selected workflow.	■ Submitted action details (read-only)

You can modify the default forms and design new forms. You can drag fields to add and reorder them on the form. You can place constraints on the values of certain fields, specify default values, or provide instructional text for the end user who is completing the form.

Because of their different purposes, the operations you can perform to design read-only forms are limited compared to the operations for designing submission forms.

Fields in the Form Designer

You can extend the workflow presentation and functionality by adding new predefined fields to the default generated forms of resource actions and XaaS blueprints.

If an input parameter is defined in the vRealize Orchestrator workflow, in vRealize Automation it appears on the default generated form. If you do not want to use the default generated fields in the form, you can delete them and drag and drop new fields from the palette. You can replace default generated fields without breaking the workflow mappings if you use the same ID as the field you are replacing.

You can also add new fields, other than the ones that were generated based on the vRealize Orchestrator workflow inputs, so that you can extend the workflow presentation and functionality in the following cases:

- Add constraints to the existing fields

  For example, you can create a new drop-down menu and name it `dd`. You can also create predefined options of Gold, Silver, Bronze, and Custom. If there is a predefined field, such as CPU, you can add the following constraints to this field:

  - If dd equals Gold, then CPU is 2000 MHz
  - If dd equals Silver, then CPU is 1000 MHz
  - If dd equals Bronze then CPU is 500 MHz
  - If dd equals Custom, the CPU field is editable, and the consumer can specify a custom value

- Add external value definitions to fields

  You can add an external value definition to a field so that you can run vRealize Orchestrator script actions and supply additional information to consumers on the forms you design. For instance, you might want to create a workflow to change the firewall settings of a virtual machine. On the resource action request page, you want to provide the user with the ability to change the open port settings, but you also want to restrict the options to ports that are open. You can add an external value definition to a dual list field and select a custom vRealize Orchestrator script action that queries for open ports. When the request form loads, the script actions runs, and the open ports are presented as options to the user.

- Add new fields that are handled in the vRealize Orchestrator workflow as global parameters

  For instance, the workflow provides an integration with a third-party system and the workflow developer defined input parameters to be handled in the general case, but has also provided a way for passing custom fields. For example, in a scripting box, all global parameters that start with `my3rdparty` are handled. Then, if the XaaS architect wants to pass specific values for consumers to provide, the XaaS architect can add a new field named `my3rdparty_CPU`.

Table 5-60. New Fields in the Resource Action or XaaS Blueprint Form

Field	Description
Text field	Single-line text box
Text area	Multi-line text box
Link	Field in which consumers enter a URL. You can use http, https, ftp, mailto, or /. Do not use file://.
Email	Field in which consumers enter an email address
Password field	Field in which consumers enter a password
Integer field	Text box in which consumers entre an integer You can make this field a slider with a minimum and maximum value, as well as an increment.
Decimal field	Text box in which consumers enter a decimal You can make this field a slider with a minimum and maximum value, as well as an increment.
Date & time	Text boxes in which consumers specify a date (by selecting a date from a calendar menu) and can also select the time (by using up and down arrows)
Dual List	A list builder in which consumers move a predefined set of values between two lists, the first list contains all unselected options and the second list contains the user's choices.
Check box	Check box
Yes/No	Drop-down menu for selecting **Yes** or **No**
Drop-down	Drop-down menu
List	List
Check box list	Check box list
Radio button group	Group of radio buttons
Search	Search text box that auto completes the query and where consumers select an object
Tree	Tree that consumers use to browse and select available objects
Map	Map table that consumers use to define key-value pairs for properties

You can also use the **Section header** form field to split form pages in sections with separate headings and the **Text** form field to add read-only informational texts.

Constraints and Values in the Form Designer

When you edit an element of the blueprint or resource action form, you can apply various constraints and values to the element.

Constraints

The constraints that you can apply to an element vary depending on the type of element you are editing or adding to the form. Some constraint values might be configured in the vRealize Orchestrator workflow. Those values do not appear on the Constraints tab because they are often dependent on conditions that are evaluated when the workflow runs. Any constraint values that you configure for the blueprint form overrides any constraints included in the vRealize Orchestrator workflow.

After calculated for a field, the minimum and maximum bindings are only recalculated when a blueprint is requested.

For each constraint you apply to an element, you can select one of the following options to define the constraint:

Not set

Gets the property from the vRealize Orchestrator workflow presentation.

Constant

Sets the element you are editing to required or optional.

Field

Binds the element to another element from the form. For example, you can set the element to be required only when another element, such as a check box, is selected.

Conditional

Applies a condition. Use the conditions to create various clauses and expressions and apply them to the state or constraints of the element.

External

Select a vRealize Orchestrator script action that defines the value.

Table 5-61. Constraints in the Forms Designer

Constraint	Description
Required	Indicates whether the element is required.
Read only	Indicates whether the field is read-only.
Value	Sets a value for the element.

Table 5-61. Constraints in the Forms Designer (continued)

Constraint	Description
Visible	Indicates whether the consumer can see the element. If you apply a visibility constraint on a display group in the vRealize Orchestrator workflow, the constraint is ignored in the XaaS Submitted Request Details form and the fields that you want hidden appear in the form. To hide fields that you do not want to appear in the Submitted Request Details form, and they are not required for the requesting user, remove the fields from the Submitted Request Details form on the Blueprints Form tab in the XaaS blueprint designer. To locate this tab, see Add a New XaaS Blueprint Form.
Minimum length	Sets a minimum number of characters of the string input element.
Maximum length	Sets a maximum allowed number of characters of the string input element.
Minimum value	Sets a minimum value of the number input element.
Maximum value	Sets a maximum value of the number input element.
Increment	Sets an increment for an element such as a **Decimal** or **Integer** field. For example, when you want an **Integer** field to be rendered as a **Slider**, you can use the value of the step.
Minimum count	Sets a minimum count of items of the element that can be selected. For example, when you add or edit a **Check box list** you can set the minimum number of check boxes that the consumer must select to proceed.
Maximum count	Sets a maximum count of items of the element that can be selected. For example, when you add or edit a **Check box list** you can set the maximum number of check boxes that the consumer must select to proceed.

Values

You can apply values to some of the elements and define what the consumers see for some of the fields. The options available depend on the type of element you are editing or adding to the form.

Table 5-62. Values in the Form Designer

Value	Description
Not set	Get the value of the element you are editing from the vRealize Orchestrator workflow presentation.
Predefined values	Select values from a list of related objects from the vRealize Orchestrator inventory.

Table 5-62. Values in the Form Designer (continued)

Value	Description
Value	Define a static custom value with labels.
External Values	Select a vRealize Orchestrator script action that defines your value with information not directly exposed by the workflow.

External Value Definitions in the Form Designer

When you edit some elements in the forms designer, you can assign external value definitions that use custom vRealize Orchestrator script actions to supply information not directly exposed by the workflow.

For instance, you might want to publish a resource action to install software on a provisioned machine. Instead of providing the consumer with a static list of all software available for download, you can dynamically populate that list with software that is relevant for the machine's operating system, software that the user has not previously installed on the machine, or software that is out of date on the machine and requires an update.

To provide custom dynamic content for your consumer, you create a vRealize Orchestrator script action that retrieves the information you want to display to your consumers. You assign your script action to a field in the form designer as an external value definition. When the resource or service blueprint form is presented to your consumers, the script action retrieves your custom information and displays it to your consumer.

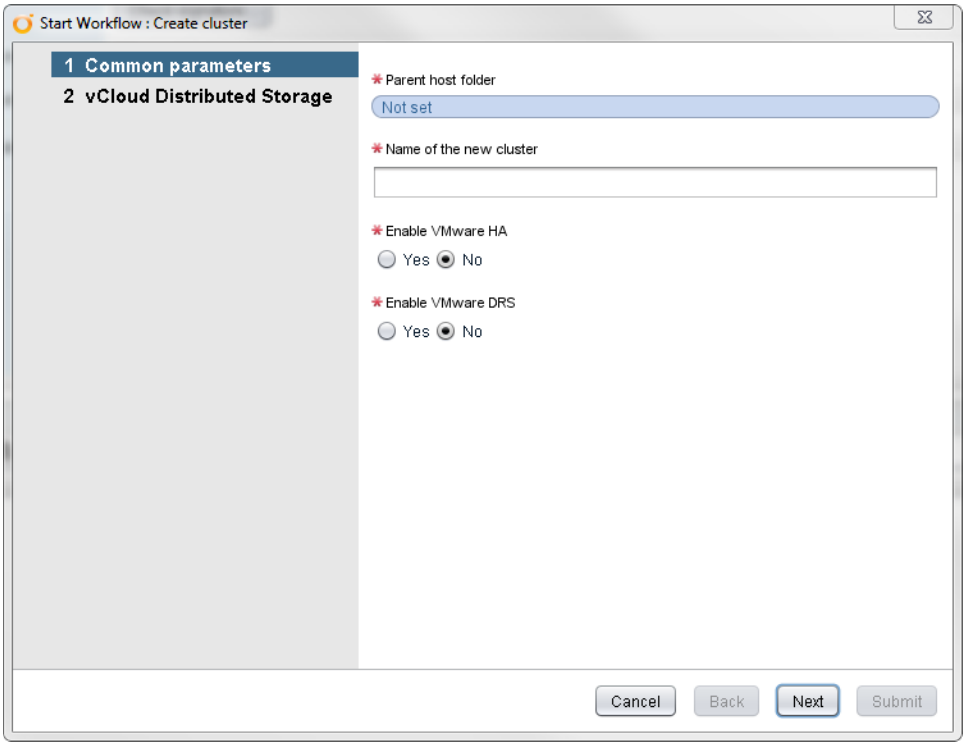
You can use external value definitions to supply default or read-only values, to build boolean expressions, to define constraints, or to provide options for consumers to select from lists, check boxes, and so on.

If you create a blueprint with a workflow that includes a mandatory field, it is mandatory in the request form even if you set it to non-mandatory.

Working With the Form Designer

When you create XaaS blueprints, custom resource actions, and custom resources, you can edit the forms of the blueprints, actions, and resources by using the form designer. You can edit the representation and define what the consumers of the item or action see when they request the catalog item or run the post-provisioning operation.

By default, any XaaS blueprint, resource action, or custom resource form is generated based on the workflow presentation in vRealize Orchestrator.



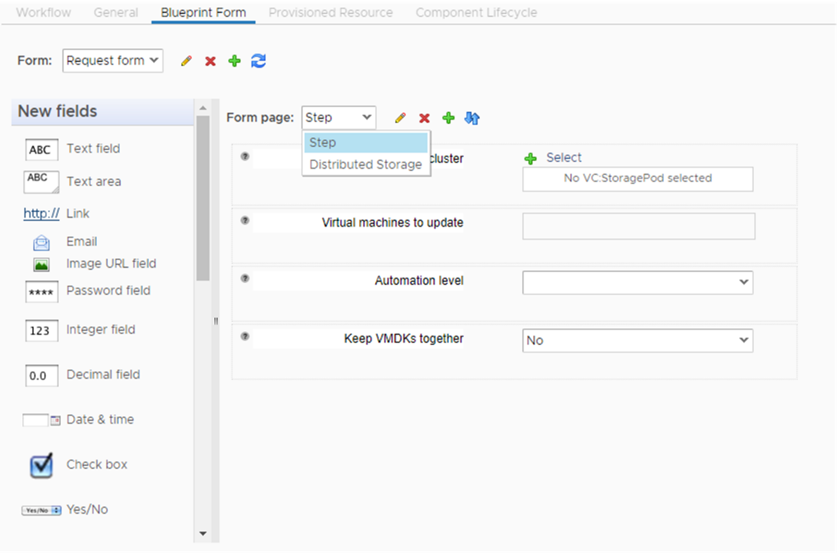
* Parent host folder: Not set

* Name of the new cluster:

* Enable VMware HA: ○ Yes ● No

* Enable VMware DRS: ○ Yes ● No

The steps in the vRealize Orchestrator presentation are represented as form pages and the vRealize Orchestrator presentation groups are represented as separate sections. The input types of the selected workflow are displayed as various fields in the form. For example, the vRealize Orchestrator type `string` is represented by a text box. A complex type such as `VC:VirtualMachine` is represented by a search box or a tree, so that the consumers can type an alphanumeric value to search for a virtual machine or browse to select a virtual machine.



Form: Request form

Form page: Step

cluster: No VC:StoragePod selected

Virtual machines to update:

Automation level:

Keep VMDKs together: No

You can edit how an object is represented in the form designer. For example, you can edit the default `VC:VirtualMachine` representation and make it a tree instead of a search box. You can also add new fields such as check boxes, drop-down menus, and so on, and apply various constraints. If the new fields you add are not valid or are not correctly mapped to the vRealize Orchestrator workflow inputs, when the consumer runs the workflow, vRealize Orchestrator skips the invalid or unmapped fields.

Designing a Custom Resource Form

All fields on the resource details form are displayed as read-only to the consumer on the item details page when they provision your custom resource. You can perform basic edit operations to the form, such as deleting, modifying, or rearranging fields, or you can add new externally defined fields that use vRealize Orchestrator script actions to supply additional read-only information to consumers.

- Edit a Custom Resource Element

  You can edit some of the characteristics of an element on the custom resource Details Form page. Each default field on the page represents a property of the custom resource. You cannot change the type of a property or the default values, but you can edit the name, size, description.

- Add a New Custom Resource Form Page

  You can add a new page to rearrange the form into multiple tabs.

- Insert a Section Header in a Custom Resource Form

  You can insert a section header to split the form into sections.

- Insert a Text Element in a Custom Resource Form

  You can insert a text box to add some descriptive text to the form.

- Insert an Externally Defined Field in a Custom Resource Form

  You can insert a new field and assign it an external value definition to dynamically provide read-only information that consumers can see on the item details page when they provision a custom resource.

Edit a Custom Resource Element

You can edit some of the characteristics of an element on the custom resource Details Form page. Each default field on the page represents a property of the custom resource. You cannot change the type of a property or the default values, but you can edit the name, size, description.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Add an XaaS Custom Resource.

Procedure

1. Select **Design > XaaS > Custom Resources**.
2. Click the custom resource to edit.

3. Click the **Details Form** tab.
4. Point to the element you want to edit and click the **Edit** icon.
5. Enter a new name for the field in the **Label** text box to change the label.
6. Edit the description in the **Description** text box.
7. Select an option from the **Size** drop-down menu to change the size of the element.
8. Select an option from the **Label size** drop-down menu to change the size of the label.
9. Click **Submit**.
10. Click **Finish**.

Add a New Custom Resource Form Page

You can add a new page to rearrange the form into multiple tabs.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Add an XaaS Custom Resource.

Procedure

1. Select **Design > XaaS > Custom Resources**.
2. Click the custom resource to edit.
3. Click the **Details Form** tab.
4. Click the **New Page** icon (✚) next to the **Form page** name.
5. Select the unused screen type and click **Submit**.

   If you already have a resource details or resource list view, you cannot create two of the same type.
6. Click **Submit**.
7. Configure the form.
8. Click **Finish**.

Results

You can delete some of the elements from the original form page and insert them in the new form page, or you can add new fields that use external value definitions to provide information to consumers that is not directly exposed by the vRealize Orchestrator workflow.

Insert a Section Header in a Custom Resource Form

You can insert a section header to split the form into sections.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.

- Add an XaaS Custom Resource.

Procedure

1 Select **Design > XaaS > Custom Resources**.

2 Click the custom resource to edit.

3 Click the **Details Form** tab.

4 Drag the **Section header** element from the Form pane to the Form page pane.

5 Type a name for the section.

6 Click outside of the element to save the changes.

7 Click **Finish**.

Insert a Text Element in a Custom Resource Form
You can insert a text box to add some descriptive text to the form.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Add an XaaS Custom Resource.

Procedure

1 Select **Design > XaaS > Custom Resources**.

2 Click the custom resource to edit.

3 Click the **Details Form** tab.

4 Drag the **Text** element from the Form pane to the Form page pane.

5 Enter the text you want to add.

6 Click outside of the element to save the changes.

7 Click **Finish**.

Insert an Externally Defined Field in a Custom Resource Form
You can insert a new field and assign it an external value definition to dynamically provide read-only information that consumers can see on the item details page when they provision a custom resource.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Add an XaaS Custom Resource.
- Develop or import a vRealize Orchestrator script action to retrieve the information you want to provide to consumers.

Procedure

1. Select **Design > XaaS > Custom Resources**.
2. Click the custom resource to edit.
3. Click the **Details Form** tab.
4. Drag an element from the New Fields pane and drop it to the Form page pane.
5. Enter an ID for the element in the **ID** text box.
6. Enter a label in the **Label** text box.

   Labels appear to consumers on the forms.
7. (Optional) Select a type for the field from the **Type** drop-down menu.
8. Enter the result type of your vRealize Orchestrator script action in the **Entity Type** search box and press Enter.

   For example, if you want to use a script action to display the current user, and the script returns a vRealize Orchestrator result type of `LdapUser`, enter **`LdapUser`** in the **Entity Type** search box and press Enter.
9. Click **Add External Value**.
10. Select your custom vRealize Orchestrator script action.
11. Click **Submit**.
12. Click **Submit** again.
13. Click **Finish**.

Results

When the form is presented to your consumers, the script action retrieves your custom information and displays it to your consumer.

Designing an XaaS Blueprint Form

When you create an XaaS blueprint, you can edit the form of the blueprint by adding new fields to the form, modifying the existing fields, deleting, or rearranging fields. You can also create new forms and form pages, and drag and drop new fields to them.

- Add a New XaaS Blueprint Form

  When you edit the default generated form of a workflow that you want to publish as a XaaS blueprint, you can add a new XaaS blueprint form.
- Edit an XaaS Blueprint Element

  You can edit some of the characteristics of an element on the Blueprint Form page of a XaaS blueprint. You can change the type of an element, its default values, and apply various constraints and values.

- Add a New Element

  When you edit the default generated form of a XaaS blueprint, you can add a predefined new element to the form. For example, if you do not want to use a default generated field, you can delete it and replace it with a new one.

- Insert a Section Header in a XaaS Blueprint Form

  You can insert a section header to split the form into sections.

- Add a Text Element to an XaaS Blueprint Form

  You can insert a text box to add some descriptive text to the form.

Add a New XaaS Blueprint Form

When you edit the default generated form of a workflow that you want to publish as a XaaS blueprint, you can add a new XaaS blueprint form.

By adding a new XaaS blueprint form, you define the look and feel of the catalog item details and submitted request details pages. If you do not add a catalog item details and submitted request details forms, the consumer sees what is defined in the request form.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Add an XaaS Blueprint.

Procedure

1. Select **Design > XaaS > XaaS Blueprints**.
2. Click the XaaS blueprint you want to edit.
3. Click the **Blueprint Form** tab.
4. Click the **New Form** icon (✚).
5. Enter a name and, optionally, a description.
6. Select the screen type from the **Screen type** menu.

Option	Description
Catalog item details	A catalog item details page that consumers see when they click a catalog item.
Request form	The default XaaS blueprint form. The consumers see the request form when they request the catalog item.
Submitted request details	A request details page that consumers see after they request the item and want to view the request details on the **Deployments** tab.

7. Click **Submit**.

What to do next

Add the fields you want by dragging them from the New fields pane to the Form page pane.

Edit an XaaS Blueprint Element

You can edit some of the characteristics of an element on the Blueprint Form page of a XaaS blueprint. You can change the type of an element, its default values, and apply various constraints and values.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Add an XaaS Blueprint.

Procedure

1. Select **Design > XaaS > XaaS Blueprints**.
2. Click the XaaS blueprint you want to edit.
3. Click the **Blueprint Form** tab.
4. Locate the element you want to edit.
5. Click the **Edit** icon (✏).
6. Enter a new name for the field in the **Label** text box to change the label that consumers see.
7. Edit the description in the **Description** text box.
8. Select an option from the **Type** drop-down menu to change the display type of the element.

   The options vary depending on the type of element you edit.
9. Select an option from the **Size** drop-down menu to change the size of the element.
10. Select an option from the **Label size** drop-down menu to change the size of the label.
11. Edit the default value of the element.

Option	Description
Not set	Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.
Constant	Sets the default value of the element you are editing to a constant value that you specify.
Field	Binds the default value of the element to a parameter of another element from the representation.
Conditional	Applies a condition. By using conditions you can create various clauses and expressions and apply them to an element.
External	Select a vRealize Orchestrator script action to define the value.

12 Apply constraints to the element on the **Constraints** tab.

Option	Description
Not set	Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.
Constant	Sets the default value of the element you are editing to a constant value that you specify.
Field	Binds the default value of the element to a parameter of another element from the representation.
Conditional	Applies a condition. By using conditions you can create various clauses and expressions and apply them to an element.
External	Select a vRealize Orchestrator script action to define the value.

13 Add one or more values for the element on the **Values** tab.

The options available depend on the type of element you are editing.

Option	Description
Not set	Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.
Predefined values	Select values from a list of related objects from the vRealize Orchestrator inventory. a Enter a value in the **Predefined values** search box to search the vRealize Orchestrator inventory. b Select a value from the search results and press Enter.
Value	Define custom values with labels. a Enter a value in the **Value** text box. b Enter a label for the value in the **Label** text box. c Click the **Add** icon (➕).
External Values	Select a vRealize Orchestrator script action to define your value with information not directly exposed by the workflow. ■ Select **Add External Value**. ■ Select your vRealize Orchestrator script action. ■ Click **Submit**.

14 Click **Submit**.

15 Click **Finish**.

Add a New Element

When you edit the default generated form of a XaaS blueprint, you can add a predefined new element to the form. For example, if you do not want to use a default generated field, you can delete it and replace it with a new one.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.

- Add an XaaS Blueprint.

Procedure

1. Select **Design > XaaS > XaaS Blueprints**.
2. Click the XaaS blueprint you want to edit.
3. Click the **Blueprint Form** tab.
4. Drag an element from the New Fields pane and drop it to the Form page pane.
5. Enter the ID of a workflow input parameter in the **ID** text box.
6. Enter a label in the **Label** text box.

   Labels appear to consumers on the forms.
7. (Optional) Select a type for the field from the **Type** drop-down menu.
8. Enter a vRealize Orchestrator object in the **Entity type** text box and press Enter.

   This step is not required for all field types.

Option	Description
Result Type	If you are using a script action to define an external value for the field, enter the result type of your vRealize Orchestrator script action.
Input Parameter	If you are using the field to accept consumer input and pass parameters back to vRealize Orchestrator, enter the type for the input parameter accepted by the vRealize Orchestrator workflow.
Output Parameter	If you are using the field to display information to consumers, enter the type for the output parameter of the vRealize Orchestrator workflow.

9. (Optional) Select the **Multiple values** check box to allow consumers to select more than one object.

   This option is not available for all field types.
10. Click **Submit**.
11. Click **Update**.

What to do next

You can edit the element to change the default settings and apply various constraints or values.

Insert a Section Header in a XaaS Blueprint Form
You can insert a section header to split the form into sections.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Add an XaaS Blueprint.

Procedure

1. Select **Design > XaaS > XaaS Blueprints**.
2. Click the XaaS blueprint you want to edit.
3. Click the **Blueprint Form** tab.
4. Drag the **Section header** element from the Form pane to the Form page pane.
5. Type a name for the section.
6. Click outside of the element to save the changes.
7. Click **Update**.

Add a Text Element to an XaaS Blueprint Form

You can insert a text box to add some descriptive text to the form.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Add an XaaS Blueprint.

Procedure

1. Select **Design > XaaS > XaaS Blueprints**.
2. Click the XaaS blueprint you want to edit.
3. Click the **Blueprint Form** tab.
4. Drag the **Text** element from the New Fields pane to the Form page pane.
5. Enter the text you want to add.
6. Click outside of the element to save the changes.
7. Click **Update**.

Designing a Resource Action Form

When you create a resource action, you can edit the form of the action by adding new fields to the form, modifying the existing fields, deleting, or rearranging fields. You can also create new forms and form pages, and drag and drop new fields to them.

Add a New Resource Action Form

When you edit the default generated form of a workflow you want to publish as a resource action, you can add a new resource action form.

By adding a new resource action form, you define how the submitted action details page looks. If you do not add a submitted action details form, the consumer sees what is defined in the action form.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Create a Resource Action.

Procedure

1. Select **Design > XaaS > Resource Actions**.
2. Click the resource action you want to edit.
3. Click the **Form** tab.
4. Click the **New Form** icon (✚).
5. Enter a name and, optionally, a description.
6. Select the screen type from the **Screen type** menu.

Option	Description
Action form	The default resource action form that consumers see when they decide to run the post-provisioning action.
Submitted action details	A request details page that consumers see when they request the action and decide to view the request details on the **Deployments** tab.

7. Click **Submit**.

What to do next

Add the fields you want by dragging them from the New fields pane to the Form page pane.

Add a New Element to a Resource Action Form

When you edit the default generated form of a resource action, you can add a predefined new element to the form. For example, if you do not want to use a default generated field, you can delete it and replace it with a new one.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Create a Resource Action.

Procedure

1. Select **Design > XaaS > Resource Actions**.
2. Click the resource action you want to edit.
3. Click the **Form** tab.
4. Drag an element from the New Fields pane and drop it to the Form page pane.
5. Enter the ID of a workflow input parameter in the **ID** text box.
6. Enter a label in the **Label** text box.

   Labels appear to consumers on the forms.
7. (Optional) Select a type for the field from the **Type** drop-down menu.

8. Enter a vRealize Orchestrator object in the **Entity type** text box and press Enter.

   This step is not required for all field types.

Option	Description
Result Type	If you are using a script action to define an external value for the field, enter the result type of your vRealize Orchestrator script action.
Input Parameter	If you are using the field to accept consumer input and pass parameters back to vRealize Orchestrator, enter the type for the input parameter accepted by the vRealize Orchestrator workflow.
Output Parameter	If you are using the field to display information to consumers, enter the type for the output parameter of the vRealize Orchestrator workflow.

9. (Optional) Select the **Multiple values** check box to allow consumers to select more than one object.

   This option is not available for all field types.

10. Click **Submit**.

11. Click **Finish**.

What to do next

You can edit the element to change the default settings and apply various constraints or values.

Edit a Resource Action Element

You can edit some of the characteristics of an element on the resource action Form page. You can change the type of an element, its default values, and apply various constraints and values.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Create a Resource Action.

Procedure

1. Select **Design > XaaS > Resource Actions**.
2. Click the resource action you want to edit.
3. Click the **Form** tab.
4. Locate the element you want to edit.
5. Click the **Edit** icon (✏).
6. Enter a new name for the field in the **Label** text box to change the label that consumers see.
7. Edit the description in the **Description** text box.
8. Select an option from the **Type** drop-down menu to change the display type of the element.

   The options vary depending on the type of element you edit.

9 Select an option from the **Size** drop-down menu to change the size of the element.

10 Select an option from the **Label size** drop-down menu to change the size of the label.

11 Edit the default value of the element.

Option	Description
Not set	Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.
Constant	Sets the default value of the element you are editing to a constant value that you specify.
Field	Binds the default value of the element to a parameter of another element from the representation.
Conditional	Applies a condition. By using conditions you can create various clauses and expressions and apply them to an element.
External	Select a vRealize Orchestrator script action to define the value.

12 Apply constraints to the element on the **Constraints** tab.

Option	Description
Not set	Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.
Constant	Sets the default value of the element you are editing to a constant value that you specify.
Field	Binds the default value of the element to a parameter of another element from the representation.
Conditional	Applies a condition. By using conditions you can create various clauses and expressions and apply them to an element.
External	Select a vRealize Orchestrator script action to define the value.

13 Add one or more values for the element on the **Values** tab.

The options available depend on the type of element you are editing.

Option	Description
Not set	Gets the value of the element you are editing from the vRealize Orchestrator workflow presentation.
Predefined values	Select values from a list of related objects from the vRealize Orchestrator inventory. a Enter a value in the **Predefined values** search box to search the vRealize Orchestrator inventory. b Select a value from the search results and press Enter.

Option	Description
Value	Define custom values with labels. a Enter a value in the **Value** text box. b Enter a label for the value in the **Label** text box. c Click the **Add** icon (✚).
External Values	Select a vRealize Orchestrator script action to define your value with information not directly exposed by the workflow. ■ Select **Add External Value**. ■ Select your vRealize Orchestrator script action. ■ Click **Submit**.

14 Click **Submit**.

15 Click **Update**.

Insert a Section Header in a Resource Action Form

You can insert a section header to split the form into sections.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Create a Resource Action.

Procedure

1 Select **Design > XaaS > Resource Actions**.

2 Click the resource action you want to edit.

3 Click the **Form** tab.

4 Drag the **Section header** element from the Form pane to the Form page pane.

5 Type a name for the section.

6 Click outside of the element to save the changes.

7 Click **Finish**.

Add a Text Element to a Resource Action Form

You can insert a text box to add some descriptive text to the form.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **XaaS architect**.
- Create a Resource Action.

Procedure

1 Select **Design > XaaS > Resource Actions**.

2 Click the resource action you want to edit.

3 Click the **Form** tab.

4 Drag the **Text** element from the New Fields pane to the Form page pane.

5 Enter the text you want to add.

6 Click outside of the element to save the changes.

7 Click **Finish**.

XaaS Examples and Scenarios

The examples and scenarios suggest ways that you can use vRealize Automation to accomplish common tasks using XaaS blueprints and resource actions.

Create an XaaS Blueprint and Action for Creating and Modifying a User

By using XaaS, you can create and publish a catalog item for provisioning a user in a group. You can also associate a new post-provisioning operation to the provisioned user. For example, an operation so that the service catalog users can change the user password.

As an XaaS architect, you create a custom resource, an XaaS blueprint, and publish a catalog item for creating a user. You also create a resource action for changing the password of the user.

As a catalog administrator, you create a service and include the blueprint catalog item in the service. In addition, you edit the workflow presentation of the catalog item by using the form designer and change the way the consumers see the request form.

As a business group manager or a tenant administrator, you entitle the newly created service, catalog item, and resource action to a consumer.

Prerequisites

Verify that the Active Directory plug-in is properly configured and you have the rights to create users in Active Directory.

Procedure

1 Create a Test User as a Custom Resource

You can create a custom resource and map it to the vRealize Orchestrator object type `AD:User`.

2 Create an XaaS Blueprint for Creating a User

You create the Create a user in a group XaaS blueprint so that you can run the workflow that adds an Active Directory user and assigns the user to an Active Directory group. You can create the blueprint as a standalone XaaS blueprint or as a blueprint component. In this scenario, you are creating a standalone blueprint.

3 Create a Resource Action to Change a User Password

You can create a resource action to allow the consumers of the XaaS create a user blueprint to change the password of the user after they provision the user.

4 Create a Service and Add Creating a Test User Blueprint to the Service

   You can create a service to display the Create a user catalog item in the service catalog.

5 Entitle the Service and the Resource Action to a Consumer

   Business group managers and tenant administrators can entitle the service and the resource action to a user or a group of users. After they are entitled, they can see the service in their catalog and request the Create a test user catalog item that is included in the service. After the consumers provision the item, they can request to change the user password.

Create a Test User as a Custom Resource

You can create a custom resource and map it to the vRealize Orchestrator object type `AD:User`.

Prerequisites

Log in to vRealize Automation as an **XaaS architect**.

Procedure

1 Select **Design > XaaS > Custom Resources**.

2 Click the **New** icon (✚).

3 In the **Orchestrator type** text box, enter **`AD:User`** and press Enter.

4 Select **AD:User** in the list.

5 Type a name for the resource.

   For example, **`Test User`**.

6 Enter a description for the resource.

   For example,
   `This is a test custom resource that I will use for my catalog item to create a user in a group.`

7 Click **Next**.

8 Leave the default values in the form.

9 Click **Finish**.

Results

You created a Test User custom resource and you can see it on the Custom Resources page.

What to do next

Create an XaaS blueprint.

Create an XaaS Blueprint for Creating a User

You create the Create a user in a group XaaS blueprint so that you can run the workflow that adds an Active Directory user and assigns the user to an Active Directory group. You can create the blueprint as a standalone XaaS blueprint or as a blueprint component. In this scenario, you are creating a standalone blueprint.

Prerequisites

- Verify that you create a custom resource action that supports provisioning Active Directory users. See Create a Test User as a Custom Resource.
- Log in to vRealize Automation as an **XaaS architect**.

Procedure

1. Select **Design > XaaS > XaaS Blueprints**.
2. Click the **New** icon (✚).
3. In the Select a workflow pane, navigate to **Orchestrator > Library > Microsoft > Active Directory > User** and select the **Create a user in a group** workflow.
4. Click **Next**.
5. Configure the **General** tab options.

   a. Change the name of the blueprint to `Create a test user`, and leave the description as is.

   b. Deselect the **Make available as a component in the design canvas** check box.

      You are publishing this blueprint directly to the service catalog rather than using it as a blueprint component in the design canvas. You do not need to configure any scale-in or scale-out workflows.

      The **Component Lifecycle** tab is removed from the user interface.
6. Click **Next**.
7. Edit the blueprint form.

   a. Click **The domain name in Win2000 form**.

   b. Click the **Constraints** tab.

   c. Click the **Value** drop-down arrow, select **Constant** in the drop-down menu, and enter `test.domain`.

   d. Click the **Visible** drop-down arrow, select **Constant** in the drop-down menu, and select **No** in the drop-down menu.

      You made the domain name invisible to the consumer of the catalog item.

   e. Click **Apply** to save the changes.
8. Click **Next**.
9. Select **newUser [Test User]** as an output parameter to be provisioned.
10. Click **Next**.
11. Click **Finish**.
12. On the **XaaS Blueprints** page, select the **Create a test user** row and click **Publish**.

Results

You created a blueprint for creating a test user and you made the blueprint available to add to a service.

What to do next

Create an action to run on the provisioned user account. See Create a Resource Action to Change a User Password.

Create a Resource Action to Change a User Password

You can create a resource action to allow the consumers of the XaaS create a user blueprint to change the password of the user after they provision the user.

Prerequisites

- Log in to vRealize Automation as an **XaaS architect**.
- Verify that you create a custom resource action that supports provisioning Active Directory users. See Create a Test User as a Custom Resource.

Procedure

1 Select **Design** > **XaaS** > **Resource Actions**.

2 Click the **New** icon (✚).

3 Navigate to **Orchestrator** > **Library** > **Microsoft** > **Active Directory** > **User** in the vRealize Orchestrator workflow library, and select the **Change a user password** workflow.

4 Click **Next**.

5 Select **Test User** from the **Resource type** drop-down menu.

This selection is the custom resource you created previously.

6 Select **user** from the **Input parameter** drop-down menu.

7 Click **Next**.

8 Change the name of the resource action to `Change the password of the Test User`, and leave the description as it appears on the **Details** tab.

9 Click **Next**.

10 (Optional) Leave the form as is.

11 Click **Finish**.

12 On the Resource Actions page, select the **Change the password of the Test User** row and click **Publish**.

Results

You created a resource action for changing the password of a user, and you made it available to add to an entitlement.

What to do next

Add the Create a test user blueprint to a service. See Create a Service and Add Creating a Test User Blueprint to the Service.

Create a Service and Add Creating a Test User Blueprint to the Service

You can create a service to display the Create a user catalog item in the service catalog.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **catalog administrator**.
- Verify that you created an XaaS blueprint. See Create an XaaS Blueprint for Creating a User.

Log in to vRealize Automation as a **tenant administrator** or **catalog administrator**.

Procedure

1. Select **Administration > Catalog Management > Services**.
2. Click the **New** icon (✚).
3. Enter `Active Directory Test User` as the name of the service.
4. Select **Active** from the **Status** drop-down menu.
5. Leave the other text boxes blank.
6. Click **OK**.
7. In the Services list, select the **Active Directory Test User** row and click **Manage Catalog Items**.
8. Click the **New** icon (✚).
9. Select **Create a test user**, and click **OK**.

   The Create a test user XaaS blueprint is added to the list of catalog items.
10. Click **Close**.

Results

The Active Directory Test User service now includes the Create a test user blueprint. You do not need to add actions to services.

What to do next

You can entitle users to request the blueprint and the run the action. See Entitle the Service and the Resource Action to a Consumer.

Entitle the Service and the Resource Action to a Consumer

Business group managers and tenant administrators can entitle the service and the resource action to a user or a group of users. After they are entitled, they can see the service in their catalog and request the Create a test user catalog item that is included in the service. After the consumers provision the item, they can request to change the user password.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **business group manager**.
- Verify that the Create a user blueprint is added to a service. See Create a Service and Add Creating a Test User Blueprint to the Service.
- Verify that the Change a User Password resource action exists. See Create a Resource Action to Change a User Password.

Procedure

1. Select **Administration > Catalog Management > Entitlements**.
2. Click the **New** icon (✚).
3. Enter `Create an Active Directory user` in the **Name** text box.
4. Leave the **Description** and **Expiration Date** text boxes empty.
5. Select **Active** from the **Status** drop-down menu.
6. Select the target business group from the **Business Group** drop-down menu.

   For example, IT account managers.
7. Select **All Users and Groups** to entitle all the members of the business group, for example, IT account managers, to create a user account.

   The users that you select can see the service and the catalog items included in the service in the catalog. They can run the change password action on the user account after it is created.
8. Click **Next**.
9. In the **Entitled Services** text box, enter `Active Directory Test User` and press Enter.
10. In the **Entitled Actions** text box, enter `Change the password of the Test User` and press Enter.
11. Click **Finish**.

Results

You created an active entitlement so that users who are members of the IT account managers business group can create users. After the user is provisioned, they can run the change password resource action on the provisioned user account.

What to do next

Log in as user who is entitled to create an Active Directory user. On the **Catalog** tab, verify that the XaaS blueprint creates the user as expected. After the user is created, run the change password action from the **Deployments** tab.

Create and Publish an XaaS Action to Migrate a Virtual Machine

You can create and publish an XaaS resource action to extend the operations that consumers can perform on IaaS-provisioned vSphere virtual machines.

In this scenario, you create a resource action for quick migration of a vSphere virtual machine.

Prerequisites

Log in to vRealize Automation as an **XaaS architect**.

Procedure

1. Create a Resource Action to Migrate a vSphere Virtual Machine

   You create a custom resource action to allow the consumers to migrate vSphere virtual machines after they provision the vSphere virtual machines with IaaS.

2. Publish the Action for Migrating a vSphere Virtual Machine

   To use the Quick migration of virtual machine resource action as a post-provisioning operation, you must publish it.

Create a Resource Action to Migrate a vSphere Virtual Machine

You create a custom resource action to allow the consumers to migrate vSphere virtual machines after they provision the vSphere virtual machines with IaaS.

Procedure

1. Select **Design > XaaS > Resource Actions**.
2. Click **Add** (✚).
3. Navigate to **Orchestrator > Library > vCenter > Virtual Machine management > Move and migrate** in the vRealize Orchestrator workflow library and select the **Quick migration of virtual machine** workflow.
4. Click **Next**.
5. Select **IaaS VC VirtualMachine** from the **Resource type** drop-down menu.
6. Select **vm** from the **Input parameter** drop-down menu.
7. Click **Next**.
8. Leave the name of the resource action and the description as they appear on the **Details** tab.
9. Click **Next**.
10. Leave the form as is.
11. Click **Finish**.

Results

You created a resource action for migrating a virtual machine and you can see it listed on the Resource Actions page.

What to do next

Publish the Action for Migrating a vSphere Virtual Machine

Publish the Action for Migrating a vSphere Virtual Machine

To use the Quick migration of virtual machine resource action as a post-provisioning operation, you must publish it.

Procedure

1 Select **Design > XaaS > Resource Actions**.

2 Select the row of the Quick migration of virtual machine resource action, and click the **Publish** button.

Results

You created and published a vRealize Orchestrator workflow as a resource action. You can navigate to **Administration > Catalog Management > Actions** and see the Quick migration of virtual machine resource action in the list of actions. You can assign an icon to the resource action. See Assign an Icon to an XaaS Resource Action.

What to do next

Add the action to the entitlements that contain the IaaS-provisioned vSphere virtual machines. See Entitle Users to Services, Catalog Items, and Actions.

Create an XaaS Action to Migrate a Virtual Machine With vMotion

By using XaaS, you can create and publish a resource action to migrate an IaaS-provisioned virtual machine with vMotion.

In this scenario, you create a resource action to migrate a vSphere virtual machine with vMotion. In addition, you edit the workflow presentation by using the form designer and change the way the consumers see the action when they request it.

Prerequisites

Log in to vRealize Automation as an **XaaS architect**.

Procedure

1 Create an Action to Migrate a vSphere Virtual Machine With vMotion

  You can create a custom resource action to allow the service catalog users to migrate a vSphere virtual machine with vMotion after they provision the machine with IaaS.

2 Edit the Resource Action Form

  The resource action form maps the vRealize Orchestrator workflow presentation. You can edit the form and define what the consumers of the resource action see when they decide to run the post-provisioning operation.

3 Add a Submitted Action Details Form and Save the Action

   You can add a new form to the Migrate a virtual machine with vMotion resource action to define what the consumers see after they request to run the post-provisioning operation.

4 Publish the Action for Migrating a Virtual Machine with vMotion

   To use the Migrate a virtual machine with vMotion resource action as a post-provisioning operation, you must publish it.

Create an Action to Migrate a vSphere Virtual Machine With vMotion

You can create a custom resource action to allow the service catalog users to migrate a vSphere virtual machine with vMotion after they provision the machine with IaaS.

Procedure

1 Select **Design > XaaS > Resource Actions**.

2 Click **Add** (✚).

3 Navigate to **Orchestrator > Library > vCenter > Virtual Machine management > Move and migrate** in the vRealize Orchestrator workflow library and select the **Migrate virtual machine with vMotion** workflow.

4 Click **Next**.

5 Select **IaaS VC VirtualMachine** from the **Resource type** drop-down menu.

6 Select **vm** from the **Input parameter** drop-down menu.

7 Click **Next**.

8 Leave the name of the resource action and the description as they appear on the **Details** tab.

9 Click **Next**.

What to do next

Edit the Resource Action Form.

Edit the Resource Action Form

The resource action form maps the vRealize Orchestrator workflow presentation. You can edit the form and define what the consumers of the resource action see when they decide to run the post-provisioning operation.

Procedure

1 Click the **Delete** icon (✖) to delete the **pool** element.

2 Edit the **host** element.

   a Click the **Edit** icon (✏) next to the **host** field.

   b Type `Target host` in the **Label** text box.

   c Select **Search** from the **Type** drop-down menu.

d Click the **Constraints** tab.

e Select **Constant** from the **Required** drop-down menu and select **Yes**.

You made the host field always required.

f Click **Submit**.

3 Edit the **priority** element.

a Click the **Edit** icon (✏) next to the **priority** field.

b Type `Priority of the task` in the **Label** text box.

c Select **Radio button group** from the **Type** drop-down menu.

d Click the **Values** tab, and deselect the **Not set** check box.

e Enter `lowPriority` in the **Predefined values** search text box, and press Enter.

f Enter `defaultPriority` in the **Predefined values** search text box, and press Enter.

g Enter `highPriority` in the **Predefined values** search text box, and press Enter.

h Click **Submit**.

When the consumers request the resource action, they see a radio button group with three radio buttons: **lowPriority**, **defaultPriority**, and **highPriority**.

4 Edit the **state** element.

a Click the **Edit** icon (✏) next to the **state** field.

b Type `Virtual machine state` in the **Label** text box.

c Select **Drop-down** from the **Type** drop-down menu.

d Click the **Values** tab, and deselect the **Not set** check box.

e Enter `poweredOff` in the **Predefined values** search text box, and press Enter.

f Enter `poweredOn` in the **Predefined values** search text box, and press Enter.

g Enter `suspended` in the **Predefined values** search text box, and press Enter.

h Click **Submit**.

When the consumers request the resource action, they see a drop-down menu with three options: **poweredOff**, **poweredOn**, and **suspended**.

Results

You edited workflow presentation of the Migrate a virtual machine with vMotion workflow.

What to do next

Add a Submitted Action Details Form and Save the Action.

Add a Submitted Action Details Form and Save the Action

You can add a new form to the Migrate a virtual machine with vMotion resource action to define what the consumers see after they request to run the post-provisioning operation.

Procedure

1 Click the **New Form** icon (✚) next to the **Form** drop-down menu.

2 Type `Submitted action` in the **Name** text box.

3 Leave the **Description** field blank.

4 Select **Submitted action details** from the **Screen type** menu.

5 Click **Submit**.

6 Click the **Edit** icon (✏) next to the **Form page** drop-down menu.

7 Type `Details` in the **Heading** text box.

8 Click **Submit**.

9 Drag the **Text** element from the Form pane and drop it to the **Form** page.

10 Type
`You submitted a request to migrate your machine with vMotion. Wait until the process completes successfully.`

11 Click outside of the text box to save the changes.

12 Click **Submit**.

13 Click **Add**.

Results

You created a resource action to migrate a virtual machine with vMotion and you can see it listed on the Resource Actions page.

What to do next

Publish the Action for Migrating a Virtual Machine with vMotion.

Publish the Action for Migrating a Virtual Machine with vMotion

To use the Migrate a virtual machine with vMotion resource action as a post-provisioning operation, you must publish it.

Procedure

1 Select **Design > XaaS > Resource Actions**.

2 Select the row of the Migrate a virtual machine with vMotion action, and lick the **Publish** button.

Results

You created and published a vRealize Orchestrator workflow as a resource action. You can navigate to **Administration > Catalog Management > Actions** and see the Migrate virtual machine with vMotion resource action in the list of actions. You can assign an icon to the resource action. See Assign an Icon to an XaaS Resource Action.

You also edited the presentation of the workflow and defined the look and feel of the action.

What to do next

Business group managers and tenant administrators can include the Migrate a virtual machine with vMotion resource action in an entitlement. For more information about how to create and publish IaaS blueprints for virtual platforms, see Designing Machine Blueprints.

Create and Publish an XaaS Action to Take a Snapshot

By using XaaS, you can create and publish a resource action to take a snapshot of a vSphere virtual machine that was provisioned with IaaS.

In this scenario, you create a resource action to take a snapshot of a vSphere virtual machine provisioned withIaaS. In addition, you edit the workflow presentation by using the form designer and change the way the consumers see the action when they request it.

Prerequisites

Log in to vRealize Automation as an **XaaS architect**.

Procedure

1 Create the Action to Take a Snapshot of a vSphere Virtual Machine

  You can create a custom resource action to allow the consumers to take a snapshot of a vSphere virtual machine after they provision the machine with IaaS.

2 Publish the Action for Taking a Snapshot

  To use the Create a snapshot resource action as a post-provisioning operation, you must publish it.

Create the Action to Take a Snapshot of a vSphere Virtual Machine

You can create a custom resource action to allow the consumers to take a snapshot of a vSphere virtual machine after they provision the machine with IaaS.

Procedure

1 Select **Design > XaaS > Resource Actions**.

2 Click **Add** (✚).

3 Navigate to **Orchestrator > Library > vCenter > Virtual Machine management > Snapshot** in the vRealize Orchestrator workflow library and select the **Create a snapshot** workflow.

4 Click **Next**.

5 Select **IaaS VC VirtualMachine** from the **Resource type** drop-down menu.

6 Select **vm** from the **Input parameter** drop-down menu.

7 Click **Next**.

8 Leave the name of the resource action and the description as they appear on the **Details** tab.

9 Click **Next**.

10 Leave the form as is.

11 Click **Add**.

Results

You created a resource action for taking a snapshot of a virtual machine and you can see it listed on the Resource Actions page.

What to do next

Publish the Action for Taking a Snapshot.

Publish the Action for Taking a Snapshot

To use the Create a snapshot resource action as a post-provisioning operation, you must publish it.

Procedure

1 Select **Design > XaaS > Resource Actions**.

2 Select the row of the Create a snapshot action, and click the **Publish** button.

Results

You created and published a vRealize Orchestrator workflow as a resource action. You can navigate to **Administration > Catalog Management > Actions** and see the Create a snapshot resource action in the list of actions. You can assign an icon to the resource action. See Assign an Icon to an XaaS Resource Action.

What to do next

Business group managers and tenant administrators can include the Create a snapshot resource action in an entitlement. For more information about how to create and publish IaaS blueprints for virtual platforms, see Designing Machine Blueprints.

Create and Publish an XaaS Action to Start an Amazon Virtual Machine

By using XaaS, you can create and publish actions to extend the operations that the consumers can perform on third-party provisioned resources.

In this scenario, you create and publish a resource action for quick starting of Amazon virtual machines.

Prerequisites

- Install the vRealize Orchestrator plug-in for Amazon Web Services on your default vRealize Orchestrator server.
- Create or import a vRealize Orchestrator workflow for resource mapping of Amazon instances.

Procedure

1 Create a Resource Mapping for Amazon Instances

   You can create a resource mapping to associate Amazon instances provisioned by using IaaS with the vRealize Orchestrator type `AWS:EC2Instance` exposed by the Amazon Web Services plug-in.

2 Create a Resource Action to Start an Amazon Virtual Machine

   You can create a resource action so that the consumers can start provisioned Amazon virtual machines.

3 Publish the Action for Starting Amazon Instances

   To use the newly created Start Instances resource action for post-provisioning operations on Amazon virtual machines, you must publish it.

Create a Resource Mapping for Amazon Instances

You can create a resource mapping to associate Amazon instances provisioned by using IaaS with the vRealize Orchestrator type `AWS:EC2Instance` exposed by the Amazon Web Services plug-in.

Prerequisites

- Log in to vRealize Automation as an **XaaS architect**.
- Create or import a vRealize Orchestrator resource mapping workflow or script action.

Procedure

1 Select **Design > XaaS > Resource Mappings**.

2 Click **Add** (➕).

3 Enter `EC2 Instance` in the **Name** text box.

4 Enter `Cloud Machine` in the **Catalog Resource Type** text box.

5 Enter `AWS:EC2Instance` in the **Orchestrator Type** text box.

6 Select **Always available**.

7 Select the type of resource mapping to use.

8 Select your custom resource mapping script action or workflow from the vRealize Orchestrator library.

9 Click **Add**.

Results

You can use your Amazon resource mapping to create resource actions for Amazon machines provisioned by using IaaS.

What to do next

Create a Resource Action to Start an Amazon Virtual Machine.

Create a Resource Action to Start an Amazon Virtual Machine

You can create a resource action so that the consumers can start provisioned Amazon virtual machines.

Prerequisites

Log in to vRealize Automation as an **XaaS architect**.

Procedure

1. Select **Design** > **XaaS** > **Resource Actions**.
2. Click **Add** (✚).
3. Select **Orchestrator** > **Library** > **Amazon Web Services** > **Elastic Cloud** > **Instances** and select the **Start Instances** workflow in the workflows folder.
4. Click **Next**.
5. Select **EC2 Instance** from the **Resource type** drop-down menu.

   This is the name of the resource mapping you previously created.
6. Select **instance** from the **Input parameter** drop-down menu.

   This is the input parameter of the resource action workflow to match the resource mapping.
7. Click **Next**.
8. Leave the name and the description as they are.

   The default name of the resource action is Start Instances.
9. Click **Next**.
10. Leave the fields as they are on the **Form** tab.
11. Click **Add**.

Results

You created a resource action for starting Amazon virtual machines and you can see it on the Resource Actions page.

What to do next

Publish the Action for Starting Amazon Instances.

Publish the Action for Starting Amazon Instances

To use the newly created Start Instances resource action for post-provisioning operations on Amazon virtual machines, you must publish it.

Prerequisites

Log in to vRealize Automation as an **XaaS architect**.

Procedure

1. Select **Design > XaaS > Resource Actions**.
2. Select the row of the Start Instances resource action, and click **Publish**.

Results

The status of the Start Instances resource action changes to Published.

What to do next

Add the start instances action to the entitlement that includes the Amazon catalog item. See Entitle Users to Services, Catalog Items, and Actions.

Troubleshooting Incorrect Accents and Special Characters in XaaS Blueprints

When you create XaaS blueprints for languages that use non-ASCII strings, the accents and special characters are displayed as unusable strings.

Cause

A vRealize Orchestrator configuration property that is not set by default, might be enabled.

Solution

1. On the Orchestrator server system, navigate to `/etc/vco/app-server/`.
2. Open the `vmo.properties` configuration file in a text editor.
3. Verify that the following property is deactivated.

   ```
   com.vmware.o11n.webview.htmlescaping.disabled
   ```

4. Save the `vmo.properties` file.
5. Restart the vRealize Orchestrator server.

Publishing a Blueprint

Blueprints are saved in the draft state and must be manually published before you can configure them as catalog items or use them as blueprint components in the design canvas.

After you publish the blueprint, you can entitle it to make it available for provisioning requests in the service catalog.

You need to publish a blueprint only once. Any changes you make to a published blueprint are automatically reflected in the catalog and in nested blueprint components.

Publish a Blueprint

You can publish a blueprint for use in machine provisioning and optionally for reuse in another blueprint. To use the blueprint for requesting machine provisioning, you must entitle the blueprint after publishing it. Blueprints that are consumed as components in other blueprints do not required entitlement.

Prerequisites

- Log in to vRealize Automation as an **infrastructure architect**.
- Create a blueprint. See *Checklist for Creating vRealize Automation Blueprints*.

Procedure

1. Click the **Design** tab.
2. Click **Blueprints**.
3. Point to the blueprint to publish and click **Publish**.
4. Click **OK**.

Results

The blueprint is published as a catalog item but you must first entitle it to make it available to users in the service catalog.

What to do next

Add the blueprint to the catalog service and entitle users to request the catalog item for machine provisioning as defined in the blueprint.

Working with Developer-driven Blueprints

In addition to the user interface-driven method of creating vRealize Automation blueprints, you can also work with blueprints programmatically using tools such as vRealize CloudClient, with standalone supplied or otherwise sourced blueprints, and in concert with other developers by using vRealize Suite applications and workflows and third-party tools.

For information about these methods, see the following topics:

- Exporting and Importing Blueprints and Content
- Downloading and Configuring the Supplied Standalone Blueprint
- Creating Blueprints and other IaaS Content in a Multi-developer Environment

Exporting and Importing Blueprints and Content

You can programmatically export blueprints and content from one vRealize Automation environment to another by using the vRealize Automation REST API or by using the vRealize CloudClient.

For example, you can create and test your blueprints in a development environment and then import them into your production environment. Or you can import a property definition from a community forum into your active vRealize Automation tenant instance.

You can programmatically import and export any of the following vRealize Automation content items:

- Application blueprints and all their components
- IaaS machine blueprints
- Software components
- XaaS blueprints
- Component profiles
- Property groups

  Property group information is tenant-specific and is only imported with the blueprint if the property group already exists in the target vRealize Automation instance.

When you export a blueprint from one vRealize Automation instance tenant into another, the property group information defined for that blueprint is not recognized for the imported blueprint unless the property group already exists in the target tenant instance. For example, if you import a blueprint that contains a property group named `mica1`, the `mica1` property group is not present in the imported blueprint unless the `mica1` property group already exists in the vRealize Automation instance in which you import the blueprint. To avoid losing property group information when exporting a blueprint from one vRealize Automation instance to another, use vRealize CloudClient to create an export package zip file that contains the property group and import that package zip file into the target tenant before you import the blueprint. For more information about using vRealize CloudClient to list, package, export, and import property groups, as well as other vRealize Automation items, see the VMware Developer Center at https://developercenter.vmware.com/tool/cloudclient.

Table 5-63. Choosing Your Import and Export Tool

Tool	More information
vRealize CloudClient	See the vRealize CloudClient page on the VMware code.vmware.com site at https://developercenter.vmware.com/tool/cloudclient.
vRealize Automation REST API	See API documentation in the VMware API Explorer for vRealize Automation at https://code.vmware.com/apis/vrealize-automation.

Note When exporting and importing blueprints programmatically across vRealize Automation deployments, for example from a test to a production environment or from one organization to another, it is important to recognize that clone template data is included in the package. When you import the blueprint package, default settings are populated based on information in the package. For example, if you export and then import a blueprint that was created using a clone-style workflow, and the template from which that clone data was derived does not exist in an endpoint in the vRealize Automation deployment in which you import the blueprint, some imported blueprint settings are not applicable for that deployment.

Scenario: Importing the Dukes Bank for vSphere Sample Application and Configuring for Your Environment

As an IT professional evaluating or learning vRealize Automation, you want to import a robust sample application into your vRealize Automation instance so you can quickly explore the available functionality and determine how you might build vRealize Automation blueprints that suit the needs of your organization.

Prerequisites

- Prepare a CentOS 6.x Linux reference machine, convert it to a template, and create a customization specification. See Scenario: Prepare for Importing the Dukes Bank for vSphere Sample Application Blueprint.
- Create an external network profile to provide a gateway and a range of IP addresses. See Create an External Network Profile by Using A Third-Party IPAM Provider.
- Map your external network profile to your vSphere reservation. See Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer. The sample application cannot provision successfully without an external network profile.

- Verify that you have both the **infrastructure architect** and **software architect** privileges. Both roles are required to import the Dukes Bank sample application and to interact with the Dukes Bank blueprints and software components.

Procedure

1 Scenario: Import the Dukes Bank for vSphere Sample Application

  You download the Dukes Bank for vSphere application from your vRealize Automation appliance. You import the sample application into your vRealize Automation tenant to view a working sample of a multi-tiered vRealize Automation blueprint that includes multiple machine components with networking and software components.

2 Scenario: Configure Dukes Bank vSphere Sample Components for Your Environment

  Using your infrastructure architect privileges, you configure each of the Dukes Bank machine components to use the customization specification, template, and machine prefixes that you created for your environment.

Results

You have configured the Dukes Bank for vSphere sample application for your environment to use as a starting point for developing your own blueprints, as a tool to evaluate vRealize Automation, or as a learning resource to assist you in understanding vRealize Automation functionality and components.

Scenario: Import the Dukes Bank for vSphere Sample Application

You download the Dukes Bank for vSphere application from your vRealize Automation appliance. You import the sample application into your vRealize Automation tenant to view a working sample of a multi-tiered vRealize Automation blueprint that includes multiple machine components with networking and software components.

Procedure

1 Log in to your vRealize Automation appliance as root by using SSH.

2 Download the Dukes Bank for vSphere sample application from your vRealize Automation appliance to `/tmp`.

```
wget --no-check-certificate  https://vRealize_VA_Hostname_fqdn:5480/blueprints/
DukesBankAppForvSphere.zip
```

  Do not unzip the package.

3 Download vRealize CloudClient from http://developercenter.vmware.com/tool/cloudclient to `/tmp`.

4 Unzip the `cloudclient-4x-dist.zip` package.

5 Run vRealize CloudClient under the `/bin` directory.

```
$>./bin/cloudclient.sh
```

6 If prompted, accept the license agreement.

7 Using vRealize CloudClient, log in to the vRealize Automation appliance as a user with **software architect** and **infrastructure architect** privileges.

```
CloudClient>vra login userpass --server https://vRealize_VA_Hostname_fqdn --user
<user@domain.com> --tenant <TenantName>
```

8 When prompted, enter your login password.

9 Validate that the `DukesBankAppForvSphere.zip` content is available.

```
vra content import --path /<Path>/DukesBankAppForvSphere.zip --dry-run true --resolution OVERWRITE
```

Note that the OVERWRITE entry is case-specific and requires uppercase.

By configuring the resolution to overwrite instead of *skip*, you allow vRealize Automation to correct conflicts when possible.

10 Import the Dukes Bank sample application.

```
vra content import --path /<Path>/DukesBankAppForvSphere.zip --dry-run false --resolution
OVERWRITE
```

Note that the OVERWRITE entry is case-specific and requires uppercase.

Results

When you log in to the vRealize Automation console as a user with software architect and infrastructure architect privileges, you see Dukes Bank blueprints and software components on the **Design > Blueprints** tab and the **Design > Software Components** tab.

Scenario: Configure Dukes Bank vSphere Sample Components for Your Environment

Using your infrastructure architect privileges, you configure each of the Dukes Bank machine components to use the customization specification, template, and machine prefixes that you created for your environment.

This scenario configures the machine components to clone machines from the template you created in the vSphere Web Client. If you want to create space-efficient copies of a virtual machine based on a snapshot, the sample application also supports linked clones. Linked clones use a chain of delta disks to track differences from a parent machine, are provisioned quickly, reduce storage cost, and are ideal to use when performance is not a high priority.

Procedure

1 Log in to the vRealize Automation console as an **infrastructure architect**.

You can configure the Dukes Bank sample application to work in your environment with only the **infrastructure architect** role, but if you want to view or edit the sample software components you also need the **software architect** role.

2 Select **Design > Blueprints**.

3 Select the **DukesBankApplication** blueprint and click the **Edit** icon.

4 Edit the appserver-node so vRealize Automation can provision this machine component in your environment.

   You configure the blueprint to provision multiple instances of this machine component so you can verify the load balancer node functionality.

   a Click the **appserver-node** component on the design canvas.

      Configuration details appear in the bottom panel.

   b Select your machine prefix from the **Machine prefix** drop-down menu.

   c Configure your blueprint to provision at least two and up to ten instances of this node by selecting a minimum of 2 instances and a maximum of 10.

      On the request form, users are able to provision at least two and up to ten appserver nodes. If users are entitled to the scale in and scale out actions, they can scale their deployment to meet changing needs.

   d Click the **Build Information** tab.

   e Select **Cloneworkflow** from the **Provisioning workflow** drop-down menu.

   f Select your **dukes_bank_template** from the **Clone from** dialog.

   g Enter your `Customspecs_sample` in the **Customization spec** text box.

      This field is case sensitive.

   h Click the **Machine Resources** tab.

   i Verify that memory settings are at least 2048 MB.

5 Edit the loadbalancer-node so vRealize Automation can provision this machine component in your environment.

   a Click the **loadbalancer-node** component on the design canvas.

   b Select your machine prefix from the **Machine prefix** drop-down menu.

   c Click the **Build Information** tab.

   d Select **Cloneworkflow** from the **Provisioning workflow** drop-down menu.

   e Select your **dukes_bank_template** from the **Clone from** dialog.

   f Enter your `Customspecs_sample` in the **Customization spec** text box.

      This field is case sensitive.

   g Click the **Machine Resources** tab.

   h Verify that memory settings are at least 2048 MB.

6 Repeat for the **database-node** machine component.

7 Click **Save and Finish**.

   Your changes are saved and you return to the **Blueprints** tab.

8 Select the **DukesBankApplication** blueprint and click **Publish**.

Results

You configured the Dukes Bank sample application blueprint for your environment and published the finished blueprint.

What to do next

Published blueprints do not appear to users in the catalog until you configure a catalog service, add the blueprint to a service, and entitle users to request your blueprint. See Checklist for Configuring the Service Catalog.

After you configure your Dukes Bank blueprint to display in the catalog, you can request to provision the sample application. See Scenario: Test the Dukes Bank Sample Application.

Scenario: Test the Dukes Bank Sample Application

You request the Dukes Bank catalog item, and log in to the sample application to verify your work and view vRealize Automation blueprint functionality.

Prerequisites

- Import the Dukes Bank sample application and configure the blueprint components to work in your environment. See Scenario: Importing the Dukes Bank for vSphere Sample Application and Configuring for Your Environment.
- Configure the service catalog and make your published Dukes Bank blueprint available for users to request. See Checklist for Configuring the Service Catalog.
- Verify that virtual machines you provision can reach the yum repository.

Procedure

1 Log in to the vRealize Automation console as a user who is entitled to the Dukes Bank catalog item.

2 Click the **Catalog** tab.

3 Locate the Dukes Bank sample application catalog item and click **Request**.

4 Fill in the required request information for each component that has a red asterisk.

   a Navigate to the JBossAppServer component to fill in the required request information.

   b Enter the fully qualified domain name of your vRealize Automation appliance in the **app_content_server_ip** text box.

   c Navigate to the Dukes_Bank_App software components to fill in the required request information.

   d Enter the fully qualified domain name of your vRealize Automation appliance in the **app_content_server_ip** text boxes.

5 Click **Submit**.

   Depending on your network and your vCenter Server instance, it can take approximately 15-20 minutes for the Dukes Bank sample application to fully provision. You can monitor the status under the **Deployments** tab. After the application provisions you can view the catalog item details on the **Deployments** tab.

6 After the application provisions, locate the IP address of the load balancer server so you can access the Dukes Bank sample application.

   a Click **Deployments**.

   b Locate your Dukes Bank sample application deployment and click the deployment name.

   c On the **Components** tab, select the Apache load balancer server.

   d Select the **Network** tab.

   e Make a note of the IP address.

7 Log in to the Dukes Bank sample application.

   a Navigate to your load balancer server at http://*IP_Apache_Load_Balancer:8081*/bank/main.faces.

      If you want to access the application servers directly, you can navigate to http://*IP_AppServer:8080*/bank/main.faces.

   b Enter `200` in the **Username** text box.

   c Enter `foobar` in the **Password** text box.

Results

You have a working Dukes Bank sample application to use as a starting point for developing your own blueprints, as a tool to evaluate vRealize Automation, or as a learning resource to assist you in understanding vRealize Automation functionality and components.

Downloading and Configuring the Supplied Standalone Blueprint

You can download a supplied standalone blueprint, and its associated software components, from the vRealize Automation appliance.

The Download and Configure vRealize Automation Standalone Blueprint document guides you through the process of downloading a standalone vRealize Automation blueprint from the vRealize Automation appliance and then importing, configuring, and using that blueprint in vRealize Automation in conjunction with several vRealize Orchestrator workflows.

Creating Blueprints and other IaaS Content in a Multi-developer Environment

Multiple developers can use vRealize Orchestrator workflows in conjunction with vRealize Suite and third party developer tools to work simultaneously on different vRealize Automation blueprint artifacts for the same or different vRealize Automation blueprints.

You can use tools such as vRealize Suite Lifecycle Manager to facilitate a multi-developer environment for vRealize Automation and other vRealize Suite tools and OVAs as well as third-party tools such as GitLab/GitHub, Houdini, and other application artifacts from the VMware Solutions Exchange.

To learn more about creating vRealize Automation blueprints and other IaaS content such as properties, event broker subscriptions, software components, and vRealize Orchestrator workflows in a multi-developer environment, see the following resources:

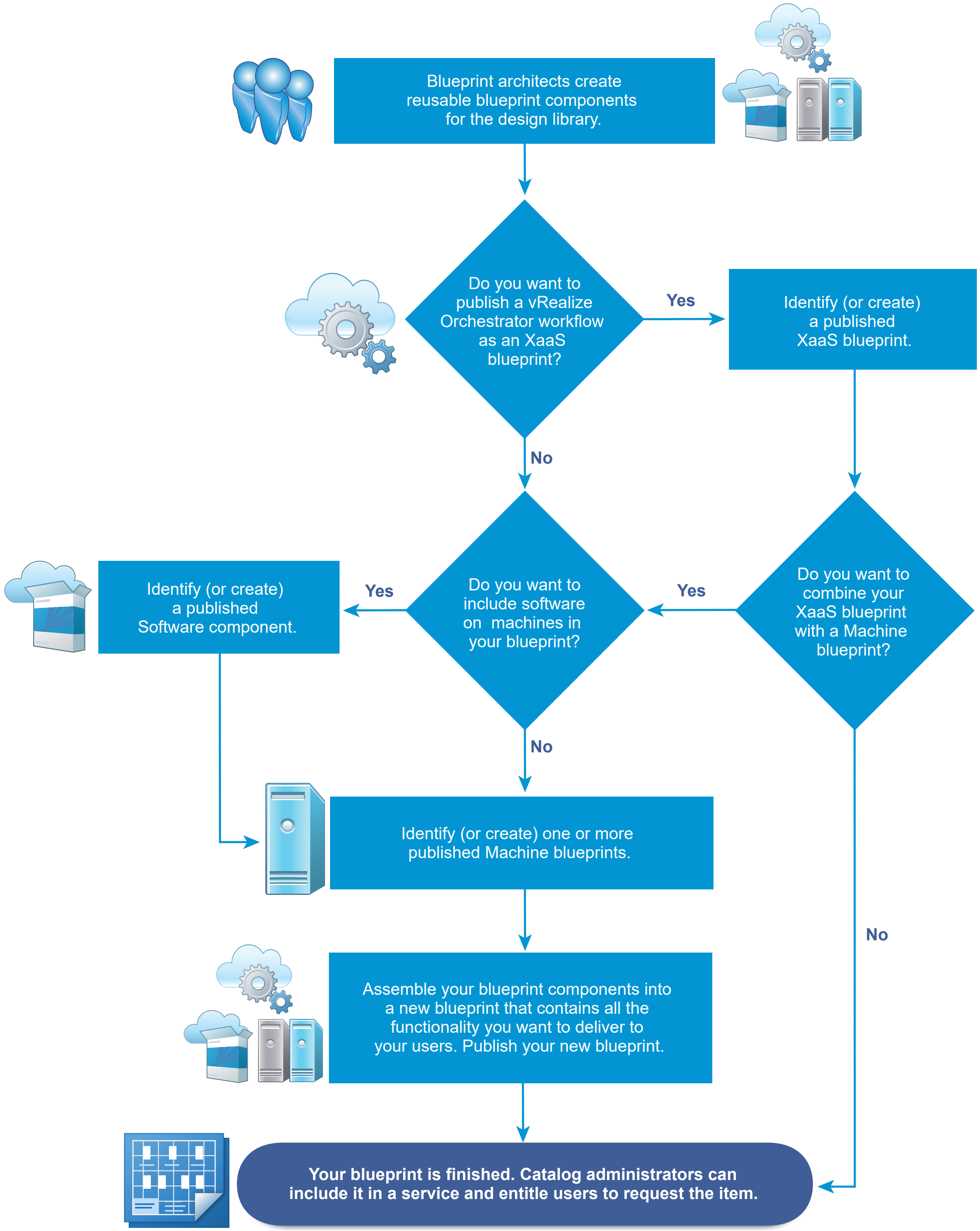
- Video - What's New in Lifecycle Manager
- Blog post - vRealize Automation with Infrastructure Blueprint - Configuring Multi-developer Environment
- Doc - Downloading and Configuring the Supplied Standalone Blueprint
- Blog post - Lifecycle Manager with GitLab Integration
- Blog post - LifeCycle Manager overview

Assembling Composite Blueprints

You can reuse published blueprints and blueprint components, combining them in new ways to create IT service packages that deliver elaborate functionality to your users.

If the component blueprints have custom forms, the custom request forms are not applied to the new blueprint. You must create new forms for the new blueprint. For more about custom request forms, see Customizing Blueprint Request Forms.

Figure 5-5. Workflow for Assembling Composite Blueprints


Blueprint architects create reusable blueprint components for the design library.
Do you want to publish a vRealize Orchestrator workflow as an XaaS blueprint?
Yes
Identify (or create) a published XaaS blueprint.
No
Do you want to combine your XaaS blueprint with a Machine blueprint?
Yes
Do you want to include software on machines in your blueprint?
Yes
Identify (or create) a published Software component.
No
Identify (or create) one or more published Machine blueprints.
No
Assemble your blueprint components into a new blueprint that contains all the functionality you want to deliver to your users. Publish your new blueprint.
Your blueprint is finished. Catalog administrators can include it in a service and entitle users to request the item.

- Understanding Nested Blueprint Behavior

You can reuse blueprints by nesting them in another blueprint as a component. You nest blueprints for reuse and modularity control in machine provisioning, but there are specific rules and considerations when you work with nested blueprints.

- Using Machine Components and Software Components When Assembling a Blueprint

  You deliver Software components by placing them on top of supported machine components when you assemble blueprints.

- Creating Property Bindings Between Blueprint Components

  In several deployment scenarios, a component needs the property value of another component to customize itself. You can bind properties of XaaS, machines, Software, and custom properties to other properties in a blueprint.

- Creating Dependencies and Controlling the Order of Provisioning

  If you need information from one of your blueprint components to complete the provisioning of another component, you can draw an explicit dependency on the design canvas to stagger provisioning so the dependent component is not provisioned prematurely. Explicit dependencies control the build order of a deployment and trigger dependent updates during a scale in or scale out operation. Software components are required to be ordered in a blueprint.

Understanding Nested Blueprint Behavior

You can reuse blueprints by nesting them in another blueprint as a component. You nest blueprints for reuse and modularity control in machine provisioning, but there are specific rules and considerations when you work with nested blueprints.

A blueprint that contains one or more nested blueprints is called an outer blueprint. When you add a blueprint component to the design canvas while creating or editing another blueprint, the blueprint component is called a nested blueprint and the container blueprint to which it is added is called the outer blueprint.

Using nested blueprints presents considerations that are not always obvious. It is important to understand the rules and considerations to make the best use of your machine provisioning capabilities.

General Rules and Considerations for Nesting Blueprints

- As a best practice to minimize blueprint complexity, limit blueprints to three levels deep, with the top-level blueprint serving as one of the three levels.
- If a user is entitled to the outer blueprint, that user is entitled to its nested blueprints.
- You can apply an approval policy to a blueprint. When approved, the blueprint catalog item and all its components, including nested blueprints, are provisioned. You can also apply different approval policies to different components. All the approval policies must be approved before the requested blueprint is provisioned.

- When you edit a published blueprint, you are not changing deployments that are already provisioned by using that blueprint. At the time of provisioning, the resulting deployment reads current values from the blueprint, including from its nested blueprints. The only changes you can pass on to provisioned deployments are edits to software components, for example edits to update or uninstall scripts.
- Settings you define in the outer blueprint override settings configured in nested blueprints with the following exceptions:
  - You can change the name of a nested blueprint, but you cannot change the name of a machine component, or any other component, inside a nested blueprint.
  - You cannot add or delete custom properties for a machine component in a nested blueprint. However, you can edit those custom properties. You cannot add, edit, or delete property groups for a machine component in a nested blueprint.
- Changes you or another architect make to nested blueprint settings appear in the outer blueprints, unless you have overridden those settings in the outer blueprint.
- Limit the maximum lease time on the outer blueprint to the lowest maximum lease value of a component blueprint.

  While the lease time specified on a nested blueprint and on the outer blueprint can be set to any value, the maximum lease time on the outer blueprint should be limited to the lowest maximum lease value of a nested blueprint. This allows the application architect to design a composite blueprint that has uniform and variable lease values, but is within the constraints identified by the infrastructure architect. If the maximum lease value defined on a nested blueprint is less than that defined on the outer blueprint, the provisioning request fails.
- When working in an outer blueprint, you can override the Machine Resources settings that are configured for a machine component in a nested blueprint.
- When working in an outer blueprint, you can drag a software component onto a machine component within a nested blueprint.
- If you open a blueprint in which a machine component in a nested blueprint was removed or its ID was changed, and the machine component was associated to components in the current blueprint, the associated components are removed and the following or similar message appears:

  A machine component in a nested blueprint that is referenced by components in the current blueprint was removed or its machine component ID was changed. All components in the current blueprint that were associated to the missing or changed machine component ID have been removed. Click Cancel to keep the association history between the missing or changed machine component ID in the nested blueprint and components in the current blueprint and correct the problem in the nested blueprint. Open the nested blueprint and re-add the missing machine component with the original ID or change the machine component ID back to its original ID. Click Save to remove all association history between the missing or changed machine component ID in the nested blueprint and components in the current blueprint.

- When you publish a blueprint, software component data is treated like a snapshot. If you later make changes to the software component's properties, only new properties are recognized by the blueprint in which the software component exists. Updates to properties that existed in the software component at the time you published the blueprint are not updated in the blueprint. Only properties that are added after you have published the blueprint are inherited by the blueprint. However, you can make changes to instances of the software component in blueprints in which the software component resides to change that particular blueprint.

Networking and Security Rules and Considerations for Nesting Blueprints

- Networking and security components in outer blueprints can be associated with machines that are defined in nested blueprints.
- NSX network, security, and load balancer components and their settings are not supported in nested blueprints.
- When app isolation is applied in the outer blueprint, it overrides app isolation settings specified in nested blueprints.
- Transport zone settings that are defined in the outer blueprint override transport zone settings that are specified in nested blueprints.
- When working in an outer blueprint, you can configure load balancer settings relative to network component settings and machine component settings that are configured in an inner or nested blueprint.
- For a nested blueprint that contains an on-demand NAT network component, the IP ranges specified in that on-demand NAT network component are not editable in the outer blueprint.
- The outer blueprint cannot contain an inner blueprint that contains on-demand network settings or on-demand load balancer settings. Using an inner blueprint that contains an NSX on-demand network component or NSX load balancer component is not supported.
- For a nested blueprint that contains NSX network or security components, you cannot change the network profile or security policy information specified in the nested blueprint. You can, however, reuse those settings for other vSphere machine components that you add to the outer blueprint.
- To ensure that NSX network and security components in nested blueprints are uniquely named in a composite blueprint, vRealize Automation prefixes the nested blueprint ID to network and security component names that are not already unique. For example, if you add a blueprint with the ID name `xbp_1` to an outer blueprint and both blueprints contain an on-demand security group component named `OD_Security_Group_1`, the component in the nested blueprint is renamed `xbp_1_OD_Security_Group_1` in the blueprint design canvas. Network and security component names in the outer blueprint are not prefixed.

- Component settings can change depending on which blueprint the component resides on. For example, if you include security groups, security tags, or on-demand networks at both the inner and outer blueprint levels, the settings in the outer blueprint override those in the inner blueprint. Network and security components are supported only at the outer blueprint level except for existing networks that work at the inner blueprint level. To avoid issues, add all your security groups, security tags, and on-demand networks only to the outer blueprint.

Software Component Considerations for Nesting Blueprints

For scalable blueprints, it is a best practice to create single layer blueprints that do not reuse other blueprints. Normally, update processes during scale operations are triggered by implicit dependencies such as dependencies you create when you bind a software property to a machine property. However, implicit dependencies in a nested blueprint do not always trigger update processes. If you need to use nested blueprints in a scalable blueprint, you can manually draw dependencies between components in your nested blueprint to create explicit dependencies that always trigger an update.

Using Machine Components and Software Components When Assembling a Blueprint

You deliver Software components by placing them on top of supported machine components when you assemble blueprints.

To support Software components, the machine blueprint you select must contain a machine component based on a template, snapshot, or Amazon machine image that contains the guest agent and the Software bootstrap agent, and it must use a supported provisioning method.

Because the Software agents do not support Internet Protocol version 6 (IPv6), use IPv4 settings.

Note Software components must have an ordered dependency in the blueprint. Unordered software components can cause blueprint provisioning to fail. If there is no actual order dependency for the software components, you can satisfy the blueprint ordering requirement by adding a faux dependency between the software components.

If you are designing blueprints to be scalable, it is a best practice to create single layer blueprints that do not reuse other blueprints. Normally, the update processes that are used during scale operations are triggered by implicit dependencies such as property bindings. However, implicit dependencies in a nested blueprint do not always trigger update processes.

While IaaS architects, application architects, and software architects can all assemble blueprints, only IaaS architects can configure machine components. If you are not an IaaS architect, you cannot configure your own machine components, but you can reuse machine blueprints that your IaaS architect created and published.

To successfully add software components to the design canvas, you must also have business group member, business group administrator, or tenant administrator role access to the target catalog.

If you need to use nested blueprints in a scalable blueprint, you can manually draw dependencies between components in your nested blueprint to create explicit dependencies that always trigger an update.

Note When you publish a blueprint, software component data is treated like a snapshot. If you later make changes to the software component's properties, only new properties are recognized by the blueprint in which the software component exists. Updates to properties that existed in the software component at the time you published the blueprint are not updated in the blueprint. Only properties that are added after you have published the blueprint are inherited by the blueprint. However, you can make changes to instances of the software component in blueprints in which the software component resides to change that particular blueprint.

Table 5-64. Provisioning Methods that Support Software

Machine Type	Provisioning Method
vSphere	Clone
vSphere	Linked Clone
vCloud Director	Clone
vCloud Air	Clone
Amazon Web Services	Amazon Machine Image

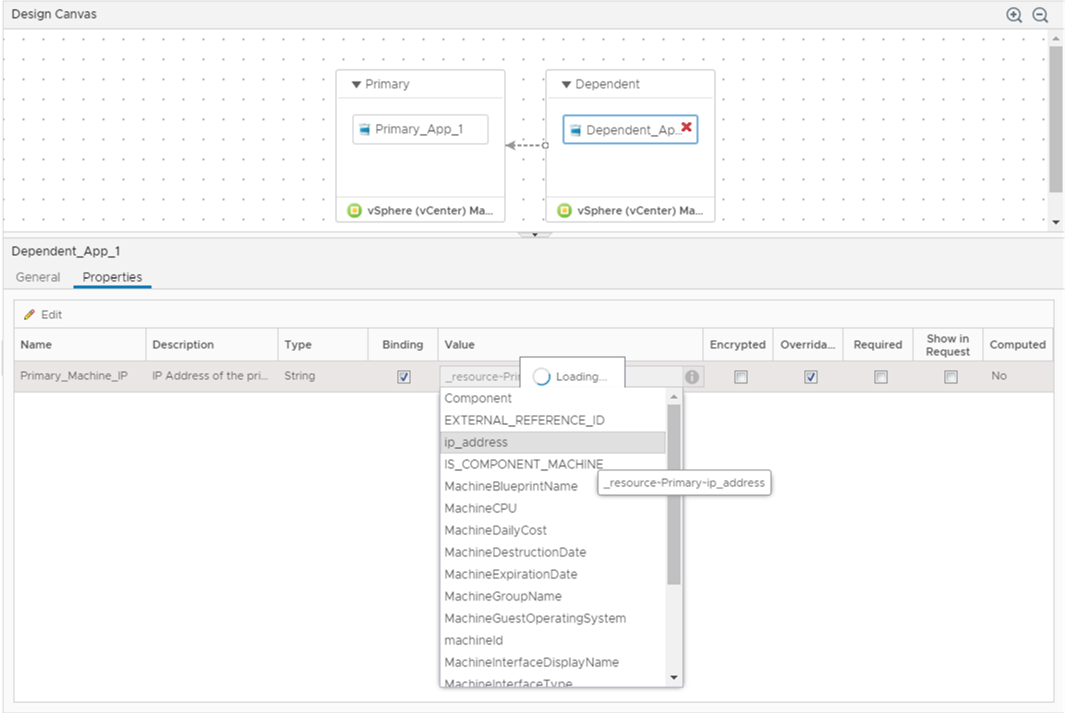
Creating Property Bindings Between Blueprint Components

In several deployment scenarios, a component needs the property value of another component to customize itself. You can bind properties of XaaS, machines, Software, and custom properties to other properties in a blueprint.

For example, your software architect might modify property definitions in the life cycle scripts of a WAR component. A WAR component might need the installation location of the Apache Tomcat server component, so your software architect configures the WAR component to set the server_home property value to the Apache Tomcat server install_path property value. As the architect assembling the blueprint, you have to bind the server_home property to the Apache Tomcat server install_path property for the Software component to provision successfully.

You set property bindings when you configure components in a blueprint. On the Blueprint page, you drag your component onto the canvas and click the **Properties** tab. To bind a property to another property in a blueprint, select the **Bind** checkbox. You can enter *ComponentName~PropertyName* in the value text box, or you can use the down arrow to generate a list of available binding options. You use a tilde character ~ as a delimiter between components and properties. For example, to bind to the property dp_port, on your MySQL software component, you could type mysql~db_port. To bind to properties that are configured during provisioning, such as the IP address of a machine or the host name of a Software component, you enter _resource~*ComponentName*~*PropertyName*. For example, to bind to the reservation name of a machine, you might enter _resource~vSphere_Machine_1~MachineReservationName.

Figure 5-6. Bind a Software Property to the IP address of a machine



Name	Description	Type	Binding	Value	Encrypted	Overrida...	Required	Show in Request	Computed
Primary_Machine_IP	IP Address of the pri...	String	☑	_resource~Pri Loading...	□	☑	□	□	No

Component
EXTERNAL_REFERENCE_ID
ip_address
IS_COMPONENT_MACHINE
MachineBlueprintName
MachineCPU
MachineDailyCost
MachineDestructionDate
MachineExpirationDate
MachineGroupName
MachineGuestOperatingSystem
machineId
MachineInterfaceDisplayName
MachineInterfaceType

_resource~Primary~ip_address

Creating Dependencies and Controlling the Order of Provisioning

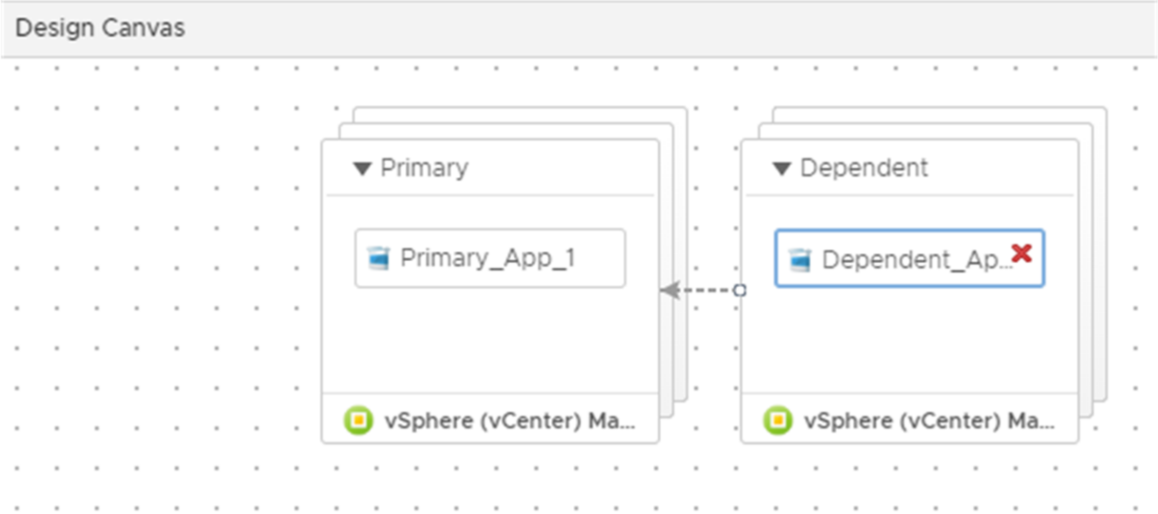
If you need information from one of your blueprint components to complete the provisioning of another component, you can draw an explicit dependency on the design canvas to stagger provisioning so the dependent component is not provisioned prematurely. Explicit dependencies control the build order of a deployment and trigger dependent updates during a scale in or scale out operation. Software components are required to be ordered in a blueprint.

When you design blueprints with multiple machines and applications, you might have properties you need from one machine to finish an application installation on another. For example, if you are building a Web server you might need the host name of the database server before you can install the application and instantiate database tables. If you map an explicit dependency, your database server starts provisioning when your Web server finishes provisioning.

Note Software components must have an ordered dependency in the blueprint. Unordered software components can cause blueprint provisioning to fail. If there is no actual order dependency for the software components, you can satisfy the blueprint ordering requirement by adding a faux dependency between the software components.

To map a dependency on your design canvas, you draw a line from the dependent component to the component you are depending on. When you are finished, the component you want to build second has an arrow pointing to the component you want to build first. For example, in the Controlling the Build Order by Mapping Dependencies figure, the dependent machine is not provisioned until the primary machine is built. Alternatively, you can configure both machines to provision simultaneously but draw a dependency between the software components.

Figure 5-7. Controlling the Build Order by Mapping Dependencies



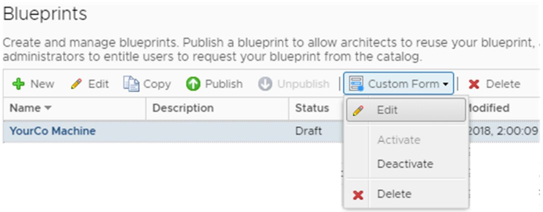
If you are designing blueprints to be scalable, it is a best practice to create single layer blueprints that do not reuse other blueprints. Normally, update processes during scale operations are triggered by implicit dependencies such as dependencies you create when you bind a software property to a machine property. However, implicit dependencies in a nested blueprint do not always trigger update processes. If you need to use nested blueprints in a scalable blueprint, you can manually draw dependencies between components in your nested blueprint to create explicit dependencies that always trigger an update.

Customizing Blueprint Request Forms

Every blueprint that you create and publish displays a form when your users request the blueprint in the catalog. You can use the default form or you can customize blueprint request forms when you create or edit a blueprint. You customize a form when the information provided or required on the default form is not what you want to present to your users.

Customizing Request Forms

You access the custom request form designer from the blueprint data grid or from the blueprint canvas.



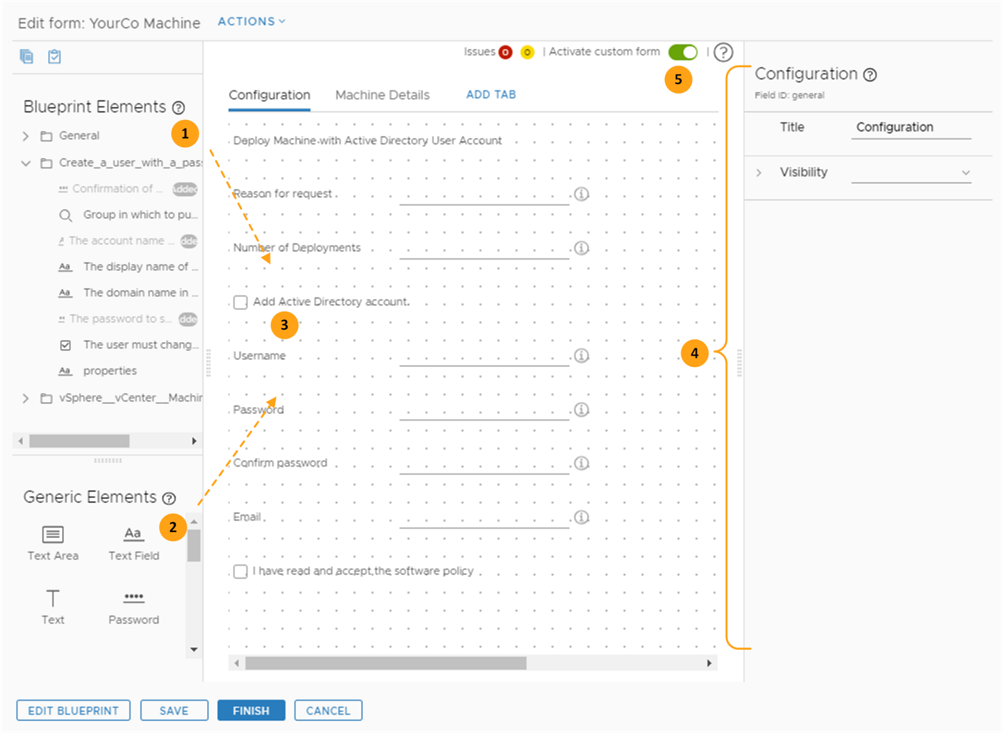
Blueprints

Create and manage blueprints. Publish a blueprint to allow architects to reuse your blueprint, administrators to entitle users to request your blueprint from the catalog.

Name	Description	Status	Modified
YourCo Machine		Draft	2018, 2:00:09

Custom Request Form Designer

You use the form designer to create your custom form.



Edit form: YourCo Machine

To create a custom form:

1 Drag elements (1 and 2) onto the design canvas (3).

2 Configure each element using the properties pane (4).

3 Activate the form (5).

Unless a property is configured prohibit overwriting, the blueprint elements list includes custom properties. If the overridable option on the property is set to no, the field is not eligible for customization.

Validation and Constraints

The custom form designer supports data validation by adding constraints to a field or by using an external validation source. For constraints options that are applied as you create a form, see Custom Form Designer Field Properties.

- For a constraint example, see Create a Custom Request Form with Active Directory Options.

- For external validation, see Using External Validation in the Custom Forms Designer.

When you add validation and dependencies in forms, the requesting user must supply or the system must validate the fields or the dependent fields might not appear on the form.

For example, if you have fields on the first tab that subsequent fields are dependent on, the depend fields might not appear on the succeeding tabs until the dependent value is provided on the preceding tabs.

Custom Request Form Actions

You use the action menu items to populate forms and share forms with other systems.

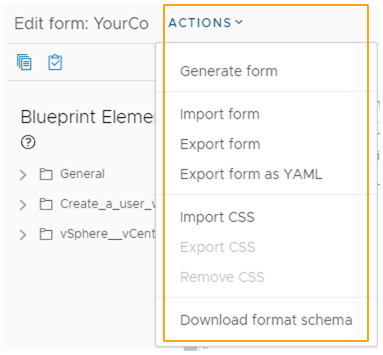


Table 5-65. Custom Request Form Action Menu Items

Action Menu Item	Description
Generate Form	Adds all the fields associated with each blueprint component to the form designer. Each component is added to a tab. If you use this menu item after you created or modified a form, the generated form overwrites your current form. If you use this menu item, you can hide or remove fields that you do not want to present to your users in the catalog. If you do not generate the form, you can still add and configure the text boxes that you want your users to see.
Import Form	Imports a custom form JSON or YAML file.
Export Form	Exports your current custom form as a JSON file. Export the file when you want to use part of it that matches a component that you use in another blueprint.
Export form as YAML	Exports your current custom form as YAML. Export the file as YAML when you want to move a custom from one vRealize Automation instance to another. For example, from your test environment to your production environment. If you prefer to edit the form as YAML, you can export the form, edit it, and then import it back into the blueprint.

Table 5-65. Custom Request Form Action Menu Items (continued)

Action Menu Item	Description

Import CSS	Imports a CSS file that enhances the catalog request form.
	The file might be similar to the following example. This example changes the font size and makes the text bold. The field referenced is the Deploy Machine with Active Directory User Account text field that appears in the image located in the Custom Request Form Designer section above.
	`#<field-ID> .grid-item {`
	`     font-size: 16px;`
	`     font-weight: bold;`
	`     width: 600px;`
	`}`
	In this example, `<field-ID>` is the ID for the field in the canvas. To locate the value, select the field in the canvas. The value is located in the right pane, below the name. In the image above, the value is **`text_d947bc97`**.
	To import the file. Save it as <filename>.css.

Export CSS	Exports your imported CSS.
Remove CSS	Discards your custom CSS. The discarded CSS is not recoverable.
Download format schema	Downloads a JSON file that contains the structure and description of the controls and states used in a custom form. You can use this schema to create a form or to modify an existing form. You can import the modified JSON file as the custom form.

Create a Custom Request Form with Active Directory Options

You create a custom form when the default form provides too much or too little information to the requesting user. You can add more fields to the form, you can hide fields on a form, or you can pre-populate fields and either show or hide them.

This use case is based on a blueprint that contains a vSphere virtual machine type and an XaaS blueprint that configures an Active Directory administrator account on the virtual machine. The XaaS blueprint is based on the Create a user with a password in a group workflow.

Your goal in this use case is to:

- Give the user the option to configure the administrator password.
- Preconfigure the machine details so that CPU and memory values are both based on GB.

How do you benefit from this use case? The use case includes examples of the following form customizations:

- Add specific fields to a blank form.
- Configure a show/hide check box.
- Hide fields until the requesting user selects a check box.
- Add validation to fields.
- Display a memory field in GB even though the blueprint field is calculated in MB.
- Use regular expressions.

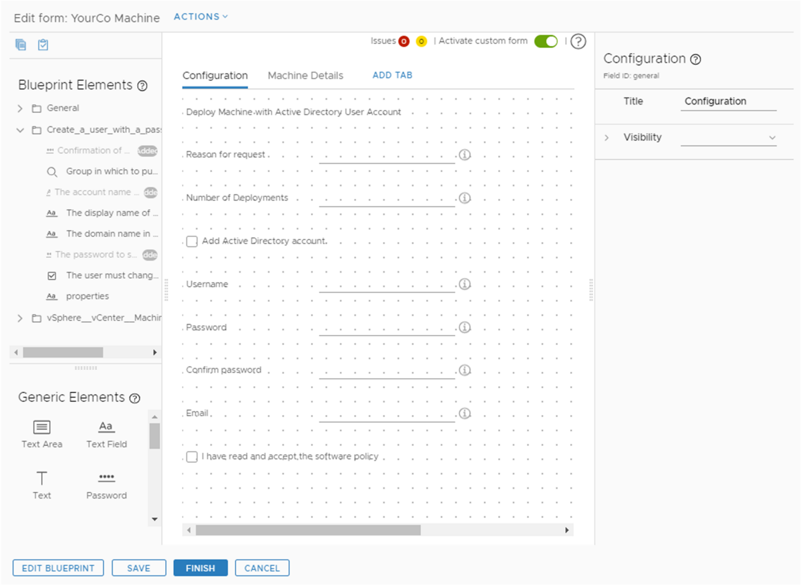
Prerequisites

- Log in to vRealize Automation as an **application architect**, **software architect**, or **infrastructure architect**.
- Create a YourCo Machine and User blueprint that includes a vSphere blueprint and an XaaS blueprint to create an Active Directory user account with a password in a group. For an example, see Create an XaaS Blueprint for Creating a User.

Procedure

1. Select **Design > Blueprints**.
2. Highlight the row containing YourCo Machine and User blueprint and click **Custom Form > Edit**.
3. Rename the General tab.
   a. Click the tab.
   b. In the **Title** property in the right property pane, enter `Configuration`.

4 On your new Configuration tab, add and configure the following fields with the provided values.



Edit form: YourCo Machine

Deploy Machine with Active Directory User Account

Reason for request

Number of Deployments

□ Add Active Directory account.

Username

Password

Confirm password

Email

□ I have read and accept the software policy

Configuration

Field ID: general

Title: Configuration

Visibility

Use the provided Appearance, Values, and Constraints values.

Resolves any errors as you build the form.

Field in Screenshot	Blueprint Element Source	Appearance	Values	Constraints
Deploy Machine with Active Directory User Account	Generic Elements > Text	Label and type ■ Display type = Text Visibility ■ Value source = Constant ■ Visible = Yes	Default value ■ Default value = Deploy Machine with Active Directory User Account ■ Value source = Constant	
Reason for Request	Blueprint Elements > vSphere_vCenter_Machine > Description	Label and type ■ Label = Reason for request ■ Display type = Text Field Visibility ■ Value source = Constant ■ Visible = Yes Read-only ■ Value source = Constant ■ Read-only = No Custom help ■ Signpost help = Provide the reason for your request.		Required ■ Value source = Constant ■ Required = Yes
Number of Deployments	Blueprint Elements > General > Number of Deployments	Label and type ■ Label = Number of Deployments ■ Display type = Integer Visibility ■ Value source = Constant ■ Visible = Yes Read-only ■ Value source = Constant ■ Read-only = No Custom help ■ Signpost help = Select the number of instances of the blueprint to deploy.	Default value ■ Value source = Constant ■ Default value = 1	Required ■ Value source = Constant ■ Required = Yes Minimum value ■ Value source = Constant ■ Min value = 1

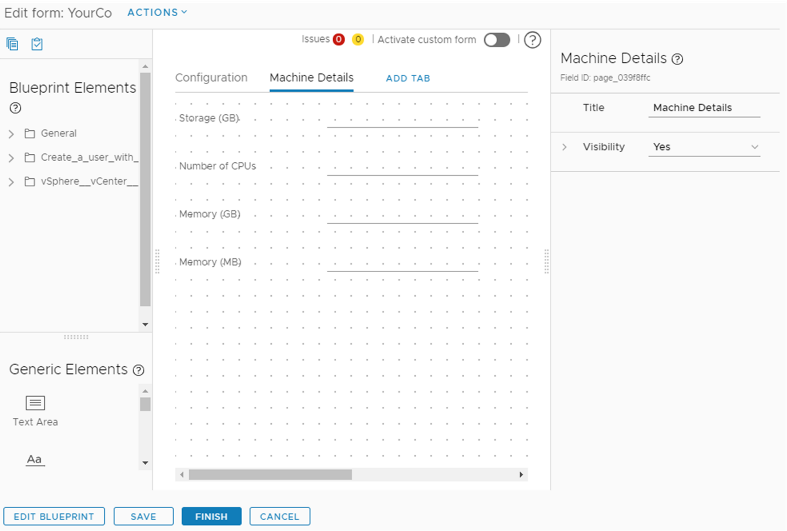
Field in Screenshot	Blueprint Element Source	Appearance	Values	Constraints
Add Active Directory account check box	Generic Elements > Checkbox	Label and Type ■ Label = Add Active Directory account. ■ Display type = Checkbox Visibility ■ Value source = Constant ■ Visible = Yes		
Username	Blueprint Elements > Create a user with a password in a group > The account name for the user	Label and type ■ Label = Username ■ Display type = Text field Visibility **Note** This visibility property, configured the same way in the subsequent fields, hides the field unless the Add Active Directory account check box is selected. ■ Value source = Conditional value ■ Expression = Set value = Yes If Add Active Directory account Equals Yes Custom help ■ Signpost help = Provide the administrator user name.	Default value ■ Value source = Constant ■ Default value = admin	Required ■ Value source = Constant ■ Required = Yes Regular expression **Note** The regular expressions must follow the JavaScript syntax. ■ Value source = Constant ■ Regular expression = "^[a-z]*$" ■ Validation error message = Your user name must not contain any special characters or numbers.

Field in Screenshot	Blueprint Element Source	Appearance	Values	Constraints
Password	Blueprint Elements > Create a user with a password in a group > The password to set for the newly created account	Label and type ■ Label = Password ■ Display type = Password Visibility ■ Value source = Conditional value ■ Expression = Set value = Yes If Add Active Directory account Equals Yes Custom help ■ Signpost help = Provide the password for your administrator account.		Required ■ Value source = Constant ■ Required = Yes Regular expression ■ Value source = Constant ■ Regular Expression = "^(? = .*[A-Z])(? = .*[0-9])(? = .*[a-z]).{8,}$" ■ Message = Your administrator password must be at least eight characters and can include alphanumeric and special characters.
Confirm password	Blueprint Elements > Create a user with a password in a group > Confirmation of the password	Label and type ■ Label = Confirm password Display type = Password Visibility ■ Value source = Conditional value ■ Expression = Set value to Yes If Add Active Directory account Equals Yes Custom help ■ Signpost help = Reenter the password for your administrator account.		Required ■ Value source = Constant ■ Required = Yes Match field ■ Match field = Password

Field in Screenshot	Blueprint Element Source	Appearance	Values	Constraints
Email	Generic Elements > Text Field	Label and type ■ Label = Email ■ Display type = Text Field Visibility ■ Value source = Conditional value ■ Expression = Set value = Yes If Add Active Directory account Equals Yes Custom help ■ Signpost help = Provide the administrator email.	Default value ■ Value source = Computed value ■ Operator = Concatenate ■ Add value = Field. Select Username ■ Add value = Constant. Enter @yourco.com	Regular expression ■ Value source = Constant ■ Regular expression = "^[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\\.[A-Za-z]{2,}$" ■ Validation error message = Provide valid email.
I have read and accept the software policy check box.	Generic Elements > Checkbox	Label and type ■ Element label = I have read and accept the software policy ■ Display type = Checkbox Visibility ■ Value source = Conditional value ■ Expression = Set value = Yes If Add Active Directory account Equals Yes		

5 Click **Add Tab** and entering `Machine Details` in the **Title** property to the right.

6 Configure the following fields in the Machine Details tab.



Machine Details
Field ID: page_039f8ffc

Title: Machine Details

Visibility: Yes

Use the provided Appearance, Values, and Constraints values.

Field in Screenshot	Blueprint Elements Source	Appearance	Values	Constraints
Storage (GB)	Blueprint Elements > vSphere_vCenter_Machine > Storage (GB)	Label and type ■ Label = Storage (GB) ■ Display type = Integer Visibility ■ Value source = Constant ■ Visibility = Yes Read-only ■ Value source = Constant ■ Read-only = No	Default value ■ Value source = Constant ■ Default value = 4	Minimum value ■ Value source = Constant ■ Minimum value = 2
Number of CPUs	Blueprint Elements > vSphere_vCenter_Machine > CPUs	Label and type ■ Label = Number of CPUs ■ Display type = Integer Visibility ■ Value source = Constant ■ Visibility = Yes	Default value ■ Value source = Constant ■ Default value = 1	Minimum value ■ Value source = Constant ■ Min value = 1

Field in Screenshot	Blueprint Elements Source	Appearance	Values	Constraints
Memory (GB)	Generic Elements > Integer	Label and type ■ Label = Memory (GB) ■ Display type = Integer Visibility ■ Value source = Constant ■ Visibility = Yes	Default value ■ Value source = Constant ■ Default value = 1	Minimum value ■ Value source = Constant ■ Minimum value = 1
Memory (MB)	Blueprint Elements > vSphere_vCenter_Machine > Memory (MB)	Label and type ■ Label = Memory (MB) ■ Display type = Integer Visibility ■ Value source = Constant ■ Visibility = No	Default value ■ Value source = Computed value ■ Operator = Multiply ■ Add Value = Field. Select Memory (GB) ■ Add Value = Constant. Enter 1024	

7 Resolve any errors. You can save the form but you cannot activate it until the form is free of errors.

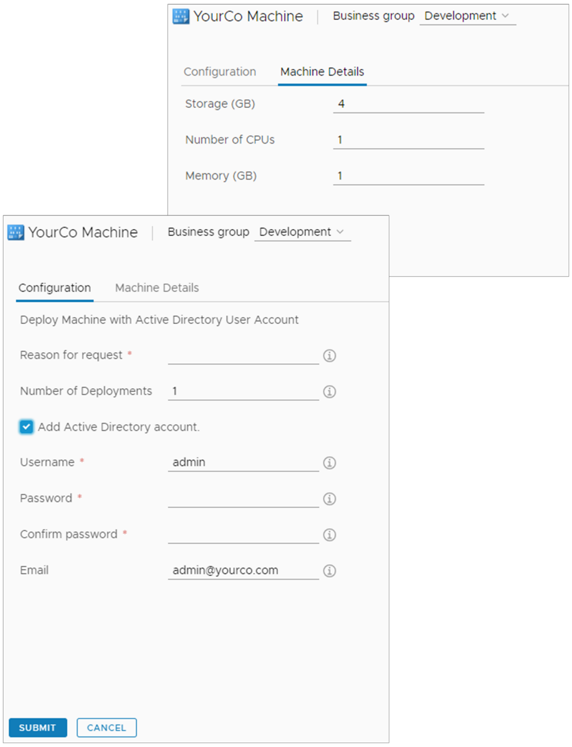
8 To save the form and close the form designer, click **Finish**.

9 Select the blueprint and click **Publish**.

10 To make the custom form available when users request the item in the service catalog, on the Blueprints page toolbar, select **Custom Form > Activate**.

What to do next

- Make the blueprint available in the service catalog. See Managing the Service Catalog.
- In the catalog, verify that the request form is similar to the following example.



Storage (GB) 4

Number of CPUs 1

Memory (GB) 1

Deploy Machine with Active Directory User Account

Reason for request *

Number of Deployments 1

☑ Add Active Directory account.

Username * admin

Password *

Confirm password *

Email admin@yourco.com

Custom Form Designer Field Properties

The fields properties determine how the selected field looks and what default values are presented to the user. And they determine what rules you want to apply to the field to ensure that the user provides a valid entry in the catalog request form in vRealize Automation.

You configure each field individually. Select the field and edit the field properties.

Field Appearance

You use the appearance properties to determine whether the field appears on the form and what label and custom help you want to provide to your catalog users.

Some blueprints might include fields that contain a fixed value. When you add this type of fields to a custom form, only the Appearance options are available, and the field is always Read-only.

Table 5-66. Appearance Tab Options

Option	Description
Label and type	Provide a label and select a display type. The available display types depend on the field. Some fields support multiple text types, some support a few types, and some only support a single type. Possible values across all types: ■ Combo Box ■ Decimal ■ Drop Down ■ Dual list ■ Image ■ Integer ■ Link ■ Multi Select ■ Multi Value Picker ■ Password ■ Radio Group ■ Text ■ Text Area ■ Text Fields The multi select and dual list field types provide the same functionality, with the dual list providing a more intuitive option when the user can select more than one item in a list. Drop-down and data grid fields include a **Placeholder** setting. The entered value appears as an internal label or instructions in the drop-down menu, or as a general label or instructions in the data grid. The value picker and tree picker fields include a **Reference type** setting. The reference type is the vRealize Orchestrator resource type that is used to limit the value or tree picker search to the vRealize Orchestrator server inventory that supports the type. You can then further limit the search by selecting an action that supports the reference type. For more information about the two pickers, see Using the Value Picker or Tree Picker Elements in the Custom Forms Designer.
Visibility	Show or hide a field on the request form. ■ **Constant.** Select Yes to display the field on the form. Select No to hide the field. ■ **Conditional value.** Visibility is determined by the first expression that is true. For example, a field is visible if a check box is selected on a form. ■ **External source.** Visibility is determined by the results of the selected vRealize Orchestrator action.

Table 5-66. Appearance Tab Options (continued)

Option	Description
Read-only	Prevent users from changing the field values. ■ **Constant.** Select Yes to display the value but prevent changes. Select No to allow changes. ■ **Conditional value.** Status is determined by the first expression that is true. For example, a field is read-only if the value in a storage field is greater than 2 GB. ■ **External source.** Status is determined by the results of the selected vRealize Orchestrator action.
Rows per page	For data grid elements only. Enter the number of rows.
Custom help	Provide information about the field to your users. This information appears in signpost help for the field. You can use simple text or HTML, including href links. For example, `<a href="https://docs.vmware.com/en/vRealize-Automation/index.html">vRealize Automation documentation</a>`.

Field Values

You use the values properties to provide any default values.

Table 5-67. Values Tab Options

Option	Description
Columns	For the data grid element only. Provide the label, ID, and value type for each column in your table. The default value for the data grid must include the header data that matches the defined columns. For example, if you have user_name ID for one column and user_role ID for another, then the first row is user_name,user_role. For configuration examples, see Using the Data Grid Element in the Custom Forms Designer.
Default value	Populates the field with a default value based on the value source. For many of the properties, you can select from various value source options. Not all source options are available for all field types or properties. Possible value sources depend on the field. ■ **Constant.** The entered string. The value does not change. Depending on the property, the value might be a string, an integer, a regular expression, or selected from a limited list, for example Yes or No. For example, you can provide 1 as a default value integer, select No for the Read-only property, or provide the regular expression to validate a field entry. ■ **Conditional value.** The value is based on one or more conditions. The conditions are processed in the order listed. If more than one condition is true, the last condition that is true determines the behavior of the field for that property. For example, you can create a condition that determines if a field is visible based on the value in another field. For example, the default value of a storage field is 1 GB if the memory field is less than 512 MB. The `contains` operator checks that the selected field contains the provided value. The `within` operator checks that the selected fields has the provided string. For example, if the expression is **`Field A within development`**, then the expression is true if Field A = dev or lop or ment, but it is evaluated as false if Field A = prod or test. ■ **External source.** The value is based on the results of a vRealize Orchestrator action. For example, calculate cost based on a scripted vRealize Orchestrator action. For an example, see Using vRealize Orchestrator Actions in the Custom Forms Designer. ■ **Bind field.** The value is the same as the selected field to which is it bound. The available fields are limited to the same field type.

Table 5-67. Values Tab Options (continued)

Option	Description
	For example, you bind default value for an authentication needed check box field to another check box field. When one target field check box is selected in the request form, the check box on the current field is selected. ■ **Computed value.** Value is based on the results of the provided field values and the selected operator. Text fields use the concatenate operator. Integer fields use the selected add, subtract, multiply or divide operations. For example, you can configure an integer field to convert megabytes to gigabytes using the multiply operation. The default value of memory in MB is based on the memory in GB multiplied by 1024.
Value option	Populates a drop-down, multi-select, radio group, or value picker fields. ■ **Constant.** The format for the list is Value|Label,Value|Label,Value|Label. For example, `2|Small,4|Medium,8|Large`. ■ **External source.** Value is based on the results of the selected vRealize Orchestrator action.
Step	For integer or decimal fields, define the incremental or decremental values. For example, if the default value is 1 and you set the step value to 3, then the allowed values are 4, 7, 10, and so on.

Field Constraints

You use the constraint properties to ensure that the requesting user provides valid values in the request form.

You might also use external validation as an alternative method for ensuring valid values. See Using External Validation in the Custom Forms Designer.

Table 5-68. Constraints Tab Options

Option	Description
Required	The requesting user must provide a value for this field. ■ **Constant.** Select Yes to require that the requesting user provides a value. Select no if the field is optional. ■ **Conditional value.** Whether the field is required is determined by the first expression that is true. For example, this field is required if the operating system family starts with Darwin in another field. ■ **External source.** Status is based on the results of the selected vRealize Orchestratoraction.
Regular expression	Provide a regular expression that validates the value and a message that appears when the validation fails. The regular expressions must follow JavaScript syntax. For an overview, see Creating a regular expression. For more detailed guidance, see Syntax. ■ **Constant.** Provide a regular expression. For example, for an email address, the regular expression might be `^[A-Za-z0-9._%+-]+@[A-Zaz0-9.-]+\\.[A-Za-z]{2,}$` and the validation error message is `The email address format is not valid. Please try again.` ■ **Conditional value.** The regular expression that is used is determined by the first expression that is true.
Minimum value	Specify a minimum numeric value. For example, a password must have at least 8 characters. Provide an error message. For example, `The password must be at least 8 characters.` ■ **Constant.** Enter the integer. ■ **Conditional value.** The minimum value is determined by the first expression that is true. For example, a minimum CPU value is 4 if the operating system does not equal Linux. ■ **External source.** Value is based on the results of the selected vRealize Orchestrator action.
Maximum value	Maximum numeric value. For example, a field is limited to 50 characters. Provide an error message. For example, `This description cannot exceed 50 characters.` ■ **Constant.** Enter the integer. ■ **Conditional value.** The maximum value is determined by the first expression that is true. For example, a maximum storage value is 2 GB if the deployment location equals AMEA. ■ **External source.** Value is based on the results of the selected vRealize Orchestrator action.
Match field	This field value must match the selected field value. For example, a password confirmation field must match the password field.

Using vRealize Orchestrator Actions in the Custom Forms Designer

When you customize the request form for a vRealize Automation blueprint, you can base the behavior of some fields on the results of a vRealize Orchestrator action.

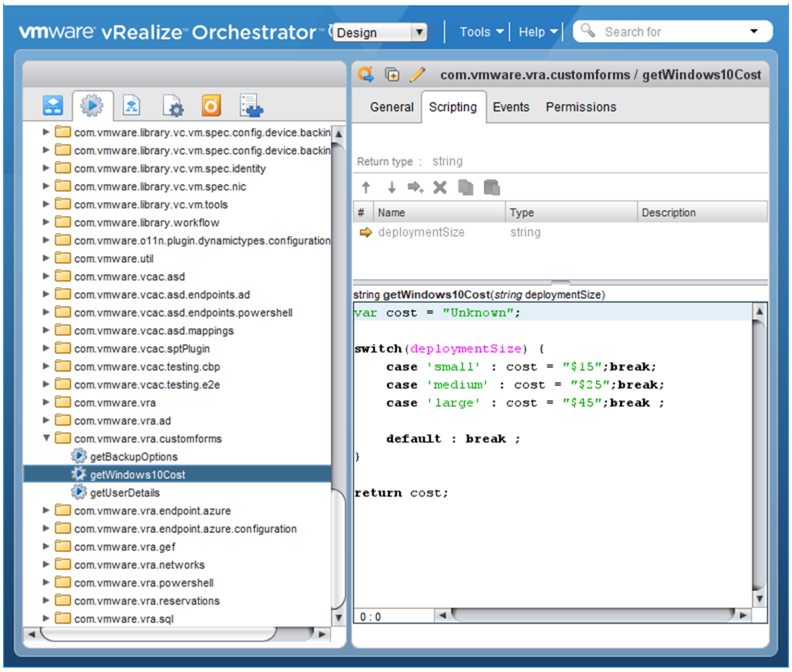
There are several ways that you can use vRealize Orchestrator actions. You might have an action that pulls the data from a third source, or you can use a script that defines the size and cost. This example uses a script.

When you create a script to populate fields using an action, do not use an Array [Any] type.

Example: Size and Cost Fields Example

In this use case, you want the catalog user to select a virtual machine size, and then display the cost of that machine per day. To do this example, you have a vRealize Orchestrator that correlates the size and cost, and you add a size field and a cost field to the blueprint custom form. The size field determines the value that appears in the cost field.

1 In vRealize Orchestrator, configure an action, `getWindows10Cost`, with a `deploymentSize` script similar to the following example.


vmware vRealize Orchestrator
Design
Tools
Help
Search for
com.vmware.library.vc.vm.spec.config.device.backin
com.vmware.library.vc.vm.spec.config.device.backin
com.vmware.library.vc.vm.spec.identity
com.vmware.library.vc.vm.spec.nic
com.vmware.library.vc.vm.tools
com.vmware.library.workflow
com.vmware.o11n.plugin.dynamictypes.configuration
com.vmware.util
com.vmware.vcac.asd
com.vmware.vcac.asd.endpoints.ad
com.vmware.vcac.asd.endpoints.powershell
com.vmware.vcac.asd.mappings
com.vmware.vcac.sptPlugin
com.vmware.vcac.testing.cbp
com.vmware.vcac.testing.e2e
com.vmware.vra
com.vmware.vra.ad
com.vmware.vra.customforms
getBackupOptions
getWindows10Cost
getUserDetails
com.vmware.vra.endpoint.azure
com.vmware.vra.endpoint.azure.configuration
com.vmware.vra.gef
com.vmware.vra.networks
com.vmware.vra.powershell
com.vmware.vra.reservations
com.vmware.vra.sql
com.vmware.vra.customforms / getWindows10Cost
General
Scripting
Events
Permissions
0 : 0

Return type : string

#	Name	Type	Description
	deploymentSize	string	

string getWindows10Cost(*string* deploymentSize)

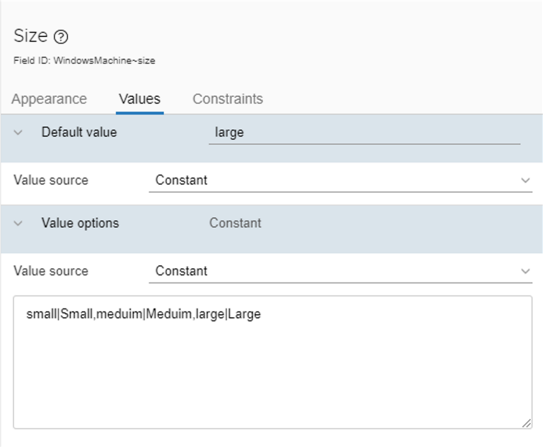
```
var cost = "Unknown";

switch(deploymentSize) {
    case 'small' : cost = "$15";break;
    case 'medium' : cost = "$25";break;
    case 'large' : cost = "$45";break ;

    default : break ;
}

return cost;
```

Use the following as a script example.

```
var cost = "Unknown";

switch(deploymentSize) {
    case 'small' : cost = "$15";break;
    case 'medium' : cost = "$25";break;
    case 'large' : cost = "$45";break ;
    default : break ;
}

return cost;
```

2 In vRealize Automation, add and configure a size field and cost field to a blueprint custom form.

Configure the size field as multi select with Small, Medium, and Large values.



Size

Field ID: WindowsMachine~size

Default value: large

Value source: Constant

Value options: Constant

Value source: Constant

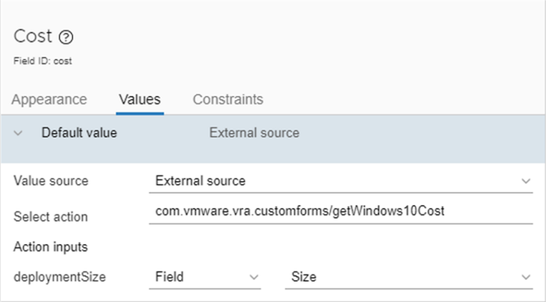
small|Small,meduim|Meduim,large|Large

In vRealize Automation, add and configure a size field and cost field to a blueprint custom form.

On the Values tab, configure the following property values.

- Default value = **`Large`**
- Value options
  - Value source = **`Constant`**
  - Value definition = **`small|Small,medium|Medium,large|Large`**

3 Configure the cost field to display the cost as defined in the vRealize Orchestrator action based on the value selected in the size field.



Cost

Field ID: cost

Default value: External source

Value source: External source

Select action: com.vmware.vra.customforms/getWindows10Cost

Action inputs

deploymentSize: Field, Size

On the Values tab, configure the following property values.

- Default value = External source
- Select action = <your vRealize Orchestrator actions folder>/getWindows10Cost

- Action inputs
  - deploymentSize. This value was configured in the action.
  - Field
  - Size

Using the Value Picker or Tree Picker Elements in the Custom Forms Designer

When you customize the request form, you can provide elements where the user can select from search results from a list or browse a tree to locate a matching value.

The value picker and the tree picker work with the Reference Type that is defined on the custom form Appearance tab. The Reference type is a vRealize Orchestrator resource. For example, AD:UserGroup or VC:Datastore. By defining the reference type, when the user enters a search string, the results or tree options are limited to or the resources that have the matching parameter.

For the value picker, you can then further limit the possible values by configuring an external source. For the tree picker, you can provide a default value by configuring an external source.

Working with the Value Picker

The value picker appears in the catalog form as a search option. The user enters a string and the picker provides options based on how you configured it. You can use the picker based on the following use cases. The most valuable use of the value picker is pairing it with an external source value.

- Value picker with a constant value source. Use this method when you want to the requesting user to select from a predefined static list of values. Similar to the combobox, drop down, multiselect, and radio group elements, this method provides search results in a list based on the defined constant values and labels.
- Value picker with no defined value source. Use this method when you want the requesting user to search the vRealize Orchestrator inventory for a specific object with the configured reference type. For example, the reference type is VC:Datastore and you want the users to select the datastore from the retrieved list.
- Value picker with an external value source. Use this method when you want the requesting user to select from results that are based on a vRealize Orchestrator action. For a value picker external source the action must return an properties array, not a sting array. For example, you have an action that retrieves two or more values from an integrated database and you want the users to select a value from the retrieved list. The action must include the filter `var filter = System.getContext().getParameter("__filter");` and must return a properties array, not a string array. If you want a string array, use the combo box field type.

Working with the Tree Picker

The tree picker appears in the catalog form as a search option. The user enters a string and the tree picker appears. The tree allows the users to select values that match the reference type. For example, if the reference type is VC:Datastore, the requesting user can select datastore objects. If the reference type is VC:VirtualMachine, the user can select virtual machines.

- Tree picker with no defined value source. Use this method when you want the requesting user to browse the hierarchical tree for a specific object with the configured reference type. For example, the reference type is VC:Datastore and you want the users to select a datastore from the retrieved tree.
- Tree picker with an external value source. Use this method when you want to provide a default selection in the tree. The requesting user can select the preset value or browse for a different value. For example, for reference type VC:Datastore, you want to preset the datastore in the tree to a particular datastore based on the results of the action input value that specifies a network.

Using the Data Grid Element in the Custom Forms Designer

When you customize the request form for a blueprint, you might add information in a table format. The requesting user can populate the rows with data that is included in the provisioning request.

You can add a table and populate it based on manually provided data or based on an external source. Some blueprint elements appear as a data grid. For example, virtual machine disks or NICs.

In addition to adding fields to the data grid, you can also add constraints ensure that the user provides acceptable values.

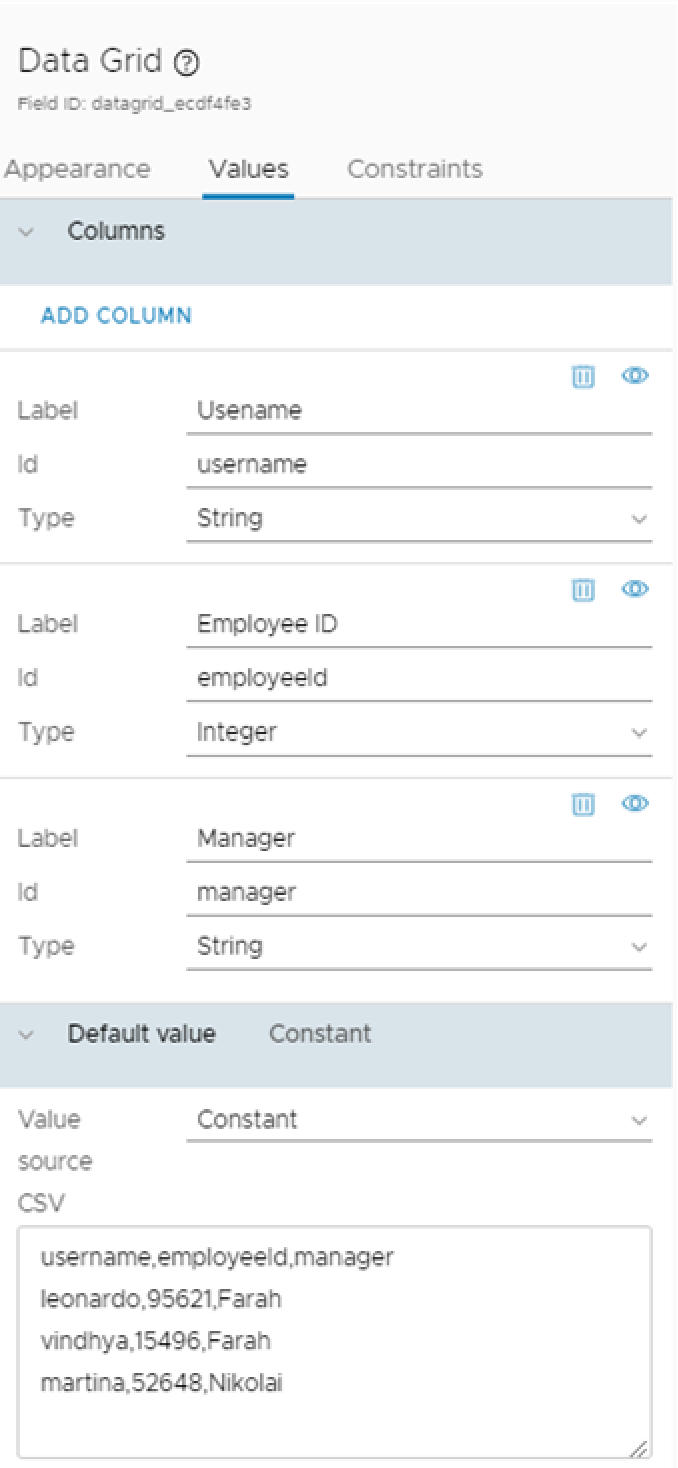
The following examples use the data grid, but you can use the Multi Value Picker as alternative way of presenting the option to your users in the request form. You can test the differences by changing the **Appearance > Label and type > Display type** field property.

Example: Provided CSV Data Example

In this example, you have a table of values that you provide in the custom request form so that they can . You provide the information in the table as a constant value source. The source is based on a CSV data structure where the first row header. The headers are the column IDs separated by a comma. Each additional row is the data that appears in each row in the table.

1 Add the Data Grid generic element to the design canvas.

2 Select the data grid and define the values in the properties pane.



Data Grid

Field ID: datagrid_ecdf4fe3

Columns

Label	Usename
Id	username
Type	String

Label	Employee ID
Id	employeeId
Type	Integer

Label	Manager
Id	manager
Type	String

Default value Constant

Value source: Constant

CSV

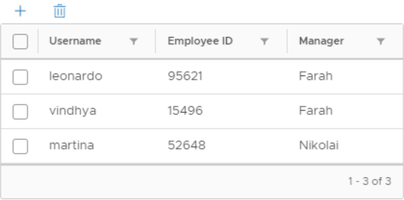
```
username,employeeId,manager
leonardo,95621,Farah
vindhya,15496,Farah
martina,52648,Nikolai
```

Label	ID	Type
Username	username	String
Employee ID	employeeId	Integer
Manger	manager	String

Define the CSV values.

```
username,employeeId,manager
leonardo,95621,Farah
vindhya,15496,Farah
martina,52648,Nikolai
```

3 Verify that the data grid displays the expected data in the blueprint request form.



□	Username	Employee ID	Manager
□	leonardo	95621	Farah
□	vindhya	15496	Farah
□	martina	52648	Nikolai

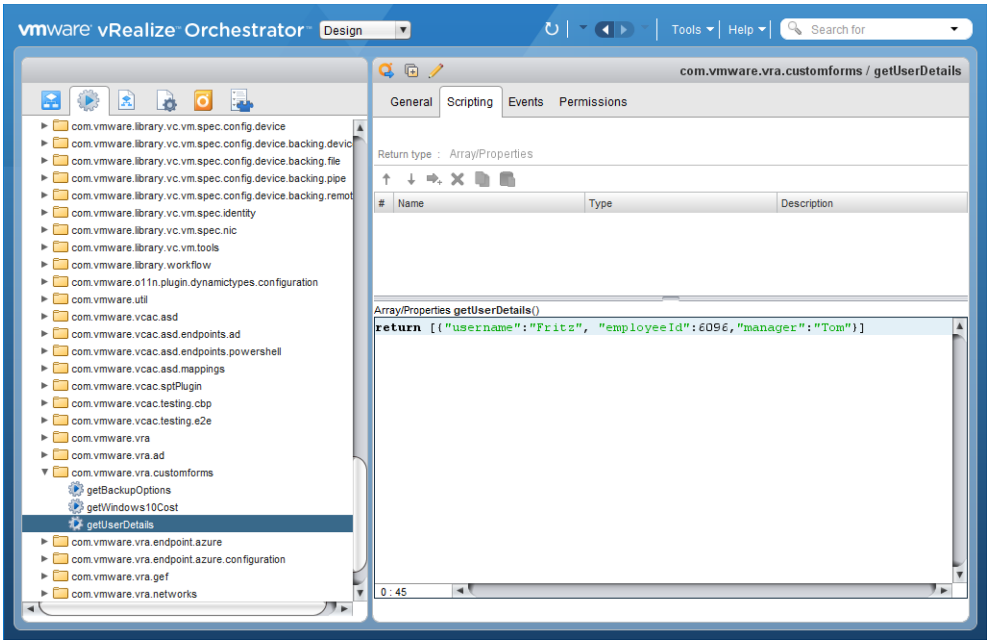
1 - 3 of 3

Example: External Source Example

This example uses the previous example but the values are based on a vRealize Orchestrator action. Although this is a simple action example, but you can use a more complex action that you retrieve this information from a local database or system.

The action that you use as validation must have an Array/Properties input parameter.

1 In vRealize Orchestrator, configure an action, `getUserDetails`, with an array similar to the following example.



vmware vRealize Orchestrator
Design
Tools
Help
Search for
com.vmware.vra.customforms / getUserDetails
General
Scripting
Events
Permissions
Return type : Array/Properties
#
Name
Type
Description
com.vmware.library.vc.vm.spec.config.device
com.vmware.library.vc.vm.spec.config.device.backing.devic
com.vmware.library.vc.vm.spec.config.device.backing.file
com.vmware.library.vc.vm.spec.config.device.backing.pipe
com.vmware.library.vc.vm.spec.config.device.backing.remot
com.vmware.library.vc.vm.spec.identity
com.vmware.library.vc.vm.spec.nic
com.vmware.library.vc.vm.tools
com.vmware.library.workflow
com.vmware.o11n.plugin.dynamictypes.configuration
com.vmware.util
com.vmware.vcac.asd
com.vmware.vcac.asd.endpoints.ad
com.vmware.vcac.asd.endpoints.powershell
com.vmware.vcac.asd.mappings
com.vmware.vcac.sptPlugin
com.vmware.vcac.testing.cbp
com.vmware.vcac.testing.e2e
com.vmware.vra
com.vmware.vra.ad
com.vmware.vra.customforms
getBackupOptions
getWindows10Cost
getUserDetails
com.vmware.vra.endpoint.azure
com.vmware.vra.endpoint.azure.configuration
com.vmware.vra.gef
com.vmware.vra.networks
0 : 45

Array/Properties getUserDetails()

```
return [{"username":"Fritz", "employeeId":6096,"manager":"Tom"}]
```

Use the following script example.

```
return [{"username":"Fritz", "employeeId":6096,"manager":"Tom"}]
```

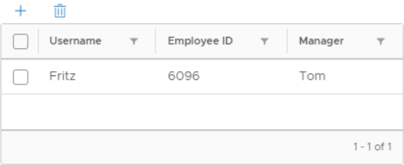
2 In vRealize Automation, add the data grid and configure the data grid columns with the following values.

Label	ID	Type
Username	username	String
Employee ID	employeeId	Integer
Manger	manger	String

3 In the Value source list, select **External source**.

4 In Select action, enter getUserDetails and select the action you created in vRealize Orchestrator.

5 Save and verify the table in the request form.



□	Username	Employee ID	Manager
□	Fritz	6096	Tom

1 - 1 of 1

Example: Blueprint Element Example

Some blueprint elements can be added to the form and appear as a data grid when the user requests the blueprint. Disks and NICs appear as data grids.

In this example, you add a disks element to the form so that your users can add additional disks when they request the catalog item. You can add constraints to better control what the user can request. For example, you can limit the capacity to 5 GB.

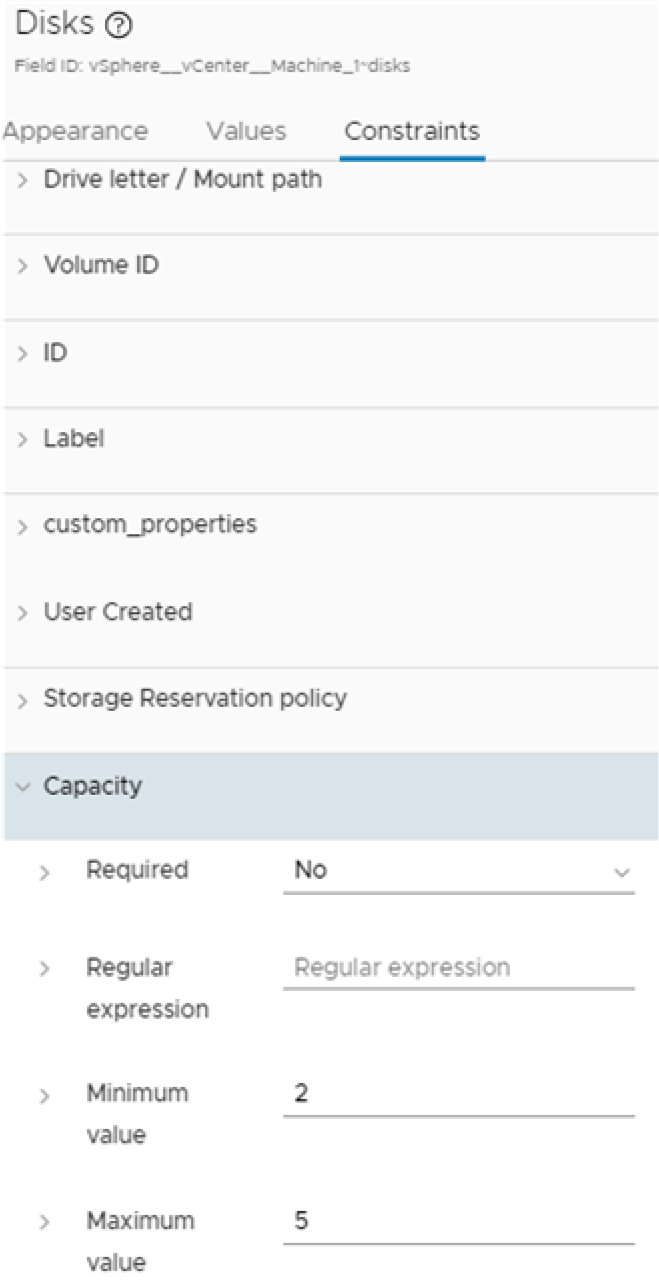
The element values defined in the blueprint, for example, disks, are not visible in the custom form. This prevents the user from modifying a configuration that is required for the successful provisioning of the request.

1 Create a blueprint with a machine with a defined 6 GB storage disk.

2 Add the Disk element to the canvas.

3 Select the data grid and define the constraints in the properties pane.

In this example, the capacity minimum is set to 2 and the maximum is 5.



Disks

Field ID: vSphere__vCenter__Machine_1~disks

Required: No

Regular expression: Regular expression

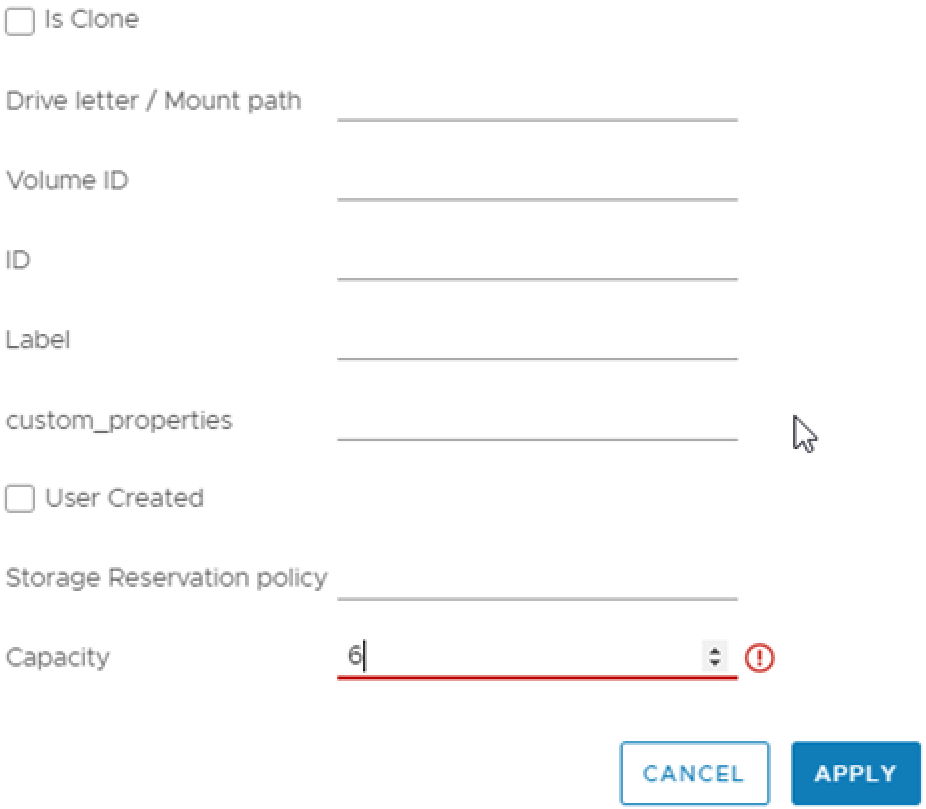
Minimum value: 2

Maximum value: 5

4 Save and verify the table constraints in the request form.

5 In the request form, click the plus sign on the data grid.

Notice that the capacity constraint is triggered if you enter a value greater than 5.



□ Is Clone

Drive letter / Mount path

Volume ID

ID

Label

custom_properties

□ User Created

Storage Reservation policy

Capacity 6

Using External Validation in the Custom Forms Designer

You can customize a request form to ensure that users provide valid values at request time by adding constraints to fields or using an external validation source.

Some field properties, such as minimum, maximum, regular expressions, match fields, or not empty, can be configured with constraints to ensure valid values. See Custom Form Designer Field Properties.

External validation checks for valid values from an external source using vRealize Orchestrator actions.

If you are validating a data grid value, the action that you use as validation must have an Array/Properties input parameter.

Some examples where you might use external validation includes:

- The valid values are defined in an external source. For example, vRealize Orchestrator.
- The validation must affect several fields. For example, a vRealize Orchestrator action collects the disk size and storage pool capacity, and validates the provided size values based on available space.

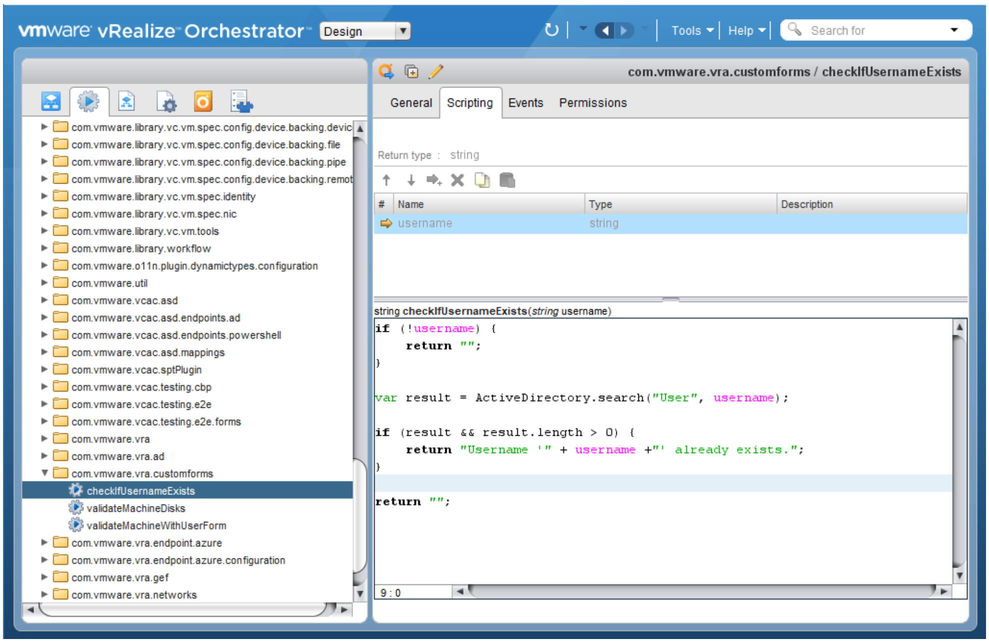
How do you order multiple external validations in one blueprint? The validations are processed in the order that the appear on the External Validation canvas. If you have two validations that validate the same field, the second validation results will overwrite the first. To reorder the validations, you can click and drag the cards on the canvas.

Example: vRealize Orchestrator User Example

In this use case, you want the catalog user to provide only a new user name. To do this example, you have a vRealize Orchestrator action that checks whether the user name provided in the form exists in your Active Directory database. If the name does exist, an error message appears on the request form.

This use case is applied to the Create a Custom Request Form with Active Directory Options example.

1 In vRealize Orchestrator, configure an action, `checkIfUsernameExists`, with a script similar to the following example.


vmware vRealize Orchestrator
Design
Tools
Help
Search for
com.vmware.vra.customforms / checkIfUsernameExists
General
Scripting
Events
Permissions
com.vmware.library.vc.vm.spec.config.device.backing.devic
com.vmware.library.vc.vm.spec.config.device.backing.file
com.vmware.library.vc.vm.spec.config.device.backing.pipe
com.vmware.library.vc.vm.spec.config.device.backing.remot
com.vmware.library.vc.vm.spec.identity
com.vmware.library.vc.vm.spec.nic
com.vmware.library.vc.vm.tools
com.vmware.library.workflow
com.vmware.o11n.plugin.dynamictypes.configuration
com.vmware.util
com.vmware.vcac.asd
com.vmware.vcac.asd.endpoints.ad
com.vmware.vcac.asd.endpoints.powershell
com.vmware.vcac.asd.mappings
com.vmware.vcac.sptPlugin
com.vmware.vcac.testing.cbp
com.vmware.vcac.testing.e2e
com.vmware.vcac.testing.e2e.forms
com.vmware.vra
com.vmware.vra.ad
com.vmware.vra.customforms
checkIfUsernameExists
validateMachineDisks
validateMachineWithUserForm
com.vmware.vra.endpoint.azure
com.vmware.vra.endpoint.azure.configuration
com.vmware.vra.gef
com.vmware.vra.networks

Return type : string

#	Name	Type	Description
	username	string	

string checkIfUsernameExists(*string* username)

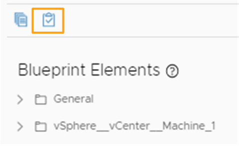
```
if (!username) {
    return "";
}

var result = ActiveDirectory.search("User", username);

if (result && result.length > 0) {
    return "Username '" + username +"' already exists.";
}

return "";
```

9 : 0

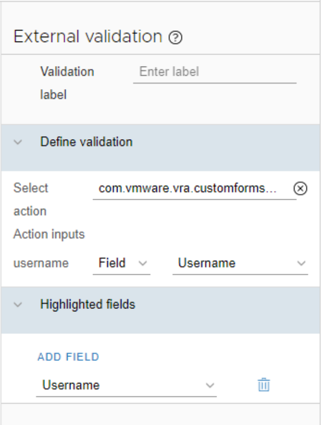
Use the following as a script example. In this example, `return` is the message that appears if the validation fails.

```
if (!username) {
    return "";
}

var result = ActiveDirectory.search("User", username);

if (result && result.length > 0) {
    return "Username '" + username +"' already exists.";
}

return "";
```

2 In vRealize Automation, open the custom form designer for your blueprint, click **External Validation**, and drag the **Orchestrator validation** type onto the canvas.



3 Configure the external validation options.



Validation label: Enter label

Select action: com.vmware.vra.customforms...

Action inputs

username: Field, Username

Highlighted fields: Username

- Validation label = Check if user name exists
- Select action = <your vRealize Orchestrator actions folder>/checkIfUsernameExists
- Action inputs
  - username = Field and Username
- Highlighted fields
  - Click **Add Field** and select Username.

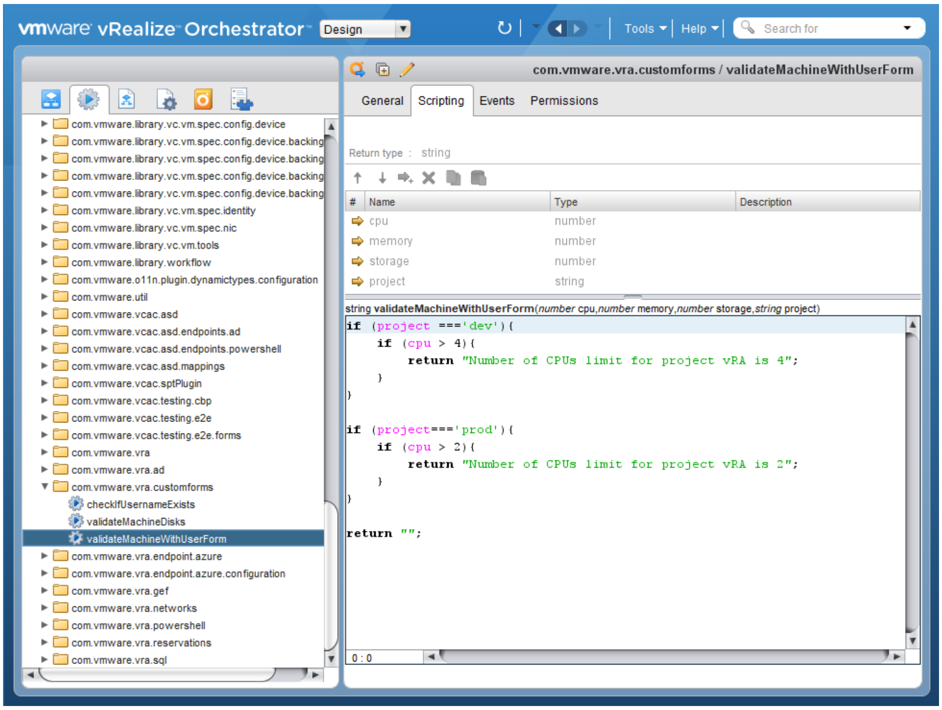
A field-level validation error appears in the catalog request form if the entered valued fails validation. If you want a global error, do not configure the highlighted field.

Example: vRealize Orchestrator Multiple Fields Example

In this use case, you want to base the validation of the CPU, memory, and storage values on the project value. For example, if the users select the Dev project, then the maximum number of CPUs is 4. If they select Prod, then the maximum value is 2.

For this use case, add a project field to the Create a Custom Request Form with Active Directory Options example. Configure project as a drop-down with Dev and Prod.

1 In vRealize Orchestrator, configure an action, `validateMachineWithUserForm`, with a script similar to the following example.


vmware vRealize Orchestrator
Design
Tools
Help
Search for
com.vmware.vra.customforms / validateMachineWithUserForm
General
Scripting
Events
Permissions
com.vmware.library.vc.vm.spec.config.device
com.vmware.library.vc.vm.spec.config.device.backing
com.vmware.library.vc.vm.spec.config.device.backing
com.vmware.library.vc.vm.spec.config.device.backing
com.vmware.library.vc.vm.spec.config.device.backing
com.vmware.library.vc.vm.spec.identity
com.vmware.library.vc.vm.spec.nic
com.vmware.library.vc.vm.tools
com.vmware.library.workflow
com.vmware.o11n.plugin.dynamictypes.configuration
com.vmware.util
com.vmware.vcac.asd
com.vmware.vcac.asd.endpoints.ad
com.vmware.vcac.asd.endpoints.powershell
com.vmware.vcac.asd.mappings
com.vmware.vcac.sptPlugin
com.vmware.vcac.testing.cbp
com.vmware.vcac.testing.e2e
com.vmware.vcac.testing.e2e.forms
com.vmware.vra
com.vmware.vra.ad
com.vmware.vra.customforms
checkIfUsernameExists
validateMachineDisks
validateMachineWithUserForm
com.vmware.vra.endpoint.azure
com.vmware.vra.endpoint.azure.configuration
com.vmware.vra.gef
com.vmware.vra.networks
com.vmware.vra.powershell
com.vmware.vra.reservations
com.vmware.vra.sql
0 : 0

Return type : string

#	Name	Type	Description
⇨	cpu	number	
⇨	memory	number	
⇨	storage	number	
⇨	project	string	

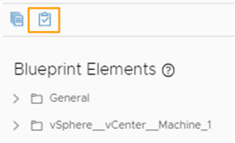
string validateMachineWithUserForm(*number* cpu,*number* memory,*number* storage,*string* project)

```
if (project ==='dev'){
    if (cpu > 4){
        return "Number of CPUs limit for project vRA is 4";
    }
}

if (project==='prod'){
    if (cpu > 2){
        return "Number of CPUs limit for project vRA is 2";
    }
}

return "";
```

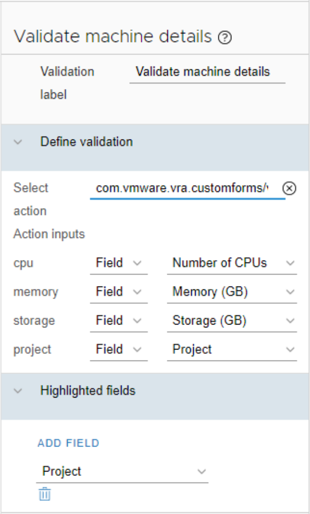
Use the following as a script example for the CPU checking. Continue adding the memory and storage values to the script, as needed. In this example, return is the message that appears if validation fails.

```
if (project ==='dev'){
    if (cpu > 4){
        return "Number of CPUs limit for project vRA is 4";
    }
}

if (project==='prod'){
    if (cpu > 2){
        return "Number of CPUs limit for project vRA is 2";
    }
}

return "";
```

2 In vRealize Automation, open the custom form designer for your blueprint, click **External Validation**, and drag the **Orchestrator validation** type onto the canvas.



3 Configure the external validation options.



Validate machine details

Validation label: Validate machine details

Define validation

Select action: com.vmware.vra.customforms/

Action inputs

cpu	Field	Number of CPUs
memory	Field	Memory (GB)
storage	Field	Storage (GB)
project	Field	Project

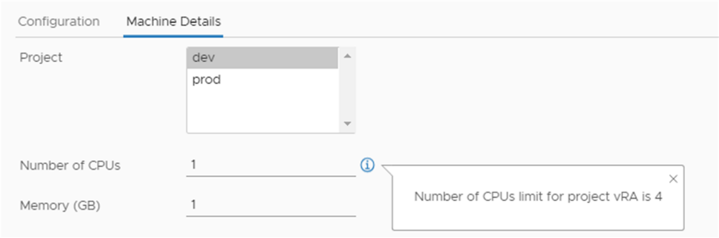
Highlighted fields

ADD FIELD

Project

- Validation label = Validate machine details
- Select action = <your vRealize Orchestrator actions folder>/validateMachineWithUserForm
- Action inputs
  - cpu = Field and Number of CPUs
  - memory = Field and Memory (GB)
  - storage = Field and Storage (GB)
  - Project = Field and Project
- Highlighted fields
  - Click **Add Field** and select `Project`.

In the catalog, your catalog user might see a validation error similar to the following example.



Project: dev, prod

Number of CPUs: 1

Memory (GB): 1

Number of CPUs limit for project vRA is 4

Testing and Troubleshooting Failed Provisioning Requests

As a blueprint architect or administrator, you want to ensure that you deliver working blueprints to your user.

Catalog requests might fail for several reasons. It might be due to network traffic, insufficient endpoint resources, or a flawed blueprint specification. Or, the provisioning request succeeded, but the deployment does not appear to be working. As a blueprint architect, you want to avoid providing blueprints that your users cannot successfully deploy.

You can create a testing service and entitlement so that you can deploy the blueprint from the catalog. See Checklist for Configuring the Service Catalog.

If the resources are not successfully provisioned, you can use vRealize Automation to troubleshooting the failed deployment.

Possible Failure States

If a provisioning request fails, you see one of the following states.

- **Failed.** A request can fail for several reasons. One cause is that the provisioning process did not work due to a lack of resources on the target endpoint, insufficient resources to support the blueprint, or a badly designed blueprint that must be fixed. Another cause is that the request required approval from someone in your organization, and approver rejected the request. It is also possible that an action that you ran on a deployment failed. The failure can be caused by the environmental or approval reasons already mentioned.

  Use the following troubleshooting workflow to investigate the cause of the problem. If you are able the resolve the problem, review your action options regarding **Dismiss** and **Resubmit**. See Action Menu Commands for Provisioned Resources.

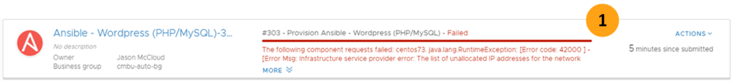
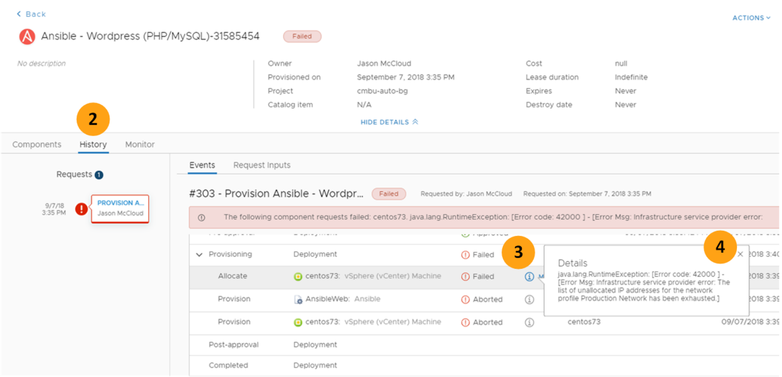
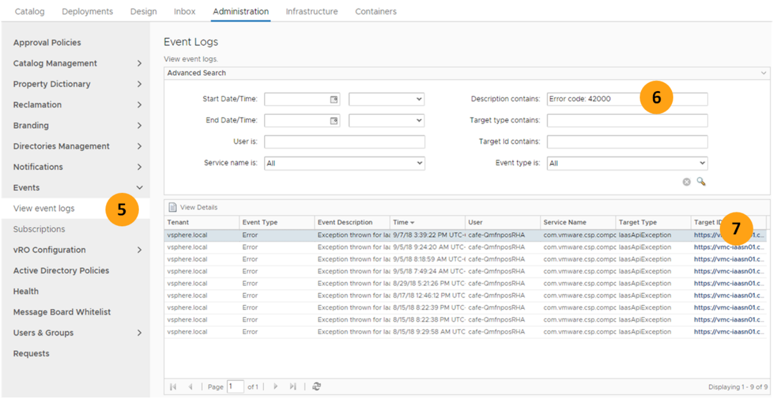
- **Partially Successful.** A request can be partially successful, meaning some components were deployed but not all of the provisioning steps completed successfully.

  Use the following troubleshooting workflow to determine which components were only partially successful and investigate the cause of the problem. If you are able the resolve the problem, review your action options regarding **Dismiss** and whether you can use **Resume**. See Action Menu Commands for Provisioned Resources and How the Resume Action Works.

Troubleshooting Workflow

You can use this workflow to begin investigating a failed deployment. If your investigation reveals that the failure was due to a transient environmental problem, you can resolve the error and resubmit the request. If the problem is with the request specification, you can update the blueprint and submit a new request.

Table 5-69. How to Begin Troubleshooting Errors

Workflow	Troubleshooting Step	Example
1	On the **Deployments** tab, failed deployments are indicated on the status bar. The card includes the last failure message. For more information, click the deployment name or progress bar.	
2	On the deployment details **History** tab, you can use the events workflow to see where the provisioning process failed. This workflow is also useful when you run an action on a deployment, but the change fails.	
3	The failed status indicates where the workflow failed.	
4	The information provides a more verbose version of the error message. If this information in the signpost help is not sufficient to identify and resolve the problem, you can do additional research in the event logs.	
5	The following steps require an administrator role. To locate an error in the context of other errors and warnings, select **Administration > Events > View event logs**.	

Ansible - Wordpress (PHP/MySQL)-3...
No description
Owner Jason McCloud
Business group cmbu-auto-bg
ACTIONS
MORE
1

#303 - Provision Ansible - Wordpress (PHP/MySQL) - Failed

The following component requests failed: centos73. java.lang.RuntimeException: [Error code: 42000] - [Error Msg: Infrastructure service provider error: The list of unallocated IP addresses for the network

5 minutes since submitted

Owner	Jason McCloud	Cost	null
Provisioned on	September 7, 2018 3:35 PM	Lease duration	Indefinite
Project	cmbu-auto-bg	Expires	Never
Catalog item	N/A	Destroy date	Never

#303 - Provision Ansible - Wordpr... Failed Requested by: Jason McCloud Requested on: September 7, 2018 3:35 PM

The following component requests failed: centos73. java.lang.RuntimeException: [Error code: 42000] - [Error Msg: Infrastructure service provider error:

Provisioning	Deployment	Failed	2018 3:40
Allocate	centos73: vSphere (vCenter) Machine	Failed	2018 3:39
Provision	AnsibleWeb: Ansible	Aborted	2018 3:39
Provision	centos73: vSphere (vCenter) Machine	Aborted	centos73 09/07/2018 3:39
Post-approval	Deployment		
Completed	Deployment		

Details
java.lang.RuntimeException: [Error code: 42000] - [Error Msg: Infrastructure service provider error: The list of unallocated IP addresses for the network profile Production Network has been exhausted.]

Event Logs

View event logs.

Advanced Search

Start Date/Time:

End Date/Time:

User is:

Service name is: All

Description contains: Error code: 42000

Target type contains:

Target Id contains:

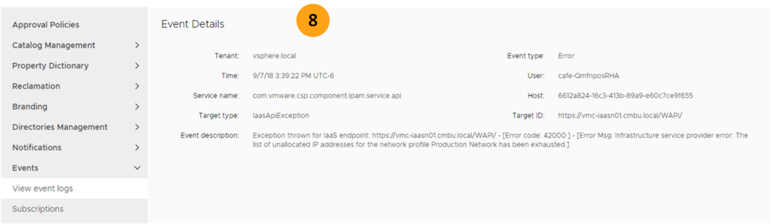
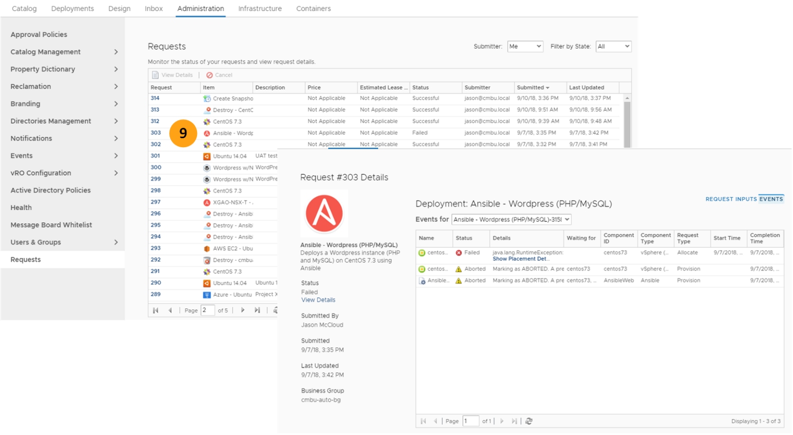
Event type is: All

Tenant	Event Type	Event Description	Time	User	Service Name	Target Type	Target ID
vsphere.local	Error	Exception thrown for Ia:	9/7/18 3:39:22 PM UTC-(	cafe-QmfnposRHA	com.vmware.csp.compc	IaasApiException	https://v...1.c...
vsphere.local	Error	Exception thrown for Ia:	9/5/18 9:24:20 AM UTC-	cafe-QmfnposRHA	com.vmware.csp.compc	IaasApiException	https://vmc-iaasn01.c...
vsphere.local	Error	Exception thrown for Ia:	9/5/18 8:18:59 AM UTC-(	cafe-QmfnposRHA	com.vmware.csp.compc	IaasApiException	https://vmc-iaasn01.c...
vsphere.local	Error	Exception thrown for Ia:	9/5/18 7:49:24 AM UTC-	cafe-QmfnposRHA	com.vmware.csp.compc	IaasApiException	https://vmc-iaasn01.c...
vsphere.local	Error	Exception thrown for Ia:	8/29/18 5:21:26 PM UTC-	cafe-QmfnposRHA	com.vmware.csp.compc	IaasApiException	https://vmc-iaasn01.c...
vsphere.local	Error	Exception thrown for Ia:	8/17/18 12:46:12 PM UTC	cafe-QmfnposRHA	com.vmware.csp.compc	IaasApiException	https://vmc-iaasn01.c...
vsphere.local	Error	Exception thrown for Ia:	8/15/18 8:22:39 PM UTC-	cafe-QmfnposRHA	com.vmware.csp.compc	IaasApiException	https://vmc-iaasn01.c...
vsphere.local	Error	Exception thrown for Ia:	8/15/18 8:22:38 PM UTC-	cafe-QmfnposRHA	com.vmware.csp.compc	IaasApiException	https://vmc-iaasn01.c...
vsphere.local	Error	Exception thrown for Ia:	8/15/18 9:29:58 AM UTC	cafe-QmfnposRHA	com.vmware.csp.compc	IaasApiException	https://vmc-iaasn01.c...

Page 1 of 1

Displaying 1 - 9 of 9

Table 5-69. How to Begin Troubleshooting Errors (continued)

Workflow	Troubleshooting Step	Example
6	You can use the advanced search to locate the error based on the message in the deployment details.	
7	To view the event details, click the Target ID link.	
8	The event details provide additional provisioning information that might assist you with your troubleshooting efforts.	
9	As an administrator, you can also view the request in the context of other requests by your users. Select **Administration > Requests** and click the request number to examine the request inputs and events.	

Event Details

Tenant:	vsphere.local	Event type:	Error
Time:	9/7/18 3:39:22 PM UTC-6	User:	cafe-QmfnposRHA
Service name:	com.vmware.csp.component.ipam.service.api	Host:	6612a824-16c3-413b-89a9-e60c7ce9f655
Target type:	IaasApiException	Target ID:	https://vmc-iaasn01.cmbu.local/WAPI/

Event description: Exception thrown for IaaS endpoint: https://vmc-iaasn01.cmbu.local/WAPI/ - [Error code: 42000] - [Error Msg: Infrastructure service provider error: The list of unallocated IP addresses for the network profile Production Network has been exhausted.]

Requests

Monitor the status of your requests and view request details.

Submitter: Me Filter by State: All

Request	Item	Description	Price	Estimated Lease ...	Status	Submitter	Submitted	Last Updated
314	Create Snapsho		Not Applicable	Not Applicable	Successful	jason@cmbu.local	9/10/18, 3:36 PM	9/10/18, 3:37 PM
313	Destroy - CentC		Not Applicable	Not Applicable	Successful	jason@cmbu.local	9/10/18, 9:51 AM	9/10/18, 9:56 AM
312	CentOS 7.3		Not Applicable	Not Applicable	Successful	jason@cmbu.local	9/10/18, 9:39 AM	9/10/18, 9:48 AM
303	Ansible - Wordp		Not Applicable	Not Applicable	Failed	jason@cmbu.local	9/7/18, 3:35 PM	9/7/18, 3:42 PM
302	CentOS 7.3		Not Applicable	Not Applicable	Successful	jason@cmbu.local	9/7/18, 3:32 PM	9/7/18, 3:41 PM
301	Ubuntu 14.04	UAT test						
300	Wordpress w/N	WordPre						
299	Wordpress w/N	WordPre						
298	CentOS 7.3							
297	XGAO-NSX-T - .							
296	Destroy - Ansib							
295	Destroy - Ansib							
294	Destroy - Ansib							
293	AWS EC2 - Ubu							
292	Destroy - cmbu							
291	CentOS 7.3							
290	Ubuntu 14.04	Ubuntu 1						
289	Azure - Ubuntu	Project X						

Page 2 of 5

Request #303 Details

Ansible - Wordpress (PHP/MySQL)
Deploys a Wordpress instance (PHP and MySQL) on CentOS 7.3 using Ansible

Status
Failed
View Details

Submitted By
Jason McCloud

Submitted
9/7/18, 3:35 PM

Last Updated
9/7/18, 3:42 PM

Business Group
cmbu-auto-bg

Deployment: Ansible - Wordpress (PHP/MySQL)

Events for Ansible - Wordpress (PHP/MySQL)-315(

Name	Status	Details	Waiting for	Component ID	Component Type	Request Type	Start Time	Completion Time
centos...	Failed	java.lang.RuntimeException: Show Placement Det...		centos73	vSphere (...	Allocate	9/7/2018, ...	9/7/2018, ...
centos...	Aborted	Marking as ABORTED. A pre	centos73	centos73	vSphere (...	Provision		9/7/2018, ...
Ansible...	Aborted	Marking as ABORTED. A pre	centos73, ...	AnsibleWeb	Ansible	Provision		9/7/2018, ...

Page 1 of 1

Displaying 1 - 3 of 3

How the Resume Action Works

You can use Resume on failed deployments to restart the provisioning process from the point of failure and under specific circumstances. When enabled, the Resume action is available for failed provisioning requests or applicable actions.

To use resume on provisioning requests, you must add the _debug_deployment = true custom property to the blueprint. By default, failed deployments are rolled back and cleaned up so that the resources are reclaimed. The _debug_deployment = true property retains the deployment at the point of failure and, where supported and based on how it works, allows for a Resume action. If you are only using resume on the supported actions, you do not need to enable the _debug_deployment property .

For more about _debug_deployment, see Custom Properties Underscore (_) .

To use resume on a provisioning request or on the available actions, you entitle users to the Resume action. See Entitle Users to Services, Catalog Items, and Actions.

You can entitle users to the Resume action for these provisioning activities.

- Provisioning requests
- Resume action
- Scale In action
- Scale Out action
- Destroy action

Resume Action Constraints

When deciding if you can use Resume rather than requesting a new instance of a blueprint, consider the constraints.

- The blueprint is unmodifiable from the time of the request.

  At request time, an unmodifiable version of the blueprint is associated with the catalog request. This static version contains all the specifications, including attributes, custom properties, settings, and so on, as it was when provisioning started. If you have a failure-producing error in your blueprint, fixing the error and then using Resume does not work because it is referring to the version associated with the request. In this scenario, you must provision a new instance.

  Examples

  - Blueprint A requests 5 GB of RAM, but the request fails because you only have reservations for 3 GB. If you update the blueprint to need only 3 GB, and then run Resume, the Resume action fails. When Resume runs, it checks the original request and is still looking for 5 GB. However, if you increase the system reservation for the business group to 5 GB and run Resume, the Resume action succeeds.
  - When you request Blueprint B, which includes a Guest Custom Specification, it fails. Investigation reveals that the Guest Custom Specification was renamed on your vCenter Server instance. If you update the blueprint with the new name and run Resume, it fails. You updated the blueprint, but the original version is used for the Resume action. If the new name is the one you want to use going forward, deploy a new instance of the blueprint rather than using Resume. Otherwise, you must change the name of the Guest Custom Specification on the vCenter Server instance back to the one expected by the original version and run Resume. If you don't want the next provisioning request to fail, don't forget to update the blueprint with the correct Guest Custom Specification.

  Resume works if you can update the target deployment environment to support the blueprint specifications as they existed at request time.

- Retry is only from the point of failure.

  The Resume action retries the component tasks from the point of failure. It does not resubmit the entire provisioning request.

Examples

- Blueprint C creates an application virtual machine and a database virtual machine. The database VM is successfully deployed, but the provisioning fails on the application VM. If you run the Resume action, only the application VM provisioning is retried.

  If a component is marked as Failed, it is treated as if it never ran. If the installation fails during the configuration phase on the database VM, for example, due to a scripting error, but the database is intact, the database still exists when the script runs during a resume action. The installation script, which includes the configuration script, does not run again. Your resume does not succeed. You must fix the script and provision a new instance.
- Another variation to consider is where the allocation of step succeeded, but the provision failed. In this example, when you resume, which retries from the point of the failed provisioning, the resume request is processing stale allocation information and the resume fails.

Working with the Resume Action and Workflow Subscriptions

If a subscription workflow fails, you cannot run a resume action to resume that workflow. The resume action can only be run on failed provisioning events, at which point a new workflow is run.

For example, if you subscribe to the Catalog Request Received event, then both the failed provisioning request and the new Resume request independently satisfy the subscription conditions, but the subscription is not aware of the failed request and the resume request as related activities.

Force Destroy a Deployment After a Failed Destroy Request

You can force destroy a deployment that is in an inconsistent state as the result of a failed destroy request.

When vRealize Automation fails to destroy a deployment resource during a destroy deployment operation, the destroy operation stops immediately without destroying the remaining deployment resources. This failure leaves the deployment in an inconsistent state, using up resources with no obvious way of destroying the deployment. Business group administrators can force destroy deployments that are left in this inconsistent state.

Prerequisites

- Verify that you are logged in to vRealize Automation as a **business group administrator**.
- Before you run the Force Destroy action, review the Destroy action description in Action Menu Commands for Provisioned Resources.

Procedure

1. On the **Deployments** tab, locate the deployment to destroy.
2. Click **Actions** and click **Destroy**.

3 Enter a description for and reason for the request.

4 Select **Force destroy** and click **Submit**.

Results

vRealize Automation attempts to fully destroy the deployment, including all resources in the deployment. If vRealize Automation is unable to destroy a deployment resource, it skips that resource and continues to destroy the remaining resources in the deployment.

What to do next

Verify that all resources in the deployment have been successfully destroyed. Any resources not destroyed during a force destroy operation must be manually destroyed. Also ensure that any provisioned virtual machine objects are destroyed, as vRealize Automation may attempt to reuse their hostnames, IP addresses, and other configuration details during subsequent provisioning operations.

Troubleshooting a Failed Deployment That Includes a vRealize Orchestrator Workflow

If a failed blueprint deployment includes a vRealize Orchestrator workflow, you can use the token ID to troubleshoot problems with the workflow. You use the token ID to locate the logs in vRealize Orchestrator.

Solution

1 Locate the token ID for the failed workflow.

   a In vRealize Automation, click the **Deployments** tab and locate the deployment or action.

   b Click the deployment name.

   The request can be a deployment or an action.

   c Click the **History** tab, and then the **Request Inputs** tab.

   If the blueprint is based on a vRealize Orchestrator workflow, the page title is vRealize Orchestrator Workflow Execution Details.

   d Locate the Token ID and copy it to your clipboard or a text file.

   For example, ff8080815a685352015a6c8d450801ee.

2 Locate the workflow logs in vRealize Orchestrator using the Control Center

   a Enter the base URL for vRealize Automation in a browser search box.

   The VMware vRealize Automation Appliance page appears.

   b Click **vRealize Orchestrator Control Center**.

   c Log in as a user with root privileges.

   d Click **Inspect Workflows**.

e Click **Finished Workflows**.

f Paste the workflow token in the Token ID text box.

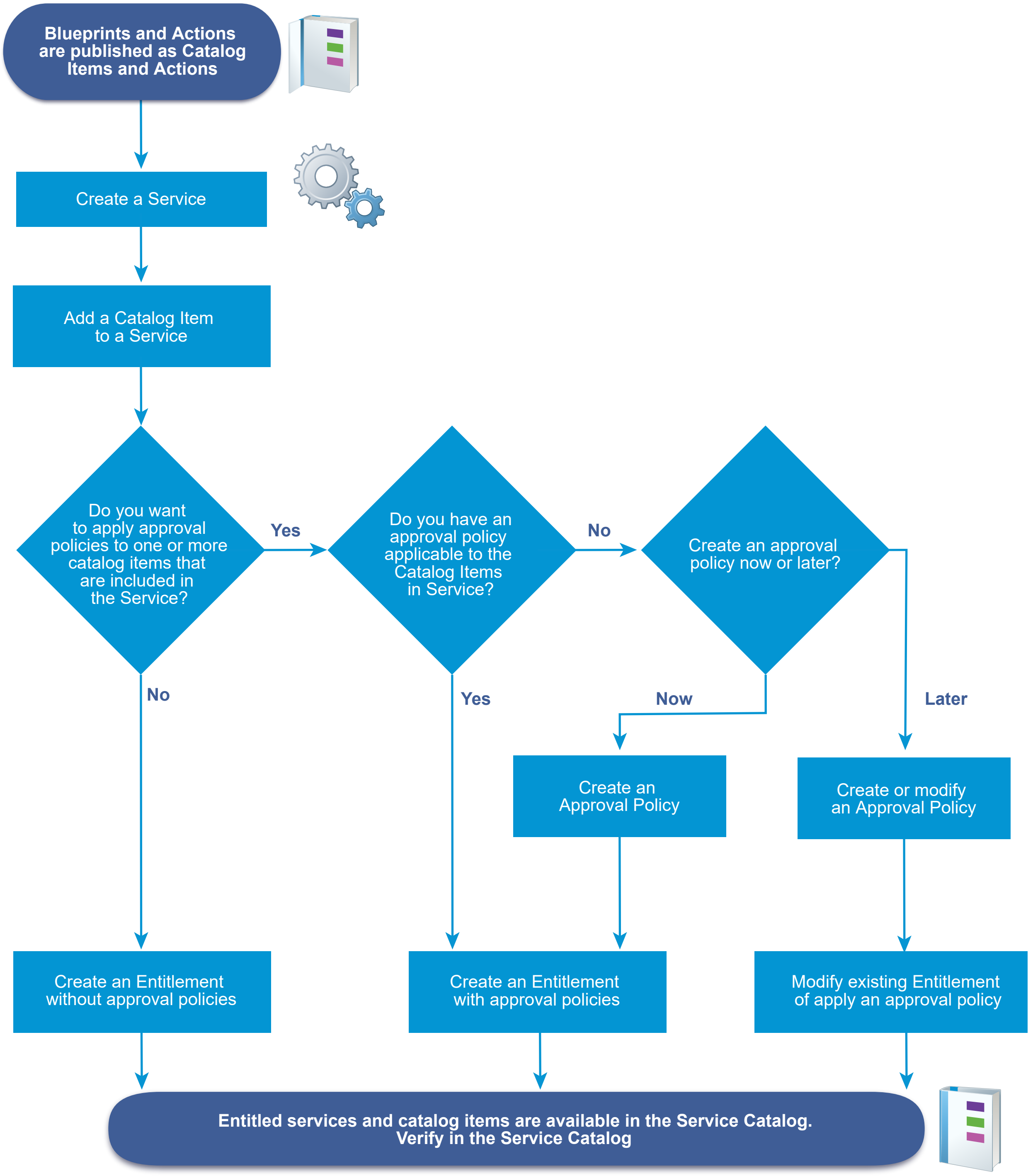
The list displays on the workflow that matches the token ID.

g Click the row and inspect the logs for the cause of the failure.

Managing the Service Catalog

The service catalog is where your customers request machines and other items to provision for their use. You manage user access to the service catalog items based on how you build services, entitle users to one or more items, and apply governance.

The workflow that you follow to add items to the service catalog varies based on whether you create and apply approval policies.


Blueprints and Actions are published as Catalog Items and Actions
Create a Service
Add a Catalog Item to a Service
Do you want to apply approval policies to one or more catalog items that are included in the Service?
Yes
No
Do you have an approval policy applicable to the Catalog Items in Service?
No
Yes
Create an approval policy now or later?
Now
Later
Create an Approval Policy
Create or modify an Approval Policy
Create an Entitlement without approval policies
Create an Entitlement with approval policies
Modify existing Entitlement of apply an approval policy
Entitled services and catalog items are available in the Service Catalog. Verify in the Service Catalog

Checklist for Configuring the Service Catalog

After you create and publish blueprints and actions, you can create a vRealize Automation service, configure catalog items, and assign entitlements and approvals.

The Configuring the Service Catalog Checklist provides a high-level overview of the steps required to configure catalog and provides links to decision points or detailed instructions for each step.

Table 5-70. Configuring the Service Catalog Checklist

Task	Required Role	Details
❑ Add a service.	tenant administrator or catalog administrator	See Add a Service.
❑ Add a catalog item to a service.	tenant administrator or catalog administrator	See Add Catalog Items to a Service.
❑ Configure the catalog item in the service.	tenant administrator or catalog administrator	See Configure a Catalog Item.
❑ Create and apply entitlements to the catalog item.	tenant administrator or business group manager	See Entitle Users to Services, Catalog Items, and Actions.
❑ Create and apply approval policies to the catalog item.	tenant administrator or approval administrator can create approval policies tenant administrator or business group manager can apply approval policies	See Create an Approval Policy.

Creating a Service

A service is a group of catalog items that you want included in the service catalog. You can entitle the service, which entitles business group users to all the associated catalog items, and you can apply an approval policy to the service.

A service operates as a dynamic group of catalog items. If you entitle a service, all the catalog items associated with the service are available in the service catalog to the specified users, and any catalog items that you add or remove from a service affect the service catalog.

As you create the service, you can use it as a service category so that you can assemble service offerings for your service catalog users. For example, a Windows desktop service that includes Windows 7, 8, and 10 operating system catalog items, or a Linux service that includes CentOS and RHEL operating system items.

Add a Service

Add a service to make catalog items available to your service catalog users. All catalog items must be associated with a service so that you can entitle the items to users.

When the service is entitled to users, the catalog items appear together in the service catalog. You can also entitle users to the individual catalog items.

Prerequisites

Log in to vRealize Automation as a **tenant administrator** or **catalog administrator**.

Procedure

1 Select **Administration > Catalog Management > Services**.

2 Click the **New** icon (✚).

3 Enter a name and description.

These values appear in the service catalog for the catalog users.

4 To add a specific icon for the service in the service catalog, click **Browse** and select an image.

The supported image file types are GIF, JPG, and PNG. The displayed image is 40 x 40 pixels. If you do not select a custom image, the default icon appears in the service catalog.

5 Select a status from the **Status** drop-down menu.

Option	Description
Inactive	The service is not available in the service catalog. When a service is in this state, you can associate catalog items with the service, but you cannot entitle the service ot users. If you select **Inactive** for a service that is active and entitled, it is removed from the service catalog until you reactivate it.
Active	(Default) The service and the associated catalog items are available to entitle to users and, if entitled, are available for in the service catalog for those users.
Deleted	Removes the service from vRealize Automation. All associated catalog items are still present, but any items associated with the service in the service catalog are not available to the catalog users.

6 Configuring the service settings.

The following settings provide information to the service catalog users. The settings do not affect service availability.

Option	Description
Hours	Configure the time to coincide with the availability of the support team. The time is based on your local time. The hours of service cannot cross from one day to another. For example, you cannot set the hours of service as 4:00 PM to 4:00 AM. To cross midnight, create two entitlements. One entitlement for 4:00 PM to 12:00 AM, and another for 12:00 AM to 4:00 AM.
Owner	Specify the user or user group who is the primary owner of the service and the associated catalog items.
Support Team	Specify the custom user group or user who is available to support any problems that the service catalog users encounter when they provision items using the service.
Change Window	Select a date and time when you plan to make a change to the service. The date and time specified is informational and does not affect the availability of the service.

7 Click **Add**.

What to do next

Associate catalog items with a service so that you can entitle users to the items. See Add Catalog Items to a Service.

Add Catalog Items to a Service

Add catalog items to services so that you can entitle users to request the items in the service catalog. A catalog item can be associated with only one service.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **catalog administrator**.
- Verify that a service exists. See Add a Service.
- Verify that one or more catalog items are published. See Configure a Catalog Item.

Procedure

1 Select **Administration > Catalog Management > Services**.

2 Select the service to which you are adding catalog items and click **Manage Catalog Items**.

3. Click the **Catalog Items** icon (✚).

   a. Select the catalog items to include in this service.

      The Select Catalog Items dialog box displays only the items that are not already associated with a service.

   b. Click **Add**.

4. Click **Close**.

What to do next

- You can add a custom icon to the catalog item that will appear with the item in the service catalog. See Configure a Catalog Item.
- Entitle users to the services or catalog items so that they can request them in the service catalog. See Creating Entitlements.

Working with Catalog Items and Actions

Catalog items are published blueprints for machines, software components, and other objects. Actions in the catalog management area are published actions that you can run on the provisioned catalog items. You can use the lists to determine what blueprints and actions are published so that you can make them available to service catalog users.

Published Catalog Items

A catalog item is a published blueprint. Published blueprints can also be used in other blueprints. The reuse of blueprints in other blueprints is not displayed in the catalog items list.

The published catalog items can also include items that are only components of blueprints. For example, published software components are listed as catalog items, but they are available only as part of a deployment.

Deployment catalog items must be associated with a service so that you can make them available in the service catalog to entitled users. Only active items appear in the service catalog. You can configure catalog items to a different service, deactivate it if you want to temporarily remove it from the service catalog, and add a custom icon that appears in the catalog.

Published Actions

Actions are changes that you can make to provisioned catalog items. For example, you can reboot a virtual machine.

Actions can include built-in actions or actions created using XaaS. Built-in actions are added when you add a machine or other provided blueprint. XaaS actions must be created and published.

Actions are not associated with services. You must include an action in the entitlement that contains the catalog item on which the action runs. Actions that are entitled to users do not appear in the service catalog. The actions are available for the provisioned item on the service catalog user's **Deployments** tab based whether they are applicable to the item and to the current state of the item.

You can add a custom icon to the action that appears on the **Deployments** tab.

Configure a Catalog Item

A catalog item is a published blueprint that you can entitle to users. You use the catalog items options to change the status or associated service. You can also view the entitlements that include the selected catalog item.

Only catalog items that are associated with a service and entitled to users appear in the service catalog. Catalog items can be associated with only one service.

If you do not want a catalog item to appear in the service catalog without removing it from an entitlement or from the published catalog items list, you can deactivate it. The status of a deactivated catalog item is retired in the grid and inactive in the configuration details. You can activate it later.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **catalog administrator**.
- Verify that you have at least one blueprint published as a catalog item. See Publish a Blueprint.

Procedure

1 Select **Administration > Catalog Management > Catalog Items**.

2 Select the catalog item and click **Configure**.

3 Configure the catalog item settings.

Option	Description
Icon	Browse for an image. The supported image file types are GIF, JPG, and PNG. The displayed image is 40 x 40 pixels. If you do not select a custom image, the default catalog icon appears in the service catalog.
Status	Possible values include **Active**, **Inactive**, and **Staging**. ■ **Active**. The catalog item appears in the service catalog and entitled users can use it to provision resources. The item appears in the catalog item list as published. ■ **Inactive**. The catalog item is not available in the service catalog. The item appears in the catalog item list as retired. ■ **Staging**. The catalog item is not available in the service catalog. Select this menu item if the item was once inactive and you are using staging to indicate that you are considering reactivating it. Appears in the catalog item list as staging.

Option	Description
Quota	Set the number of instances of this catalog item that a user can deploy. If the user exceeds the number, a notification appears on the catalog request and the request is not submitted.
Service	Select a service. All catalog items must be associated with a service if you want it to appear in the service catalog for entitled users. The list includes active and inactive services.

4 To view the entitlements where the catalog item is made available to users, click the **Entitlements** tab.

5 Click **Update**.

What to do next

- To make the catalog item available in the service catalog, you must entitle users to the service associated with the item or to the individual item. See Creating Entitlements.
- To specify the entitlements processing order so that the approval policies for individual users are applied correctly, set the priority order for multiple entitlements for the same business group. See Prioritize Entitlements.

Configure an Action for the Service Catalog

An action is a change or workflow that can run on provisioned items. You can add an icon or view the entitlements that include the selected action.

An action is either a built-in action for a provisioned machine, network, and other blueprint components, or it is a published XaaS action.

For the icon, the supported image file types are GIF, JPG, and PNG. The displayed image is 40 x 40 pixels. If you do not select a custom image, the default action icon appears on the **Deployments** tab.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **catalog administrator**.
- Verify that you have at least one published action. See Publish a Blueprint and Publish a Resource Action.

Procedure

1 Select **Administration > Catalog Management > Actions**.

2 Select the shared action and click **View Details** or, for XaaS actions, **Configure**.

3 Browse for an image.

4 To view the entitlements where the action is made available to users, click the **Entitlements** tab.

5 Click **Finish**.

What to do next

Entitle Users to Services, Catalog Items, and Actions.

Creating Entitlements

Entitlements control what items and actions are available in the service catalog for the members of the selected business group. An entitlement must be active for the items to appear in the service catalog. If you have items that require governance, you can use entitlements to apply approval policies to different items.

To configure the entitlement, the catalog items must be included in a service. Entitlements can include multiple services, catalog items from services that are included in other entitlements, and actions that you can run on the deployed catalog items.

Understanding Entitlement Option Interactions

How you configure an entitlement determines what appears in the service catalog. The interaction of services, catalog items and components, action, and approval policies affects what the service catalog user can request and how approval policies are applied.

You must consider the interactions of services, catalog items, actions, and approvals when you create an entitlement.

- Services in Entitlements

  An entitled service operates as a dynamic group of catalog items. If a catalog item is added to a service after it is entitled, the new catalog item is available to the specified users without any additional configuration.

- Catalog Items and Components in Entitlements

  Entitled catalog items are blueprints that you can request in the service catalog. Entitled components are part of the blueprints, but you cannot specifically request them in the service catalog.

- Actions in Entitlements

  Actions run on deployed catalog items. Provisioned catalog items, and the actions you are entitled to run on them, appear in your Deployments tab. To run actions on a deployed item, the action must be included in the same entitlement as the catalog item that provisioned the item from the service catalog.

- Approval Policies in Entitlements

  Approval policies are applied in entitlements so that you can manage resources in your environment.

Services in Entitlements

An entitled service operates as a dynamic group of catalog items. If a catalog item is added to a service after it is entitled, the new catalog item is available to the specified users without any additional configuration.

If you apply an approval policy to a service, all the items, when requested, are subject to the same approval policy.

Catalog Items and Components in Entitlements

Entitled catalog items are blueprints that you can request in the service catalog. Entitled components are part of the blueprints, but you cannot specifically request them in the service catalog.

Entitled catalog items and components can include any of the following items:

Catalog Items

- Items from any service that you want to provide to entitled users, even services not included in the current entitlement.

  For example, as a catalog administrator you associated several different versions of the Red Hat Enterprise Linux with a Red Hat service and entitle the service to the quality engineers for product A. Then you receive a request to create service catalog items that includes only the latest version of Linux-based operating systems for the training team. You create an entitlement for the training team that includes the latest versions of the other operating systems in a service. You already have the latest version of RHEL associated with another service, so you add RHEL as a catalog item rather than add the entire Red Hat service.

- Items that are included in a service that is included in the current entitlement, but you want to apply an approval policy to the individual catalog item that differs from the policy you applied to the service.

  For example, as a business group manager, you entitle your development team to a service that includes three virtual machine catalog items. You apply an approval policy that requires the approval of the virtual infrastructure administrator for machines with more than four CPUs. One of the virtual machines is used for performance testing, so you add it as a catalog item and apply less restrictive approval policy for the same group of users.

Components

- Components are not available by name in the service catalog because they are a part of a catalog item. You entitle them individually so that you can apply a specific approval policy that differs from the catalog item in which it is included.

  For example, an item includes a machine and software. The machine is available as a provisionable item and has an approval policy that requires site manager approval. The software is not available as a standalone, provisionable item, only as part of a machine request, but the approval policy for the software requires approval from your organization's software licensing administrator. When the machine is requested in the services catalog, it must be approved by the site administrator and the software licensing administrator before it is provisioned. After it is provisioned, the machine, with the software entry, appears in the requestor's Deployments tab as part of the machine.

Actions in Entitlements

Actions run on deployed catalog items. Provisioned catalog items, and the actions you are entitled to run on them, appear in your Deployments tab. To run actions on a deployed item, the action must be included in the same entitlement as the catalog item that provisioned the item from the service catalog.

For example, entitlement 1 includes a vSphere virtual machine and a create snapshot action, and entitlement 2 includes only a vSphere virtual machine. When you deploy a vSphere machine from entitlement 1, the create snapshot action is available. When you deploy a vSphere machine from entitlement 2, there is no action. To make the action available to entitlement 2 users, add the create snapshot action to entitlement 2.

If you select an action that is not applicable to any of the catalog items in the entitlement, it will not appear as an action on the Deployments tab. For example, your entitlement includes a vSphere machine and you entitle a destroy action for a cloud machine. The destroy action is not available to run on the provisioned machine.

You can apply an approval policy to an action that is different from the policy applied to the catalog item in the entitlement.

If the service catalog user is the member of multiple business groups, and one group is only entitled to power on and power off and the other is only entitled to destroy, that user will have all three actions available to them for the applicable provisioned machine.

Best Practices When Entitling Users to Actions

Blueprints are complex and entitling actions to run on provisioned blueprints can result in unexpected behavior. Use the following best practices when entitling service catalog users to run actions on their provisioned items.

- When you entitle users to the Destroy Machine action, entitle them to Destroy Deployment. A provisioned blueprint is a deployment.

  A deployment can contain a machine. If the service catalog user is entitled to run the Destroy Machine action and is not entitled to run the Destroy Deployment, when the user runs the Destroy Machine action on the last or only machine in a deployment, a message appears indicating that they do not have permission to run the action. Entitling both actions ensures that the deployment is removed from your environment. To manage governance on the Destroy Deployment action, you can create a pre approval policy and apply it to the action. This policy will allow the designated approver to validate the Destroy Deployment request before it runs.
- When you entitle service catalog users to the Change Lease, Change Owner, Expire, Reconfigure and other actions that can apply to machines and to deployments, entitle them to both actions.

Approval Policies in Entitlements

Approval policies are applied in entitlements so that you can manage resources in your environment.

To apply an approval policy when you create the entitlement, the policy must already exist. If it does not, you can still create the entitlement and leave it in a draft or inactive state until you create the approval policies needed for the catalog items and actions in this entitlement, and then apply the policies later.

You are not required to apply an approval policy to any of the items or actions. If no approval policy is applied, the items and actions are deployed when requested without triggering an approval request.

Entitle Users to Services, Catalog Items, and Actions

When you add a service, catalog item, or action to an entitlement, you allow the users identified in the entitlement to request the provisionable items in the service catalog. Actions are associated with items and appear on the **Deployments** tab for the requesting user.

There are several user roles with permission to create entitlements for business groups.

- Tenant administrators can create entitlements for any business group in their tenant.
- Business group managers can create entitlements for the groups that they manage.
- Catalog administrators can create entitlements for any business group in their tenant.

When you create an entitlement, you must select a business group and the members in the business group for the entitlement.

To understand how to create an entitlement so that you can use the interactions of services, catalog items, and actions with approvals, see Creating Entitlements.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **catalog administrator**.
- Verify that the catalog items to which you are entitling users are associated with a service. See Add Catalog Items to a Service.
- Verify that the business group for which you are defining the entitlement exists and that the member users and user groups are defined. See Create a Business Group.
- Verify that the approval policies exist if you plan to add approvals when you create this entitlement. See Create an Approval Policy. If you want to entitle users to the items in the service catalog without approvals, you can modify the entitlement later to add approvals.

Procedure

1 Select **Administration** > **Catalog Management** > **Entitlements**.

2 Click the **New** icon (✚).

3 Configure the **Details** options.

Details determine how the entitlement appears in the entitlement list and which users have access to the items in the service catalog.

Option	Description
Name and Description	Information about the entitlement that appears in the entitlements list.
Expiration Date	Set the date and time if you want the entitlement to become inactive on a particular date.
Status	Possible values include Active, Inactive, and Deleted. ■ Active. Items are available in the service catalog. This option is available when you add or edit entitlements. ■ Inactive. Items are not available in the service catalog. The entitlement was deactivated by the expiration date or by a user. ■ Deleted. Deletes the entitlement.
Business Group	Select a business group. You can create entitlements for only one business group and entitled users must be members of the business group. If you want to make an entitlement available to all users, you must have an All Users business group, or you must create entitlements for each business group. If you are logged in as a business group manager, you can create entitlements only for your business group.
Users and Groups	Select **All Users and Groups** to entitle all the members of the business group to the catalog items and actions, or you can entitle individual users or groups. To activate an entitlement, you must select at least one business group user or group.

4 Click **Next**.

5 Click an **New** icon (✚) to entitle users to services, catalog items, or actions with this entitlement.

You can create an entitlement with various combinations of the services, items, and actions.

Option	Description
Entitled Services	Add a service when you want to allow entitled users access to all the published catalog items associated with the service. An entitled service is a dynamic entitlement. If an item is added to the service later, it is added to the service catalog for the entitled users. Entitlements can include both services and individual catalog items.
Entitled Catalog Items and Components	Add individual items that are available to the entitled users. Entitlements can include both services and individual catalog items. To apply a different approval policy to an item that is included in the service, add it as a catalog item. The approval policy on an item takes precedence over the approval policy on the service to which it belongs when they are in the same entitlement. If they are in different entitlements, the order is based on the set priority. Catalog items must be associated with a service to be available in the service catalog. The catalog item can be associated with any service, not only a service in the current entitlement. Components are a part of a catalog item but are not available by name in the service catalog. For example, MySQL software is a component of a CentOS virtual machine catalog item. Components are entitled with the catalog item. If you want to apply an approval policy that is specific to software, you entitle the item individually. Otherwise, you do not need to entitle a component for it to be deployed with the parent item.
Entitled Actions	Add actions when you want to allow users to run the actions for a provisioned item. Actions that you want to run on the items provisioned from this entitlement must be included in the same entitlement. Entitled actions do not appear in the service catalog. They appear on the Deployments tab for a provisioned item.
Actions only apply to items defined in this entitlement	Determines if the entitled actions are entitled for all applicable service catalog items or only the items in this entitlement. If selected, the actions are entitled to the business group members for the applicable items in this entitlement. This method of entitling the actions allows you to specify the actions for the specific items. If this option is not selected, the actions are entitled to the users specified in the entitlement for all applicable catalog items, whether or not the items are included in this entitlement. Any applied approval policies on these actions are also active.

6 Use the drop-down menus in each section to filter the available items.

7 Select the check boxes to include items to the entitlement.

8 To add an approval policy to the selected service, item, or action, select an approval policy from the **Apply this Policy to selected Items** drop-down menu.

   If you apply an approval policy to a service, all the items in the service have the same approval policy. To apply a different policy to an item, add it as a catalog item an apply the appropriate policy.

9 Click **OK**.

   The service, item, or action is added to the entitlement.

10 Click **Finish** to save the entitlement.

Results

If entitlement status is active, the service and items are added to the service catalog.

What to do next

Verify that the entitled services and catalog items appear in the service catalog for the entitled users and that the requested items provision the target objects as expected. You can request the item on behalf of the selected users.

Prioritize Entitlements

If multiple entitlements exist for the same business group, you can prioritize the entitlements so that when a service catalog user makes a request, the entitlement and associated approval policy are processed in the specified order.

If you configure an approval policy for a user group, and you want a group member to have a unique policy for one or more of the services, catalog items, or actions, prioritize the member entitlement before the group entitlement. When the member requests an item in the service catalog, the approval policy that is applied is based on the priority order of the entitlements for the business group. The first time that the member's name is found, either as part of a custom user group or as an individual user, that is the applied approval policy.

For example, you create two entitlements for the same catalog item so that you can apply one approval policy for the accounting user group and a different approval policy for Chris, a member of that group.

Table 5-71. Example Entitlements

Entitlement 1	Entitlement 2
Business Group: Finance Users and Groups: Accounting group Catalog Item 1: Policy A	Business Group: Finance Users and Groups: Chris Catalog Item 1: Policy C

Chris requests Catalog Item 1 in the service catalog. Depending on the priority order of the entitlements for the Finance business group, a different policy is applied to Chris's request.

Table 5-72. Example Results

Configuration and Result	Priority Order	Priority Order
Priority Order	1: Entitlement 1 2: Entitlement 2	1: Entitlement 2 2: Entitlement 1
Applied Policy	Policy A is applied. Chris is a member of the Accounting user group. The search for Chris as an entitled user stops at Entitlement 1 and the approval policy is applied.	Policy C is applied. The search for Chris as an entitled user stops at Entitlement 2 and the approval policy is applied.

Prerequisites

Log in to vRealize Automation as a **tenant administrator** or **catalog administrator**.

Procedure

1 Select **Administration > Catalog Management > Entitlements**.

2 Click the **Prioritize** icon (⇅).

3 Select a business group from the **Business Group** drop-down list.

4 Drag an entitlement to a new location in the list to change its priority.

5 Select an update method.

Option	Description
Update	Saves your changes.
Update & Close	Saves your changes and closes the **Prioritize Elements** window.

Working with Approval Policies

Approval policies are governance that you add to service catalog requests so that you can manage resources in your environment. Each policy is a defined set of conditions that can be applied to services, catalog items, and actions when you entitle users to those items.

Approval Policy Process

First, a tenant administrator or approval administrator creates the approval policies where provisioning governance is needed.

Approval policies are created for approval policy types or specific items. If the policy is based on a policy type, you can apply it to matching catalog item types. For example, if a policy is based on a software policy type, then you can define it for and apply it to any software items in the entitlements. If the policy is for a specific item, you should apply it only to that item. For example, if the item is a specific software item, then you should apply it only to that specific database software item in the entitlement.

Policies can include pre-approval and post-approval requirements. For pre approval, the request must be approved before the requested item is provisioned. Post approval policies require that the approver accept the request before the provisioned item is made available to the requesting user.

The pre and post approval configurations are composed of one or more levels that determine when the approval policy is triggered and who or how the request is approved. You can include multiple levels. For example, an approval policy can have one level for manager approval, followed by a level for finance approval.

Next, a tenant administrator or business group manager applies the approval policies to the services, catalog items, and actions as appropriate.

Finally, when a service catalog user requests an item to which an approval policy is applied, the approvers approve or reject the request on their **Inbox** tab. The requesting user can track the approval status for a specific request on their **Deployments** tab.

Examples of Approval Policies Based on the Virtual Machine Policy Type

You can create an approval policy that you can apply to the same catalog item type, but it produces different results when an item is requested in the service catalog. Depending on how the approval policy is defined and applied, the effect on the service catalog user and the approver varies.

The following table includes examples of different approval policies that are all based on the same approval policy type. These examples illustrate some of the ways that you can configure approval policies to accomplish different types of governance.

Table 5-73. Examples of Approval Policies and Results

Governance Goals	Selected Policy Type	Pre or Post Approval	When is Approval Required	Who are the Approvers	How is the Policy Applied in the Entitlement	Results When the Item is Requested in the Service Catalog
The business group manager must approve any virtual machine requests. The approval policy must be applicable to multiple business groups in multiple entitlements.	Service Catalog - Catalog Item Request - Virtual Machine	Add to Pre Approval tab	Select Always required	Select **Determine approvers from the request**. Select condition **Business Group > Managers > Users > manager**. Select **Anyone can approve**.	Entitlements are based on business groups. This approval can be used in any entitlement where manager approval is required for the virtual machine.	When the service catalog user requests a virtual machine to which this approval was applied, the business group manager must approve the request before the machine is provisioned.
The virtual infrastructure administrator must verify the correct provisioning of the virtual machine and approve the request before the virtual machine is released to the requesting user.	Service Catalog - Catalog Item Request - Virtual Machine	Add to Post Approval tab	Select Always required	Select **Specific Users and Groups**. Select your virtual infrastructure administrators custom users group. Select **Anyone can approve**.	This approval can be used in any entitlement where you want the virtual infrastructure administrator to check the virtual machine on the vCenter Server after it is provisioned.	When the service catalog user requests a virtual machine to which this approval was applied, the virtual machine is provisioned. If each member of the VI admin group approves the request, the machine is released to the user.

Table 5-73. Examples of Approval Policies and Results (continued)

Governance Goals	Selected Policy Type	Pre or Post Approval	When is Approval Required	Who are the Approvers	How is the Policy Applied in the Entitlement	Results When the Item is Requested in the Service Catalog
To manage virtual infrastructure resources and to control prices, you add two pre-approval levels because one approval is for machine resources and the other is for price of machine per day.	Service Catalog - Catalog Item Request - Virtual Machine	Add To Pre Approval tab	Level 1 Select **Required based on conditions**. Configure the conditions where CPUs > 6 or Memory > 8 or Storage > 100 GB.	Select **Determine approvers from the request**. Select condition Requested by > manager. Select . Click **System Properties** and select **CPUs**. **Memory**, and **Storage** so that the approver can change the value to an acceptable level.	This approval policy can be used in an entitlement where you want the requesting user's manager and a member of the finance department to approve the request.	When the service catalog user requests a virtual machine, the request is evaluated to determine whether the requested CPU, memory, or storage amounts are over the amounts specified in level 1. If they are not, then the level 2 condition is evaluated. If the requests exceeds at least one of the level 1 conditions, then the manager must approve the request. The manager has the option to decrease the requested configuration amounts and approve or the manager can reject the request.

Table 5-73. Examples of Approval Policies and Results (continued)

Governance Goals	Selected Policy Type	Pre or Post Approval	When is Approval Required	Who are the Approvers	How is the Policy Applied in the Entitlement	Results When the Item is Requested in the Service Catalog
			Level 2 Select **Required based on conditions**. Configure the condition Price > 15.00 per day.	Select **Specific Users and Groups**. Select the finance custom users group. Select **Anyone can approve**.		
For parameterized blueprint catalog items, a cloud administrator must approve deployment requests in which a vSphere machine component profile of `size` is set to `large`.	Service Catalog - Catalog Item Request - Virtual Machine	Add To Pre Approval tab	Level 1 Select **Required based on conditions**. Level 2 Select **Single Condition**. Select **Component profile > vSphere Machine Size**. Configure the condition size = large.	Select **Specific Users and Groups**. Select users and groups who are allowed to approve the request. Select **Anyone can approve**.	This approval policy can be used in an entitlement where you want a cloud administrator to approve the provisioning request.	When the service catalog user requests a virtual machine to which this approval was applied, a cloud administrator must approve the request before the machine is provisioned.

Example of Actions with Approval Policies Applied in a Composite Deployment

When you apply approval policies to actions that can run on various components in a composite blueprint, the approval process varies depending on how the entitlement is configured and how the approval policies are applied.

This example uses specific details to build the blueprint and then apply approval policies to actions that you can run from the service catalog on the provisioned blueprint in different entitlements. The blueprint is a composite blueprint that includes another blueprint. The actions used are to destroy the provisioned items, destroy a deployment for the blueprints and destroy a virtual machine for the machine. The resulting behavior includes what is destroyed and when the applied approval policies trigger approval requests.

Example Blueprint

In this example, you configure a blueprint that includes a nested blueprint with a virtual machine.

- Blueprint 1 - Continuous Integration Blueprint
  - Blueprint 2 - Pre-Production Blueprint
    - Virtual Machine 1 - TestAsAService vSphere VM

Approval Policies for Destroy Actions

You configure the two approval policies to destroy provisioned items. A Destroy - Deployment action can run on Blueprint 1 or Blueprint 2 in this example. A Destroy - Virtual machine action can run on Virtual Machine 1. You create the approval policies so that you can apply them to the actions in the entitlement.

Approval Policy Name	Approval Policy Type
Approval Policy A	Service Catalog - Resource Action Request - Destroy - Deployment
Approval Policy B	Service Catalog - Resource Action Request - Destroy - Virtual Machine

Entitlements and Approval Policies Applied to Actions

You configure three entitlements. Each entitlement includes the composite blueprint. In each entitlement, you add the destroy actions and apply the approval policies.

Entitlement Name	Entitled Action on Provisioned Machine	Applied Approval Policy
Entitlement 1	Destroy - Deployment	Approval Policy A
Entitlement 2	Destroy - Virtual Machine	Approval Policy B
Entitlement 3	Destroy - Deployment Destroy - Virtual Machine	Approval Policy A Approval Policy B

User Actions in the Service Catalog

When the service catalog user runs the action, blueprints or machines are destroyed depending on which item your user ran the action.

User Action in the Service Catalog	Selected Action	Destroyed Blueprints or Machines
Action 1	Destroy - Deployment action runs on Blueprint 1 - Continuous Integration Blueprint	Blueprint 1, Blueprint 2, and Virtual Machine 1
Action 2	Destroy - Deployment action runs on the nested Blueprint 2 - Pre-production Blueprint	Blueprint 2 and Virtual Machine 1
Action 3	Destroy - Virtual Machine action runs on the machine that is inside a deployment, Virtual Machine 1 - TestAsAService vSphere VM	Virtual Machine 1

Approval Policies Applied to Actions in the Entitlements

You apply the approval policies, the approvers receive an approval request depending on the blueprint or machine on which your service catalog user ran the action.

Entitlement Name	Approval Policy on Actions	User Action	Approval Request Triggered	If Approved, Destroyed Blueprints or Machines
Entitlement 1 - Destroy Deployment Approval Policy	Policy A (Destroy Deployment Approval Policy) on Destroy - Deployment action only	Action 1 (Run Destroy - Deployment action on Blueprint 1)	Approval requests are triggered for Blueprint 1 only	Blueprint 1, Blueprint 2, and Virtual Machine 1
		Action 2 (Run Destroy - Deployment action on the Blueprint 2)	Approval requests are triggered for Blueprint 2 only	Blueprint 2 and Virtual Machine 1
		Action 3 (Destroy - Virtual Machine action runs on Virtual Machine 1)	No Approval requests are triggered	Virtual Machine 1
Entitlement 2	Policy B (Destroy - Virtual Machine Policy) on Destroy - Virtual Machine action only	Action 1 (Run Destroy - Deployment action on Blueprint 1)	No Approval requests are triggered	Blueprint 1, Blueprint 2, and Virtual Machine 1
		Action 2 (Run Destroy - Deployment action on the Blueprint 2)	No Approval requests are triggered	Blueprint 2 and Virtual Machine 1
		Action 3 (Destroy - Virtual Machine action runs on Virtual Machine 1)	Approval requests are triggered for Virtual Machine 1 only	Virtual Machine 1
Entitlement 3	Policy A (Destroy Deployment Approval Policy) on Destroy - Deployment action and Policy B (Destroy - Virtual Machine Policy) on Destroy - Virtual Machine action	Action 1 (Run Destroy - Deployment action on Blueprint 1)	Approval requests are triggered for Blueprint 1 only	Blueprint 1, Blueprint 2, and Virtual Machine 1
		Action 2 (Run Destroy - Deployment action on the Blueprint 2)	Approval requests are triggered for Blueprint 2 only	Blueprint 2 and Virtual Machine 1
		Action 3 (Destroy - Virtual Machine action runs on Virtual Machine 1)	Approval requests are triggered for Virtual Machine 1 only	Virtual Machine 1

Example of an Approval Policy in Multiple Entitlements

If you apply an approval policy to an item that is used in multiple entitlements that are entitled to same users in a business group, the approval policy is triggered on the item even in the service where the approval policy is not explicitly applied in the entitlement.

For example, you create the following blueprints, services, approval policies, and entitlements.

Blueprints

- RHEL vSphere virtual machine
- QE Testing includes RHEL vSphere virtual machine
- QE Training includes RHEL vSphere virtual machine

Services

- The QE Testing blueprint is associated with the Testing service
- The QE Training blueprint is associated with the Training service

Entitlements

- Entitlement 1
- Entitlement 2

Table 5-74. Entitlement Configurations

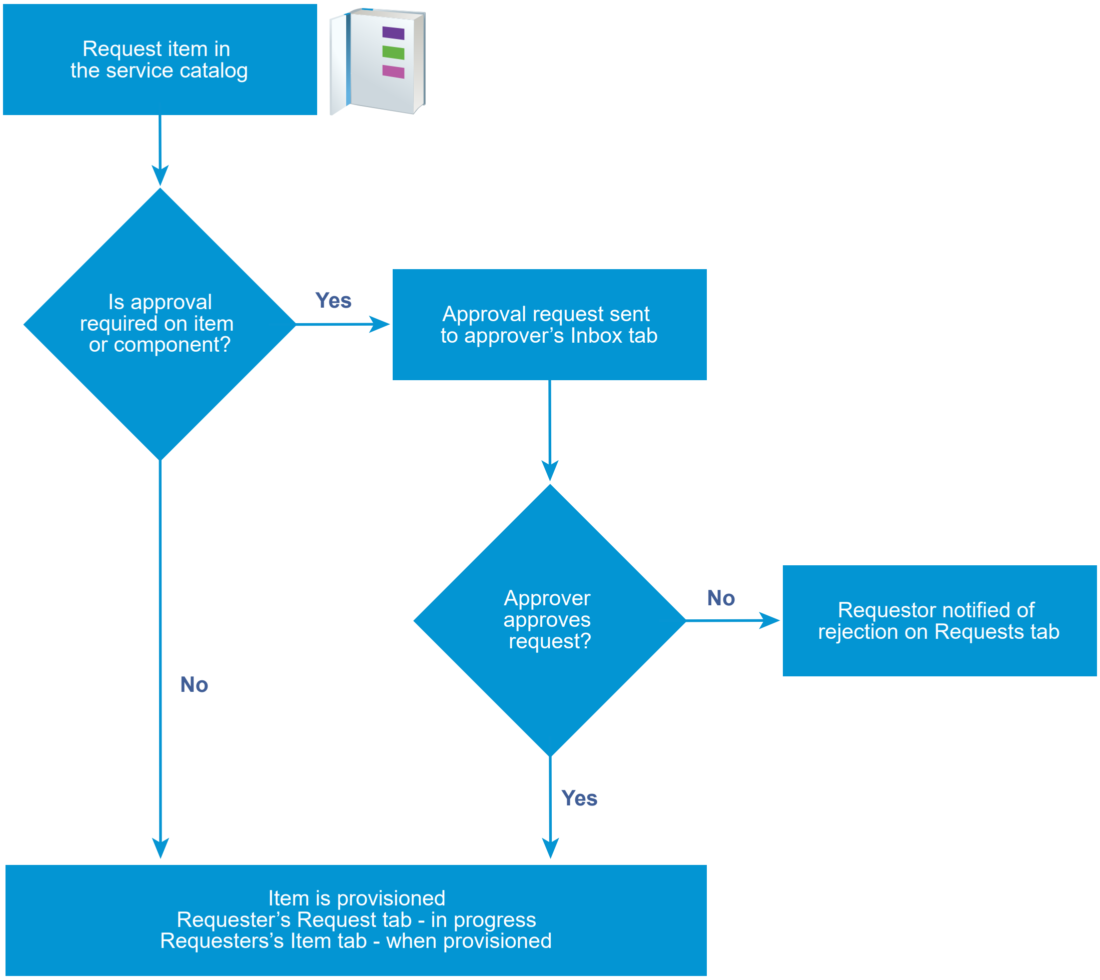
Entitlement Name	Business Group	Entitled Service	Entitled Item
Entitlement 1	QE	Testing	Catalog Item Request - Virtual Machine applied to Virtual Machine Component
Entitlement 2	QE	Training	

Results

When the user selects QE Training in the service catalog, the approval policy is triggered for RHEL vSphere virtual machine because it is a blueprint based on virtual machine component that is used in the QE Training blueprint.

Processing Approval Policies in the Service Catalog

When a user requests an item in the service catalog that has an approval policy applied, the request is processed by the approver and the requesting user similar to the following workflow


Request item in the service catalog
Is approval required on item or component?
Yes
Approval request sent to approver's Inbox tab
No
Approver approves request?
No
Requestor notified of rejection on Requests tab
Yes
Item is provisioned
Requester's Request tab - in progress
Requesters's Item tab - when provisioned

Create an Approval Policy

Tenant administrators and approval administrators can define approval policies and use them in entitlements. You can configure the approval policies with multiple levels for pre-approval and post-approval events.

If you modify a setting in a software component blueprint and an approval policy uses that setting to trigger an approval request, the approval policy might not work as expected. If you must modify a setting in a component, verify that your changes do not affect one or more approval policies.

Prerequisites

Log in to vRealize Automation as a **tenant administrator** or **approval administrator**.

Procedure

1 Specify Approval Policy Information

When you create an approval policy, define the approval policy type, name, description, and status.

2. Create an Approval Level

   When you create an approval policy, you can add pre-approval and post-approval levels.

3. Configure the Approval Form to Include System and Custom Properties

   You can add system and custom properties that appear on an approval form. You add these properties so that the approvers can change the values of system properties for machine resource settings such as CPU or memory, and custom properties before they complete an approval request.

4. Approval Policy Settings

   When you create an approval policy, you configure various options that determine when an item requested by a service catalog users must be approved. The approval can be required before the request begins provisioning or after the item is provisioned but before it is released to the requesting user.

Specify Approval Policy Information

When you create an approval policy, define the approval policy type, name, description, and status.

Prerequisites

Log in to vRealize Automation as a **tenant administrator** or **approval administrator**.

Procedure

1. Select **Administration > Approval Policies**.
2. Click the **New** icon (✚).

3 Select a policy type or software component.

Option	Description
Select an approval policy type	Create an approval policy based on the policy request type. Select this option to define an approval policy that is applicable to all catalog items of that type. The request type can be a generic request, a catalog item request, or a resource action request. The available condition configuration options vary depending on the type. The more specific the type the more specific the configuration fields. For example, Service Catalog - Catalog Item Request provides only the fields that are common to all catalog item requests, but a Service Catalog - Catalog Item Request - Virtual Machine also includes the common options and options specific to virtual machines. The request type limits the catalog items or actions to which you can apply the approval policy.
Select an item	Create an approval policy based on a specific item. Select this option to define an approval policy that is applicable to specific items that are not available as individual items in the service catalog, only as part of a machine or other deployment. For example, software components. The available condition configuration fields are specific to the item and can be more detailed than the criteria offered for a policy type item.
List	Lists the available policy type or catalog items. Search or sort the columns to locate a specific item or type.

4 Click **OK**.

5 Enter a name and, optionally, a description.

6 Select the state of the policy from the **Status** drop-down menu.

Option	Description
Draft	Saves the approval policy in an editable state.
Active	Saves the approval policy in a read-only state that you can use in an entitlement.
Inactive	Saves the approval policy in a read-only state that you cannot use in an entitlement until you activate the policy.

What to do next

Create the pre-approval and post-approval levels.

Create an Approval Level

When you create an approval policy, you can add pre-approval and post-approval levels.

You can create multiple approval levels for an approval policy. When a service catalog user requests an item to which an approval policy with multiple levels is applied, each the first level must be accepted before the approval request is sent to the next approver. See Working with Approval Policies.

If you configure an approval policy that is triggered by a lease duration request, you must select Always Required as the approval requirement.

Prerequisites

Specify Approval Policy Information.

Procedure

1 On the **Pre Approval** or **Post Approval** tab, click the **New** icon (✚).

2 Enter a name and, optionally, a description.

3 Select an approval requirement.

Option	Description
Always Required	The approval policy is triggered for every request.
Required based on conditions	The approval policy is based on one or more condition clauses. If you select this option, you must create the conditions. When this approval policy is applied to eligible services, catalog items, or actions in an entitlement, then the conditions are evaluated. If the conditions are true, then the request must be approved by the specified approver method before it is provisioned. If the conditions are false, then the request is provisioned without requiring an approval. For example, any requests for a virtual machine with 4 or more CPUs must be approved by the virtual infrastructure administrator. The availability of the fields on which to base the conditions is determined by the selected approval policy type or catalog item. When you enter a value for a condition, the values are case-sensitive. To configure more than one condition clause, select the Boolean operation for the clauses.

4 Select the approvers.

Option	Action
Specific Users and Groups	Sends the approval request to the selected users.
Determine approvers from the request	Sends the approval request to the users based on the defined condition. **Note** Ensure that all users that will be dynamically determined by the request and requester exist in vRealize Automation, that they are synced in the Active Directory, and can be browsed from **Administration > Users & Groups > Directory Users and Groups**. If a user is not synced in the Directories Management identity provider and this user is referenced in any way during the catalog request, the request will fail with a Requested Item Approval runtime error.
Use event subscription	Processes the approval request based on defined event subscriptions. The workflow subscription must be defined in **Adminstration > Events > Subcriptions**. The applicable workflow subscriptions are pre-approval and post-approval.

5. Indicate who must approve the request or action.

Option	Description
Anyone can approve	Only one of the approvers must approve before the request is processed. When the item is requested in the service catalog, requests for approval are sent to all approvers. If one approver approves the request, the request is approved and the request for approval is removed from the other approvers' inboxes.
All must approve	All of the specified approvers must approve before the request is processed.

6. Add properties to an approval form or save the level.
   - To add properties to the approval form, click **System Properties** or **Custom Properties**.
   - To save the level, click **OK**.

What to do next

To add properties to the approval form, see Configure the Approval Form to Include System and Custom Properties.

Configure the Approval Form to Include System and Custom Properties

You can add system and custom properties that appear on an approval form. You add these properties so that the approvers can change the values of system properties for machine resource settings such as CPU or memory, and custom properties before they complete an approval request.

The available system properties depend on the approval policy type and how the blueprint is configured. For some properties, the configured field in the blueprint must include a minimum and maximum value before the property appears in the system properties list.

Custom properties can be added when you add the approval level. If a custom property is configured and included in a blueprint, the custom properties you add to the approval form overwrite any other instances of that custom property for example, in blueprints, property groups, or endpoints.

The approver can modify selected or configured properties in the approval form.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **approval administrator**.
- Create an Approval Level.

Procedure

1. On the **Pre Approval** or **Post Approval** tab, click the **New** icon (✚).
2. Click the **System Properties** tab.

3 Select the check box for each system property that you want the approver to configure during the approval process.

4 Configure the custom properties.

   Add one or more custom properties that you want the approver to configure during the approval process.

   a Click the **Custom Properties** tab.

   b Click the **New** icon (✚).

   c Enter the custom property values.

Option	Description
Name	Enter the property name.
Label	Enter the label that is presented to the approver in the approval form.
Description	Enter the extended information for the approver. This information appears as the field tooltip in the form.

   d Click **Save**.

   e To delete multiple custom properties, select the rows and click **Delete**.

5 Click **OK**.

What to do next

- Add additional pre-approval or post-approval levels.
- Save the approval policy. The policy must be active to apply to services, items, or actions in the **Entitlements**.

Approval Policy Settings

When you create an approval policy, you configure various options that determine when an item requested by a service catalog users must be approved. The approval can be required before the request begins provisioning or after the item is provisioned but before it is released to the requesting user.

Select **Administration > Approval Polices**. Click **New**.

- Approval Policy Type Settings

  The approval policy type determines how the approval policy is configured and to what items or actions you can apply it in the entitlement. When you add approval levels, the policy type or item affects which fields are available to create conditions for the approval levels.

- Add Approval Policy Settings

  You configure the basic information about the approval policy, including the state to the policy, so that you can manage the policy.

- Add Level Information to Approval Policy Settings

  An approval level includes the conditions that trigger an approval process when the service catalog user requests the item, and any system properties and customer properties that you want to include. When triggered, the approval requests are sent to the designated approvers.

- Add System Properties to Approval Policy Settings

  You selected system properties that you want to add to the approval form and allow the approver to modify the value.

- Add Custom Properties to Approval Policy Settings

  You configure custom properties that you want to add to the approval form to allow the approver to modify the value.

Approval Policy Type Settings

The approval policy type determines how the approval policy is configured and to what items or actions you can apply it in the entitlement. When you add approval levels, the policy type or item affects which fields are available to create conditions for the approval levels.

Select **Administration > Approval Polices**. Click **New**.

Table 5-75. Approval Policy Type Options

Option	Description
Select an approval policy type	Create an approval policy based on the policy request type. Select this option to define an approval policy that is applicable to all catalog items of that type. The request type can be a generic request, a catalog item request, or a resource action request. The available condition configuration options vary depending on the type. The more specific the type the more specific the configuration fields. For example, Service Catalog - Catalog Item Request provides only the fields that are common to all catalog item requests, but a Service Catalog - Catalog Item Request - Virtual Machine also includes the common options and options specific to virtual machines. The request type limits the catalog items or actions to which you can apply the approval policy.
Select an item	Create an approval policy based on a specific item. Select this option to define an approval policy that is applicable to specific items that are not available as individual items in the service catalog, only as part of a machine or other deployment. For example, software components. The available condition configuration fields are specific to the item and can be more detailed than the criteria offered for a policy type item.
List	Lists the available policy type or catalog items. Search or sort the columns to locate a specific item or type.

Add Approval Policy Settings

You configure the basic information about the approval policy, including the state to the policy, so that you can manage the policy.

To define the basic approval policy information, select **Administration > Approval Polices**. Click **New**. Select the policy type and click **OK**.

Table 5-76. Approval Policy Options

Option	Description
Name	Name that appears when applying the approval policy in an entitlement.
Description	Provide a verbose description of how the approval policy is constructed. This information will help you manage your approval policies.

Table 5-76. Approval Policy Options (continued)

Option	Description
Status	Possible values include: ■ Draft. The approval policy is not available to apply in entitlements. After you make a policy active, you can never return it to draft. ■ Active. The approval policy is available to apply in entitlements. ■ Inactive. The approval policy is not available to apply in entitlements. If the policy has not been applied to entitlements and you make inactive, you can delete the policy but you cannot reactivate it. If the policy has been applied and you make inactive, the items to which it applies must be linked to a different policy or the items are unlinked. Unlinked items and actions are still entitled to users, but they do not have an applied approval policy.
Policy Type	Displays the approval policy request type. If you selected a catalog item on which to base the approval policy, the associated request type is displayed.
Item	Displays the selected catalog item. If you selected a request type on which to base the approval policy, this field is blank.
Last Updated By	Name of the user who made changes to the approval policy.
Last Updated On	Date of the last change to the approval policy.
Pre Approval Level	To require approval before the requested items is provisioned or the actions run, configure one or more conditions that trigger an approval process when the service catalog user requests the item.
Post Approval Level	To require approval after the item is provisioned but before the provisioned or modified item is released to the requesting service catalog user, configure one or more conditions that trigger an approval process. For example, the virtual infrastructure administrator verifies that the virtual machine is in a workable state before releasing it to the service catalog user.
View Linked Entitlements	Displays all the entitlements where the approval policy is applied to services, catalog items, or actions. You can link the items in one entitlement to a different policy. This option is only available when you view an active approval policy.

Add Level Information to Approval Policy Settings

An approval level includes the conditions that trigger an approval process when the service catalog user requests the item, and any system properties and customer properties that you want to include. When triggered, the approval requests are sent to the designated approvers.

To define the basic approval policy information, select **Administration > Approval Polices**. Click **New**. Select the policy type and click **OK**. On the Pre Approval or Post Approval tab, click the **New** icon (✚).

You prioritize levels based on the order that you want them processed. When the approval policy is triggered, if the first level of approval is rejected, the request is rejected.

Table 5-77. Level Information Options

Option	Description
Name	Enter a name. The level name appears when you are reviewing requests with approval policies.
Description	Enter a level description. For example, CPU>4 to VI Admin.
When is approval required?	Select when the approval policy is triggered.
Always required	The approval policy is triggered for every request. If you select this option and apply this approval policy to eligible services, catalog items, or actions in an entitlement, then the request must be approved by the specified approver method before it is provisioned. For example, all requests must be approved by the requesting user's manager.

Table 5-77. Level Information Options (continued)

Option	Description
Required based on conditions	The approval policy is based on one or more condition clauses. If you select this option, you must create the conditions. When this approval policy is applied to eligible services, catalog items, or actions in an entitlement, then the conditions are evaluated. If the conditions are true, then the request must be approved by the specified approver method before it is provisioned. If the conditions are false, then the request is provisioned without requiring an approval. For example, any requests for a virtual machine with 4 or more CPUs must be approved by the virtual infrastructure administrator. The availability of the fields on which to base the conditions is determined by the selected approval policy type or catalog item. When you enter a value for a condition, the values are case-sensitive. To configure more than one condition clause, select the Boolean operation for the clauses. ■ All of the following. The approval is triggered when all of the clauses are true. This a Boolean AND operator between each clause. ■ Any of the following. The approval level is triggered when at least one of clauses is true. This is a Boolean OR operator between each clause. ■ Not the following. The approval level is triggered is none of the clauses are true. This is a Boolean NOT operator between each clause.
Approvers	Select the approver method.
Specific Users and Groups	Sends the approval request to the selected users. Select the users or user groups that must approve the service catalog request before it is provisioned or an action runs. For example, the request goes to the virtual infrastructure administrator group with **Anyone can approve** selected.
Determine users from the request	Sends the approval request to the users based on the defined condition. For example, if you are applying this approval policy across business groups and you want the business group manger to approve the request, select **Business group > Consumer > Users > Manager**.
Use event subscription	Processes the approval request based on defined event subscriptions. The workflow subscription must be defined in **Adminstration > Events > Subcriptions**. The applicable workflow subscriptions are pre-approval and post-approval.

Table 5-77. Level Information Options (continued)

Option	Description
Anyone can approve	Only one of the approvers must approve before the request is processed. When the item is requested in the service catalog, requests for approval are sent to all approvers. If one approver approves the request, the request is approved and the request for approval is removed from the other approvers' inboxes. If the first approver rejects the request, the requesting user is notified about the rejection and the approval request is removed from the approvers' inboxes. If the first approver approves and the approval request is open in the second approver's console, the approver is not allowed to submit the approval request. It was considered completed by the first approvers response. If you select **Specific Users and Groups** or **Determine approvers from the request**, and there is more than one approver, this is one of the additional options. If there is only one approver, this option to not apply.
All must approve	All of the specified approvers must approve before the request is processed. If you select **Specific Users and Groups** or **Determine approvers from the request**, and there is more than one approver, this is one of the additional options. If there is only one approver, this option to not apply.

Add System Properties to Approval Policy Settings

You selected system properties that you want to add to the approval form and allow the approver to modify the value.

For example, for a virtual machine approval, select CPU if you want to allow the approver to modify a request for 6 CPUs to 4 CPUs.

To select system properties, select **Administration > Approval Polices**. Click **New**. Select the policy type and click **OK**. On the Pre Approval or Post Approval tab, click the **New** icon (✚) and click the **System Properties** tab.

Table 5-78. System Properties Options

Option	Description
Properties	The list of available system properties depends on the selected request type or catalog item, and whether system properties exist for the item. Some properties are available only when the blueprint is configured in a particular way. For example, CPUs. The blueprint to which you are applying the approval policy with the CPU system property must be configured as a range. For example, CPU minimum is 2 and the maximum is 8.

Add Custom Properties to Approval Policy Settings

You configure custom properties that you want to add to the approval form to allow the approver to modify the value.

For example, for a virtual machine approval, add `VMware.VirtualCenter.Folder` if you want to allow the approver to specify the folder to which the machine is added in vCenter Server.

You can also add a custom property that is specific to this approval policy form.

To select system properties, select **Administration > Approval Polices**. Click **New**. Select the policy type and click **OK**. On the Pre Approval or Post Approval tab, click the **New** icon (✚) and click the **Custom Properties** tab.

Table 5-79. Custom Properties

Option	Description
Name	Enter the property name.
Label	Enter the label that is presented to the approver in the approval form.
Description	Enter the extended information for the approver. This information appears as the field tooltip in the form.

Modify an Approval Policy

You cannot modify an active or inactive approval policy. You must create a copy of the original policy and replace the policy that is not producing the required results. Active and inactive approval policies are read-only. You can modify approval polices that are in a draft state.

When you make the copy of the approval policy, the new policy is based on the original policy type. You can edit all of the attributes except the policy type. You do this when you want to modify the approval levels to modify, add, or remove levels, or to add system or custom properties to the forms.

You can create pre-approval and post-approval levels. For instructions about creating an approval level, see Create an Approval Level.

Prerequisites

Log in to vRealize Automation as a **tenant administrator** or **approval administrator**.

Procedure

1 Select **Administration > Approval Policies**.

2 Select the row of the approval policy to copy.

3 Click the **Copy** icon (📋).

   A copy of the approval policy is created.

4 Select the new approval policy to edit.

5 Enter a name in the **Name** text box.

6 (Optional) Enter a description in the **Description** text box.

7 Select the state of the policy from the **Status** drop-down menu.

Option	Description
Draft	Saves the approval policy in an editable state.
Active	Saves the approval policy in a read-only state that you can use in an entitlement.
Inactive	Saves the approval policy in a read-only state that you cannot use in an entitlement until you activate the policy.

8 Edit the pre-approval and post-approval levels.

9 Click **OK**.

Results

You created a new approval policy based on an existing approval policy.

What to do next

Apply the new approval policy in an entitlement. See Entitle Users to Services, Catalog Items, and Actions.

Deactivate an Approval Policy

When you determine that an approval policy is outdated, you can deactivate the policy so that it is not available during provisioning.

To deactivate an approval policy, you must assign a new policy for each entitlement to which the approval policy is currently applied.

You can later reactiveate a deactivated approval policy, or you can delete a deactivated policy.

Prerequisites

Log in to vRealize Automation as a **tenant administrator** or **approval administrator**.

Procedure

1 Select **Administration > Approval Policies**.

2 Click the approval policy name.

3 Click **View Linked Entitlements**.

   a In the **Replace All With** drop-down menu, select the new approval policy.

   If the list includes more than one entitlement, the new approval policy is applied to all the listed entitlements.

   b Click **OK**.

4 After you verify that no entitlements that are linked to the approval policy, select **Inactive** from the Status drop-menu.

5 Click **OK**.

6 To delete an approval policy, select the row containing the inactive policy.

   a Click **Delete**.

   b Click **OK**.

Results

The approval policy is unlinked from any entitlements where it is used and deactivated. You can later reactivate and reapply it to items in an entitlement.

What to do next

If you not longer need the approval policy, you can delete it. See Delete an Approval Policy.

Delete an Approval Policy

If you have approval policies that you deactivated and do not need, you can delete them from vRealize Automation.

Prerequisites

- Unlink and deactivate approval policies. See Deactivate an Approval Policy.
- Log in to vRealize Automation as a **tenant administrator** or **approval administrator**.

Procedure

1 Select **Administration > Approval Policies**.

2 Select the row containing the inactive policy.

3 Click **Delete**.

4 Click **OK**.

Results

The approval policy is deleted.

Scenario: Create and Apply CentOS with MySQL Approval Policies

As the tenant administrator for the development and quality engineering business group, you want to apply strict governance to catalog item requests. Before your users can provision the CentOS with MySQL catalog item, you want your vSphere virtual infrastructure administrator to approve the machine request and you want your software manager to approve the software request.

You create and apply one approval policy for the vSphere CentOS with MySQL service catalog request to require approval for the machine by a vSphere virtual infrastructure administrator based on specific conditions, and another approval policy for the MySQL Software component to require approval by your software manager for every request.

Approval administrators can only create the approvals, and a business group managers can apply them to entitlements. As a tenant administrator, you can both create the approvals and apply them to entitlements.

Prerequisites

- Log in to the vRealize Automation console as a **tenant administrator**. Only a tenant administrator can both create and apply approval policies.
- Ensure that the CentOS with MySQL catalog item is included in a service. See Scenario: Make a CentOS with MySQL Application Blueprint Available in the Service Catalog.

Scenario: Create a CentOS with MySQL Virtual Machine Approval Policy

As the tenant administrator you want to ensure that the development and quality engineering group receives virtual machines that are properly provisioned in your environment, so you create an approval policy that requires pre approval for certain types of requests.

Because the CentOS with MySQL virtual machine consumes vCenter Server resources, you want the vSphere virtual infrastructure administrator to approve requests when the requested memory is more than 2048 MB or more than 2 CPUs to ensure that the resources are consumed wisely. You also you give the approver the ability to modify the requested CPU and memory values before approving a request.

Procedure

1 Select **Administration > Approval Policies**.

2 Create an approval policy for virtual machine provisioning.

   a Click the **New** icon (✚).

   b Select **Select an approval policy type**.

   c In the list, select **Service Catalog - Catalog Item Request - Virtual Machine**.

   d Click **OK**.

   e Configure the following options:

Option	Configuration
Name	Enter `CentOS on vSphere CPU or Memory VM`.
Description	Enter `Requires VI Admin approval for CPU>2 or Memory>2048`.
Status	Select **Active**.

3 On the **Pre Approval** tab, click the **Add** icon (✚).

4 Configure the **Level Information** tab with the triggering criteria and the approval actions.

   a In the **Name** text box, enter `CPU>2 or Memory>2048 – VI Admin`.

   b In the **Description** text box, enter `VI Admin approval for CPU and Memory`.

   c Select **Required based on conditions**.

   d In the Clause drop-down list, select **Any of the following**.

   e In the new Clause drop-down list, select **CPUs** and configure the clause with the values **CPU > 2**.

   f Click **Add expression** and configure the clause with the values `Memory (MB) > 2048`.

   g Select **Specific Users and Groups**.

   h Enter the name of the vSphere virtual infrastructure administrator or administrator group in the search text box and click the search icon (🔍).

   i Select the user or group.

   j Select **Anyone can approve**.

   The request only needs one virtual infrastructure administrator to verify the resources and approve the request.

5 Click the **System Properties** tab and select the properties that allow the approver to modify the requested CPU and Memory values before approving a request.

   a Select the **CPUs** and **Memory (MB)** check boxes.

   b Click **OK**.

6 Click **OK**.

Results

You created an approval policy for virtual machine requests, but you still want to create an approval for the MySQL component. Until you apply the policies to an entitlement, no approvals are triggered.

Scenario: Create a MySQL Software Component Approval Policy

As the tenant administrator, your software managers asked you to create and apply approval policies for MySQL installations to track licensing usage. You create a policy to notify the software license manager whenever the MySQL for Linux Virtual Machines Software component is requested.

In some environments you might need this type of approval because license keys must be provided by the software manager. In this scenario, you only need the software manager to track and approve the request. After you create the approval policy, you apply the policy to the MySQL for Linux Virtual Machines catalog item. This approval policy is very specific and can only be applied to the MySQL for Linux Virtual Machines Software component in the entitlements.

Procedure

1. Select **Administration > Approval Policies**.
2. Create an approval policy for the MySQL Software component.
    - a Click the **New** icon (✚).
    - b Select **Select an item**.
    - c Select **MySQL for Linux Virtual Machines**.
    - d Click **OK**.
    - e Configure the following options:

Option	Configuration
Name	Enter `MySQL tracking approval`.
Description	Enter `Approval request sent to software manager`.
Status	Select **Active**.

3. On the **Pre Approval** tab, click the **Add** icon (✚).
4. Configure the **Level Information** tab with the triggering criteria and the approval actions.
    - a In the **Name** text box, enter `MySQL software deployment notice`.
    - b In the **Description** text box, enter `Software mgr approval of software installation`.
    - c Select **Always required**.
    - d Select **Specific Users and Groups**.
    - e Enter the name of the software manager in the search text box and click the search icon (🔍) and select the user.
    - f Select **Anyone can approve**.

      The request only needs one software manager to approve the request.

      Click **OK**.
5. Click **OK**.

Results

You created the approval policies for virtual machines and for MySQL for Linux Virtual Machines Software components. Until you apply the approval policies to an entitlement, no approvals are triggered.

Scenario: Apply Approval Policies to CentOS with MySQL Components

As the tenant administrator, you can create approval policies and entitlements. You modify the Dev and QE entitlement to apply the approval policies that you created so that approvals are triggered when a service catalog user requests the item.

While it might be easier to entitle the entire catalog service to your business group, it does not allow you to have the same control and governance as when you create individual entitlements for catalog items. For example, if you entitle users to a service, they can request any catalog items that are in the service and all items that are added to the service in the future. It also means that you can only use very high-level approval policies that apply to every catalog item in the service, such as always requiring approval from a manager. If you choose to entitle catalog items individually, you can create and apply very specific approval policies for each item and tightly control who can request which items in the service. If you choose to entitle the individual components of catalog items individually, you can have even greater control.

If you do not know what approval policies you want to apply to items in an entitlement, you can return later and apply them. In this scenario, you apply different approval policies to two components of the same published application blueprint.

Procedure

1. Select **Administration > Catalog Management > Entitlements**.
2. Click the **Dev and QE Entitlement**.
3. Click the **Items and Approvals** tab.
4. Add the CentOS with MySQL machine and apply the approval policy.

    a. Click the **Add Items** icon (✚) beside the Entitled Items heading.

    b. Select the **CentOS with MySQL** check box.

    c. Click the **Apply this policy to selected items** drop-down arrow.

       The CentOS on vSphere CPU and Memory policy is not in the list.

    d. Click **Show all** and click the down-arrow to view all approval policies.

    e. Select **CentOS on vSphere CPU and Memory [Service Catalog - Catalog Item Request - Virtual Machine]**.

       The vSphere CentOS machine is a machine blueprint in an application blueprint. Review the policy names so that you select the one that is appropriate to your catalog item type. If you apply the wrong policy, the approval policy fails or triggers approval requests based on incorrect conditions.

    f. Click **OK**.

5. Add the MySQL for Linux Virtual Machine software component as an item and apply an approval policy to the MySQL item.

    a. Click the **Add Catalog Items and Components** icon (✚) beside the Entitled Catalog Items and Components heading.

    b. In the **Catalog Items and Components** drop-down menu, select **No**.

       Software components are always associated with a machine. They are not available to individually request in the service catalog.

c Select the **MySQL for Linux Virtual Machines** check box.

d Click the **Apply this policy to selected items** drop-down arrow.

e Select **MySQL tracking approval [Service Catalog - Catalog Item Request - Software Component]**.

You do not need the advanced option because the approval policy was created for this specific software component, which is added to a virtual machine.

f Click **OK**.

6 Add actions that the users can run on the provisioned machine.

Approval policies are not applied to actions in this scenario.

a Click the **Add Actions** icon (✚) beside the Entitled Actions heading.

b Select the following actions.

Name / Type	Description
Create Snapshot / Virtual Machine	Creates a snapshot of the virtual machine, including the installed software. Allows the developers to create snapshots to which they can revert during development.
Destroy / Deployment	Destroys the entire provisioned blueprint, not just the machine. Use this action to avoid orphaned components.
Power Off / Machine	Turns the virtual machine off.
Power On / Machine	Turns the virtual machine on.
Revert to Snapshot / Virtual Machine	Reverts to a previously created snapshot.

c Click **OK**.

7 Click **Finish**.

Results

This entitlement allows you to require different approvals on different blueprint components.

What to do next

Request the CentOS with MySQL item in the service catalog as a member of the business group to verify that the entitlement and the approvals are behaving as expected.

Request Machine Provisioning By Using a Parameterized Blueprint

When you request machine provisioning for a vSphere machine blueprint that has been designed to include the size or image component profiles, you specify provisioning setting by selecting an available value set.

When you request provisioning, you can select from available `Size` and `Image` options. When you choose one of the value sets, the corresponding property values are bound to the request.

The component profile value set is applied to all vSphere machines in a cluster.

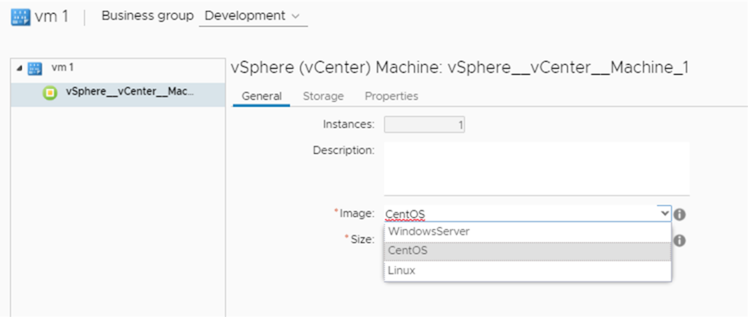
For information about component profile configuration, see Understanding and Using Blueprint Parameterization.

Prerequisites

- Define value sets for `Size` or `Image` component profiles. See Configure Component Profile Size Settings for Catalog Deployments and Configure Component Profile Image Settings for Catalog Deployments.
- Create a blueprint that contains a vSphere machine component that contains an `Image` or `Size` component profile. See Configure a Machine Blueprint and vSphere Machine Component Settings in vRealize Automation.
- Publish the blueprint to the catalog. See Publish a Blueprint.
- Configure the blueprint in the catalog. See Checklist for Configuring the Service Catalog and Examples of Approval Policies Based on the Virtual Machine Policy Type.

Procedure

1. Click **Catalog**.
2. Select the catalog service to request and click **Request**.
3. Select the vSphere machine component to provision and specify the number of instances to provision.
4. Select an image value set option from the **Image** drop-down menu.



vm 1 | Business group Development

vSphere (vCenter) Machine: vSphere__vCenter__Machine_1

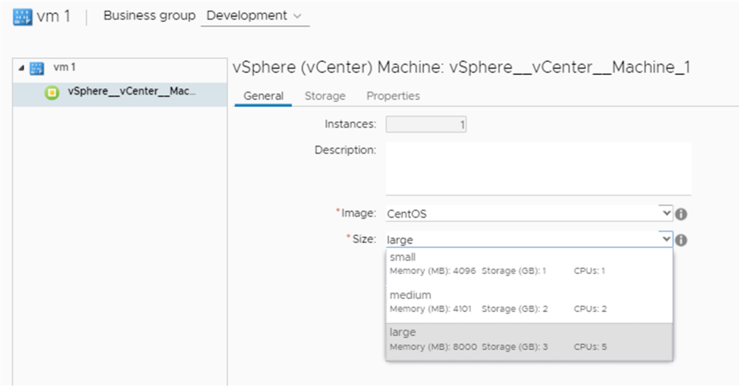
Instances: 1

Description:

*Image: CentOS

*Size:

5 Select a size value set option from the **Size** drop-down menu.



Business group: Development

vSphere (vCenter) Machine: vSphere__vCenter__Machine_1

Instances: 1

Description:

*Image: CentOS

*Size: large

small
Memory (MB): 4096 Storage (GB): 1 CPUs: 1

medium
Memory (MB): 4101 Storage (GB): 2 CPUs: 2

large
Memory (MB): 8000 Storage (GB): 3 CPUs: 5

6 Click **Submit**.

What to do next

The value sets that you defined for the `Size` and `Image` component profiles are now available on the **Image** and **Size** drop-down menus on the **Catalog** tab in the catalog provisioning request form.

Scenario: Make a CentOS with MySQL Application Blueprint Available in the Service Catalog

As the tenant administrator, you requested that your blueprint architects create a catalog item for MySQL on CentOS, on which your development and quality engineering group can run test cases. Your software architect has informed you that the catalog item is ready for users. To make the item available to your business users, you need to associate the blueprints and Software component with a catalog service and then entitle the business group members to request the catalog item.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **catalog administrator**.
- Publish a blueprint for MySQL on a vSphere CentOS virtual machine. See the processes for creating machine and software component blueprints in Building Your Design Library.
- If you create blueprints in a development environment, import your blueprint into your production environment. See Exporting and Importing Blueprints and Content.

- Create a reservation to allocate vSphere resources to your Dev and QE business group. See Create a Reservation for Hyper-V, KVM, SCVMM, vSphere, or XenServer.

Procedure

1 Scenario: Create a Development and Quality Engineering Catalog Service

   As the tenant administrator, you want to create a separate catalog service for your development and quality engineering group so your other groups, such as finance and human resources, don't see the specialized catalog items. You create a catalog service called Dev and QE Service to publish all the catalog items development and engineering need to run their test cases.

2 Scenario: Add CentOS with MySQL to Your Dev and QE Service

   As the tenant administrator, you want to add the CentOS with MySQL catalog item to the Dev and QE service.

3 Scenario: Entitle Users to Request Dev and QE Service Items as a Catalog Item

   As the tenant administrator, you create a Dev and QE entitlement and add the catalog items and some relevant actions so your development and quality engineering users can request the CentOS with MySQL catalog item, and run actions against the machine and the deployment.

Scenario: Create a Development and Quality Engineering Catalog Service

As the tenant administrator, you want to create a separate catalog service for your development and quality engineering group so your other groups, such as finance and human resources, don't see the specialized catalog items. You create a catalog service called Dev and QE Service to publish all the catalog items development and engineering need to run their test cases.

Procedure

1 Select **Administration > Catalog Management > Services**.

2 Click the **New** icon (✚).

3 Enter the name `Dev and QE Service` in the **Name** text box.

4 Enter the description `Dev and QE application catalog items for test cases` in the **Description** text box.

5 Select **Active** from the **Status** drop-down menu.

6 As the catalog administrator who is creating the service, use the search option to add your name as the Owner.

7 Add the Support Team custom user group.

   For example, add a custom user group that includes the IaaS architects and software architects so that you and the service catalog users have someone to contact if you encounter problems provisioning the catalog items.

8 Click **OK**.

Results

You created and activated a Dev and QE catalog service, but it doesn't contain any catalog items yet.

Scenario: Add CentOS with MySQL to Your Dev and QE Service

As the tenant administrator, you want to add the CentOS with MySQL catalog item to the Dev and QE service.

Procedure

1. Select **Administration** > **Catalog Management** > **Services**.
2. Select the Dev and QE Service row in the **Services** list and click **Manage Catalog Items**.
3. Click the **New** icon (✚).
4. Select **CentOS with MySQL**.

   Only published blueprints and components that are not yet associated with a service appear in the list. If you do not see the blueprint, verify that it was published or that it is not included in another service.
5. Click **OK**.
6. Click **Close**.

Results

You published the CentOS with MySQL catalog item to the Dev and QE service, but until you entitle users to the item or the service, no one can see or request the item.

Scenario: Entitle Users to Request Dev and QE Service Items as a Catalog Item

As the tenant administrator, you create a Dev and QE entitlement and add the catalog items and some relevant actions so your development and quality engineering users can request the CentOS with MySQL catalog item, and run actions against the machine and the deployment.

In this scenario, you entitle the service because you want users to be entitled to any future catalog items that are added to this service. You also want to allow your users to manage their provisioned deployment, so you add actions like power on and off, snapshot, and destroy deployment to the entitlement.

Procedure

1. Select **Administration** > **Catalog Management** > **Entitlements**.
2. Click the **New** icon (✚).
3. Configure the details.

   a. Enter the name `Dev and QE Entitlement` in the **Name** text box.

   b. In the **Status** drop-down menu, select **Active**.

c In the **Business Group** drop-down menu, select the **Dev and QE** group.

d In the Users and Groups area, add one or more users.

Add yourself only, unless you are certain that the blueprint is working as intended. If it is, you can add individual users and you can add custom user groups.

e Click **Next**.

4 Add the service.

Although you are adding the CentOS and MySQL catalog items separately, adding the service ensures that any addition items that you add to the service at a later date are available to the business group members in the service catalog.

a Click the **Add Services** icon (✚) beside the Entitled Services heading.

b Select **Dev and QE Service**.

c Click **OK**.

Dev and QE service is added to the Entitled Services list.

5 Add actions.

a Click the **Add Actions** icon (✚) beside the Entitled Actions heading.

b Click the Type column header to sort the list.

Select the following actions based on type. These actions are useful to the development and quality engineering users working with their test case machines, and are the only actions that you want these business group members to use.

Type	Action Name
Machine	Power On
Machine	Power Off
Virtual Machine	Create Snapshot
Virtual Machine	Revert To Snapshot
Deployment	Destroy The deployment destroy action destroys the entire deployment and not just the virtual machine.

c Click **OK**.

The five actions are added to the Entitled Actions list.

6 Click **Finish**.

Results

You added the CentOS with MySQL catalog item to your new Dev and QE catalog service and entitled your business group members to request and manage the item.

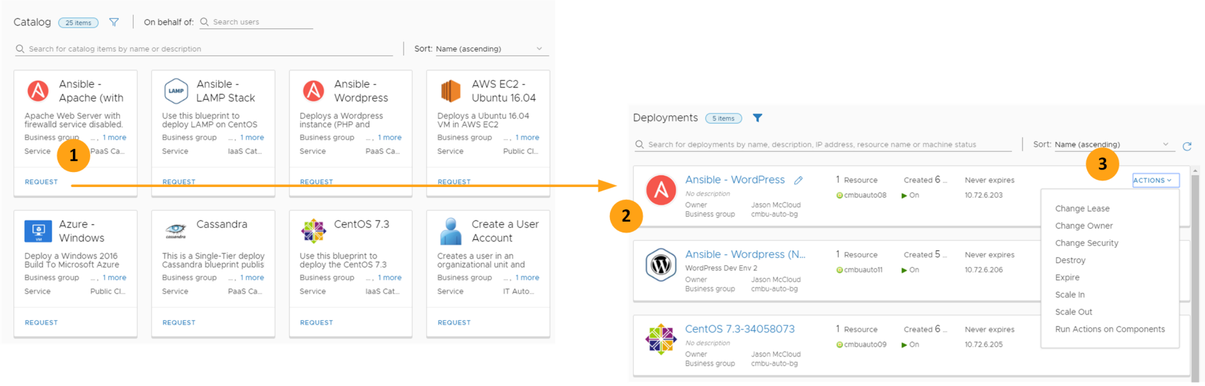
What to do next

After you verify your work by provisioning the CentOS with MySQL catalog item, you can add additional users to the entitlement to make the catalog item publicly available to your development and quality engineering users. If you want to further govern the provisioning of resources in your environment, you can create approval policies for the MySQL Software component and the CentOS for Software Testing machine. See Scenario: Create and Apply CentOS with MySQL Approval Policies.

Using the Catalog and Managing Deployments

6

The catalog is your available blueprints and deployments are your provisioned blueprints. Your administrator provides the catalog items. You can then request and manage the resources as deployments. As part of managing deployments, you can run actions to make changes.



The following workflow begins with the catalog.

1 You request items in the catalog. The catalog contains published blueprints that are entitled to the business groups that you are a member of.

2 The provisioned resources are managed as deployments. You can monitor the provisioning process, manage your deployments, and run actions on your deployments.

3 You use the actions to make changes on the deployment after it is deployed. Actions might include increasing memory, decreasing CPU, or destroying the deployment when you no longer need it.

This chapter includes the following topics:

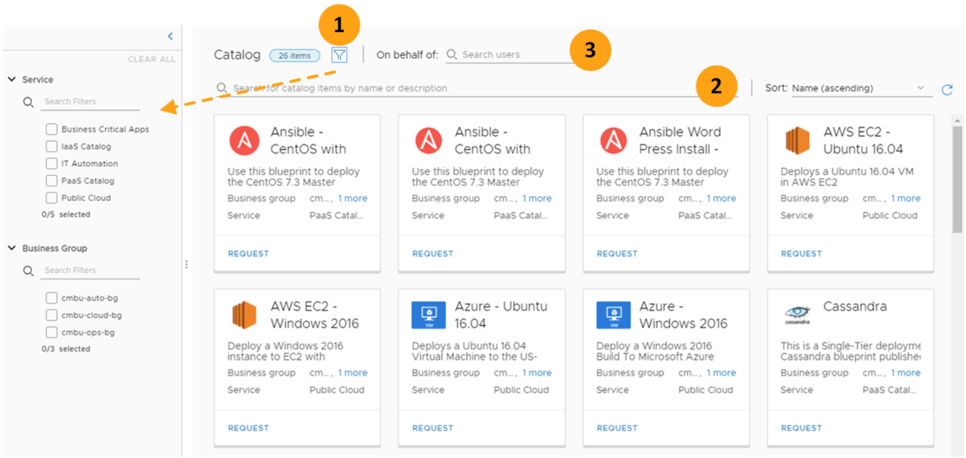
Working with the Catalog

The catalog is the list of blueprints that you can deploy. The blueprint architect determines the design of the components, what custom options you can select when you request the item, and where it is deployed based on your organizations vRealize Automation endpoints.

The available catalog items are based on your membership in one or more business groups and how your business groups are entitled to provision the blueprints.

Finding Catalog Items

This example shows a small catalog. In larger enterprise environments, you might have more than fits on one page.


1
2
3
CLEAR ALL
Service
Search Filters
□ Business Critical Apps
□ IaaS Catalog
□ IT Automation
□ PaaS Catalog
□ Public Cloud
0/5 selected
Business Group
Search Filters
□ cmbu-auto-bg
□ cmbu-cloud-bg
□ cmbu-ops-bg
0/3 selected
Catalog 26 items
On behalf of: Search users
Search for catalog items by name or description
Sort: Name (ascending)
Ansible - CentOS with
Use this blueprint to deploy the CentOS 7.3 Master
Business group cm..., 1 more
Service PaaS Catal...
REQUEST
Ansible - CentOS with
Use this blueprint to deploy the CentOS 7.3 Master
Business group cm..., 1 more
Service PaaS Catal...
REQUEST
Ansible Word Press Install -
Use this blueprint to deploy the CentOS 7.3 Master
Business group cm..., 1 more
Service PaaS Catal...
REQUEST
AWS EC2 - Ubuntu 16.04
Deploys a Ubuntu 16.04 VM in AWS EC2
Business group cm..., 1 more
Service Public Cloud
REQUEST
AWS EC2 - Windows 2016
Deploy a Windows 2016 instance to EC2 with
Business group cm..., 1 more
Service Public Cloud
REQUEST
Azure - Ubuntu 16.04
Deploys a Ubuntu 16.04 Virtual Machine to the US-
Business group cm..., 1 more
Service Public Cloud
REQUEST
Azure - Windows 2016
Deploy a Windows 2016 Build To Microsoft Azure
Business group cm..., 1 more
Service Public Cloud
REQUEST
Cassandra
This is a Single-Tier deployme Cassandra blueprint publishe
Business group cm..., 1 more
Service PaaS Catal...
REQUEST

Use the following options to find the blueprint that you want to deploy.

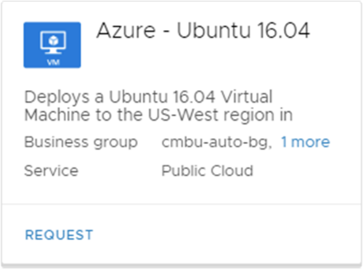
1 **Filter** the list based on services and business groups.

2 **Search** and **Sort** to locate and organize the catalog items.

3 Select an **On behalf of** user to limit number of catalog items, and then request the item for that user. You can only deploy blueprints that are entitled to business groups that the user is a member of. When you select the user name, the list of available catalog items reflects that membership. On behalf of permission is available to administrators, business group managers, and can be assigned to one or more business group members when you configure the business group. See Create a Business Group.

Catalog Cards

Catalog cards represent the blueprint that might deploy single machines or an entire application. They can also represent XaaS workflows that provision in other ways. For example, add users to Active Directory.



Azure - Ubuntu 16.04

Deploys a Ubuntu 16.04 Virtual Machine to the US-West region in

Business group	cmbu-auto-bg, 1 more
Service	Public Cloud

The information on the card includes the business groups that are entitled to request the catalog item, and the service that the item is associated with.

How to Submit a Catalog Request

When you submit a catalog request, the request form for each blueprint might be different. The differences in the forms are configured by your blueprint designer.

The form variations are based on how much you are allowed to customize your request. You might have multiple options that you can select to customize your request, or you might have no options.

For example, the blueprint architect might design a blueprint so that you can select a specific number of CPUs or where you select large, medium, or small, each of which is a predetermined number of CPUs. Or a blueprint might be proscriptive, not allowing any changes to the blueprint before you submit it.

After the request is successfully provisioned, the deployed workload or service is yours to manage.

Prerequisites

- You must be a member of a business group that is entitled to one or more catalog items. See Creating Entitlements.
- If you are deploying on behalf of another user, you must be assigned the support role in the business group. See Create a Business Group.

Procedure

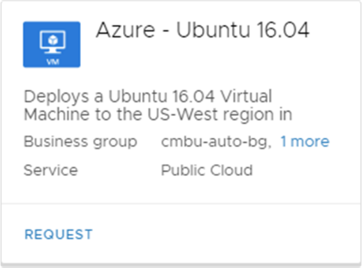
1 Click **Catalog**.

2 If you are assigned the support role in one or more business groups, and you are deploying on the behalf of other group member, enter the user or custom group name in the **On behalf of** search area.

   The list of catalog items is limited to items that are entitled to the business groups the selected user or group is a member of.

   If you do not select a user, the request is submitted for you.

3 Use the search and sort options to locate the item you want to deploy, and click **Request**.



Azure - Ubuntu 16.04

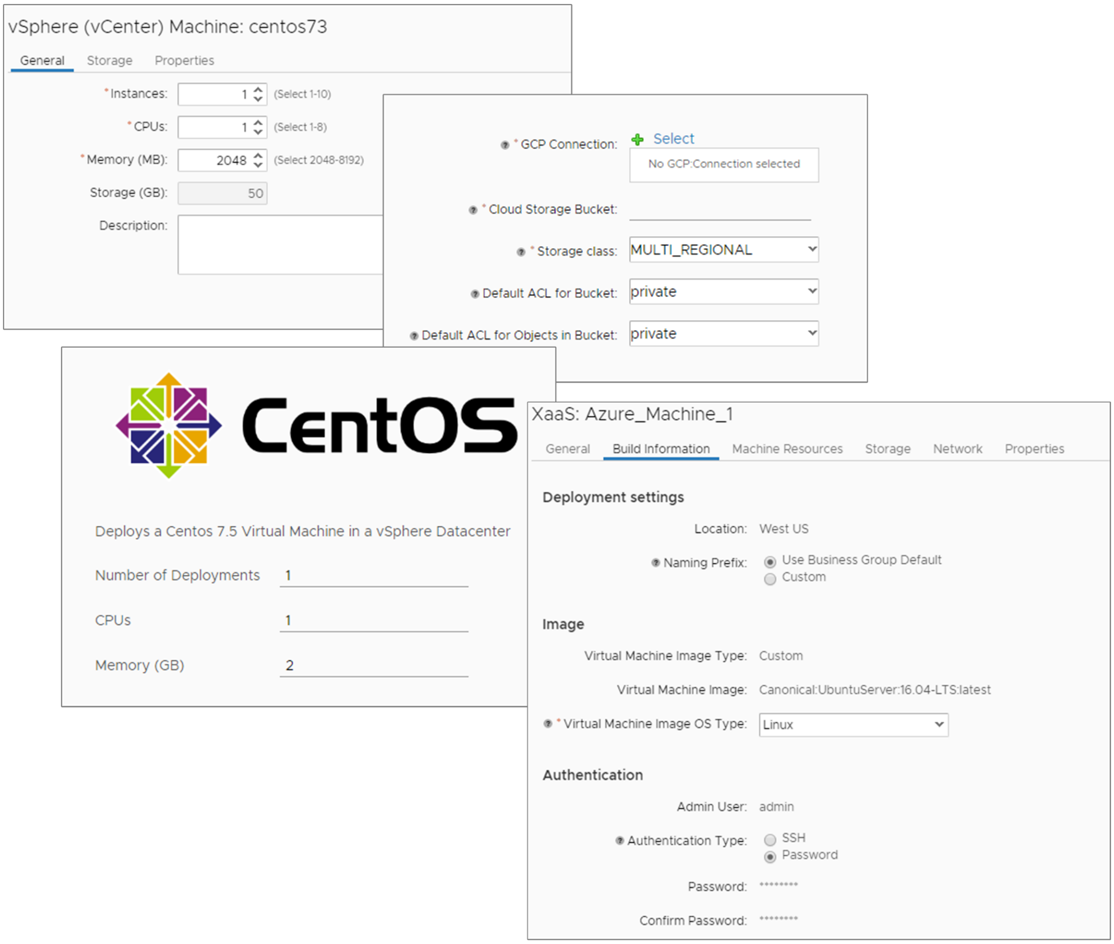
Deploys a Ubuntu 16.04 Virtual Machine to the US-West region in

Business group	cmbu-auto-bg, 1 more
Service	Public Cloud

4 If you are a member of more than one business group that is entitled to the blueprint, select the business group to associate with the deployment.

5 On the request form, configure any required and available options.

Depending on how the blueprint is configured, the form might vary. The following are examples that range from simple to more complex with multiple tabs.



vSphere (vCenter) Machine: centos73

* Instances:	1	(Select 1-10)
* CPUs:	1	(Select 1-8)
* Memory (MB):	2048	(Select 2048-8192)
Storage (GB):	50	
Description:		

* GCP Connection:	Select No GCP:Connection selected
* Cloud Storage Bucket:	
* Storage class:	MULTI_REGIONAL
Default ACL for Bucket:	private
Default ACL for Objects in Bucket:	private

CentOS

Deploys a Centos 7.5 Virtual Machine in a vSphere Datacenter

Number of Deployments	1
CPUs	1
Memory (GB)	2

XaaS: Azure_Machine_1

Deployment settings

Location:	West US
Naming Prefix:	● Use Business Group Default ○ Custom

Image

Virtual Machine Image Type:	Custom
Virtual Machine Image:	Canonical:UbuntuServer:16.04-LTS:latest
* Virtual Machine Image OS Type:	Linux

Authentication

Admin User:	admin
Authentication Type:	○ SSH ● Password
Password:	********
Confirm Password:	********

6 Click **Submit**.

Results

The request is submitted for provisioning and the Deployments tab opens so that your can track the progress of your request.

What to do next

Verify that your request is deployed. See Monitoring Provisioning Requests.

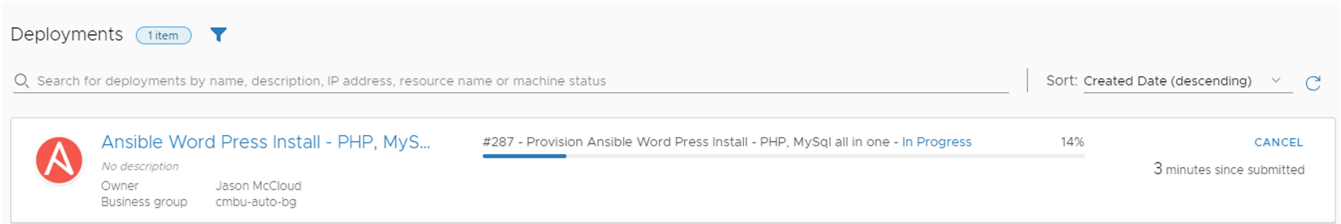
Working with Your Deployments

Deployments are provisioned blueprints that you requested from the catalog. You can monitor the status of submitted requests throughout the provisioning process, track your deployed resources, and manage those deployed resources using actions.

Monitor Requests Status

Requests that are in progress appear on the Deployments tab. You use the card to track the provisioning process to completion.

If the provisioning process fails, you can review the error message and events to determine where the request failed and resolve the problem. See Testing and Troubleshooting Failed Provisioning Requests.


Deployments
1 item
Search for deployments by name, description, IP address, resource name or machine status
CANCEL

Sort: Created Date (descending)

Ansible Word Press Install - PHP, MyS...

No description

Owner Jason McCloud

Business group cmbu-auto-bg

#287 - Provision Ansible Word Press Install - PHP, MySql all in one - In Progress 14%

3 minutes since submitted

Manage Deployed Resources

You manage requests on the Deployments tab.

Management includes verifying that the deployment is turned on. It might also mean changing the deployment to meet your needs by scaling it in or out. Or you might need to review the deployment details. For more information, see Managing Deployed Catalog Items.

Monitoring Provisioning Requests

You use deployments to monitor the progress of a request that you made in the catalog. If the resource is successfully provisioned, you can also manage the deployed resource.

If you do not see a request that is in progress, it was not submitted or it completed.

Monitor Requests

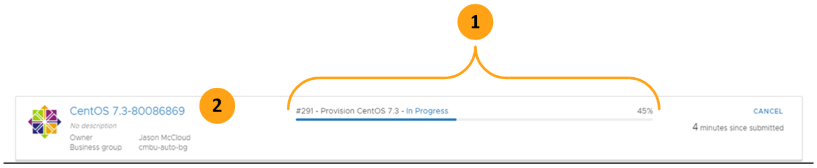
To monitor catalog requests, select **Deployments**.

Track the status of your request in the deployments list.

1 Track the status of your request on the deployment card (1). If it is the first time that the catalog item was requested, the status bar shows progress without a percentage. After the first deployment, subsequent requests provide the calculated percent complete.

  If you run an action on deployed resource, the status bar indicates the status of the change that you selected.

2 To see the in-progress details, click the deployment status bar (1) or the deployment name (2).



CentOS 7.3-80086869

No description

Owner Jason McCloud

Business group cmbu-auto-bg

#291 - Provision CentOS 7.3 - In Progress 45%

CANCEL

4 minutes since submitted

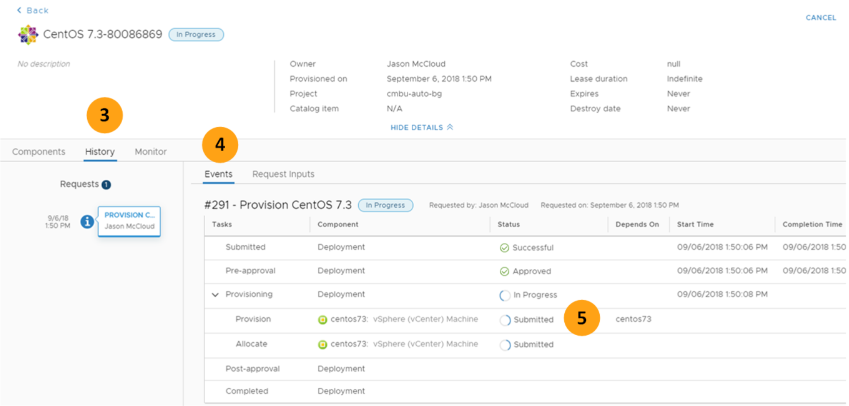
Review the provisioning details during the deployment process.

1 The History tab (3) provides the deployment events and the input values.

2 The Events tab (4) provides the details of the provisioning request.

3 You can review the provisioning workflow (5) to identify what components are currently being deployed.

If a request does not complete the provisioning process, see Testing and Troubleshooting Failed Provisioning Requests.



< Back

CentOS 7.3-80086869 In Progress

CANCEL

No description

Owner	Jason McCloud	Cost	null
Provisioned on	September 6, 2018 1:50 PM	Lease duration	Indefinite
Project	cmbu-auto-bg	Expires	Never
Catalog item	N/A	Destroy date	Never

HIDE DETAILS

Components History Monitor

Requests 1

9/6/18 1:50 PM PROVISION C... Jason McCloud

Events Request Inputs

#291 - Provision CentOS 7.3 In Progress Requested by: Jason McCloud Requested on: September 6, 2018 1:50 PM

Tasks	Component	Status	Depends On	Start Time	Completion Time
Submitted	Deployment	✓ Successful		09/06/2018 1:50:06 PM	09/06/2018 1:50
Pre-approval	Deployment	✓ Approved		09/06/2018 1:50:06 PM	09/06/2018 1:50
Provisioning	Deployment	In Progress		09/06/2018 1:50:08 PM	
Provision	centos73: vSphere (vCenter) Machine	Submitted	centos73		
Allocate	centos73: vSphere (vCenter) Machine	Submitted			
Post-approval	Deployment				
Completed	Deployment				

Canceling In-Progress Requests

If you submitted a request, but then decide to cancel it, the provisioning process stops and any deployed resources are rolled back and cleaned up.

If the canceling process is taking too long, you can request that your administrator force the cancellation. As the administrator, you can cancel a request that is in a cancelling state. If you force the cancellation, the roll back might not be completed and you must manually clean up resources on the target system.

Troubleshooting Failed Catalog Requests

When you request a catalog item, it might fail for several reasons. It might be due to network traffic, insufficient endpoint resources, or a flawed blueprint specification. Or, the provisioning request succeeded, but the deployment does not appear to be working. You can use vRealize Automation to examine your deployment, review any error messages, and determine if the problem is in the environment that you can resolve.

If your role in vRealize Automation is as a catalog consumer and you do not have administrator privileges, you can use this workflow do initial troubleshooting. You might need someone in your organization to do more indepth research.

Possible Failure States

If a provisioning request fails, you see one of the following states.

- **Failed.** A request can fail for several reasons. One cause is that the provisioning process did not work due to a lack of resources on the target endpoint, insufficient resources to support the blueprint, or a badly designed blueprint that must be fixed. Another cause is that the request required approval from someone in your organization, and approver rejected the request. It is also possible that an action that you ran on a deployment failed. The failure can be caused by the environmental or approval reasons already mentioned.

  Use the following troubleshooting workflow to investigate the cause of the problem. If you are able the resolve the problem, review your action options regarding **Dismiss** and **Resubmit**. See Action Menu Commands for Provisioned Resources.

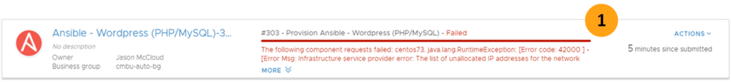
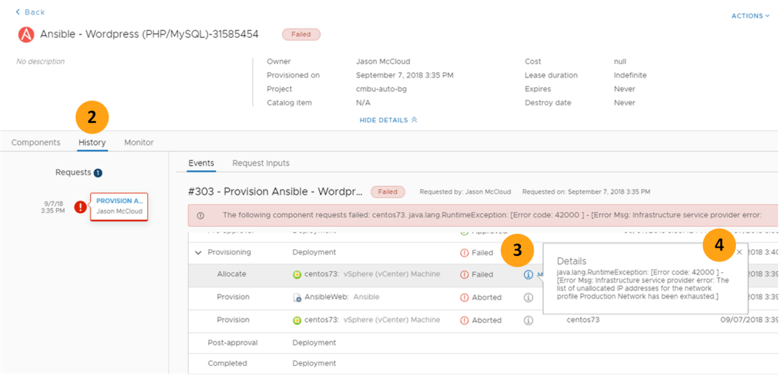
- **Partially Successful.** A request can be partially successful, meaning some components were deployed but not all of the provisioning steps completed successfully.

  Use the following troubleshooting workflow to determine which components were only partially successful and investigate the cause of the problem. If you are able the resolve the problem, review your action options regarding **Dismiss** and whether you can use **Resume**. See Action Menu Commands for Provisioned Resources and How the Resume Action Works.

Troubleshooting Workflow for Catalog Consumers

You can use this workflow to begin investigating a failed deployment. If your investigation reveals that the failure was due to a transient environmental problem, you can resolve the error and resubmit the request. If the problem is with the request specification, you might need to contact your blueprint architect.

Table 6-1. How to Begin Troubleshooting Errors

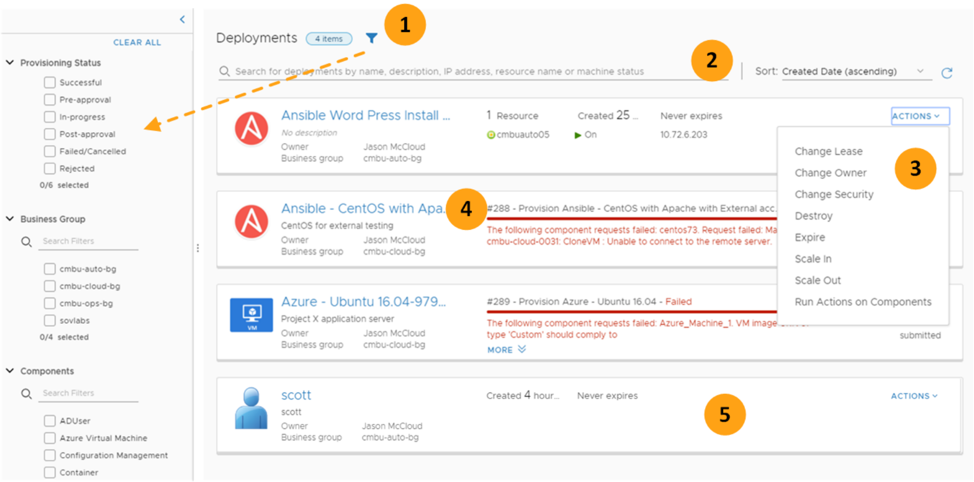
Workflow	Troubleshooting Step	Example
1	On the **Deployments** tab, failed deployments are indicated on the status bar. The card includes the last failure message. For more information, click the deployment name or progress bar.	 Ansible - Wordpress (PHP/MySQL)-3... No description Owner Jason McCloud Business group cmbu-auto-bg #303 - Provision Ansible - Wordpress (PHP/MySQL) - Failed The following component requests failed: centos73. java.lang.RuntimeException: [Error code: 42000] - [Error Msg: Infrastructure service provider error: The list of unallocated IP addresses for the network MORE ACTIONS 5 minutes since submitted 1
2	On the deployment details **History** tab, you can use the events workflow to see where the provisioning process failed. This workflow is also useful when you run an action on a deployment, but the change fails.	 Back ACTIONS Ansible - Wordpress (PHP/MySQL)-31585454 Failed No description Owner Jason McCloud Provisioned on September 7, 2018 3:35 PM Project cmbu-auto-bg Catalog item N/A Cost null Lease duration Indefinite Expires Never Destroy date Never HIDE DETAILS Components History Monitor Requests 1 9/7/18 3:35 PM PROVISION A... Jason McCloud Events Request Inputs #303 - Provision Ansible - Wordpr... Failed Requested by: Jason McCloud Requested on: September 7, 2018 3:35 PM The following component requests failed: centos73. java.lang.RuntimeException: [Error code: 42000] - [Error Msg: Infrastructure service provider error: Provisioning Deployment Failed 2018 3:4 Allocate centos73: vSphere (vCenter) Machine Failed 2018 3:3 Provision AnsibleWeb: Ansible Aborted 2018 3:3 Provision centos73: vSphere (vCenter) Machine Aborted centos73 09/07/2018 3:3 Post-approval Deployment Completed Deployment Details java.lang.RuntimeException: [Error code: 42000] - [Error Msg: Infrastructure service provider error: The list of unallocated IP addresses for the network profile Production Network has been exhausted.] 2 3 4
3	The failed status indicates where the workflow failed.	
4	The information provides a more verbose version of the error message. If this information in the signpost help is not sufficient to identify and resolve the problem, you can do additional research in the event logs. To view the event logs, you must have the necessary user role. You blueprint architect or administrator can do additional troubleshooting. See Testing and Troubleshooting Failed Provisioning Requests.	

Managing Deployed Catalog Items

As the owner of a deployment, or as an administrator who assists other users, you can use the deployment details to manage the lifecycle of deployed items. The deployment details provide the current information about each component and use the history to track changes over time. When working with the deployments, you can use the actions to modify the deployed items. There are also some changes that you can make that do not use the actions.

Managing Deployments from the Cards

The deployment card list provides an overview of your deployments. Did they succeed? Are they running?



The following component requests failed: centos73. Request failed: Ma cmbu-cloud-0031: CloneVM : Unable to connect to the remote server.

The following component requests failed: Azure_Machine_1. VM image ... type 'Custom' should comply to

Use the following options to find and manage your deployed resource from vRealize Automation.

1 **Filter** the list based on the current status of the request, the business group it was deployed for, what subcomponents are included, the owning user, and provisioning or expiration date ranges. The Provisioning Status and Request Number filters apply only to the initial provisioning process, not to any subsequent actions that you might run. The other filters apply to the deployment in general.

2 **Search** and **Sort** to locate and organize your deployments.

3 To manage the deployment, click **Actions** to run entitled deployment-level actions. You must open the deployment details to run actions on individual components. The actions might be standard actions that you entitled for design blueprints, or they might be custom XaaS resource actions that you created and entitled for the XaaS blueprint. re For more about the standard actions, see Running Actions on Deployed Resources.

4 To view and manage the deployment details, including provisioning events, history, and component-level actions, click the deployment name. The top three represent initial provisioning requests for standard blueprints.

5 You can also manage XaaS deployment requests that run workflows. The workflows can result in resources or the workflows run on external systems. In this example the XaaS added a user to an Active Directory domain.

Managing a Deployment Using the Deployment Details

You use the deployment details to perform the following management information.

- **Details.** The basic information that you find on the card. You can also change the deployment name and description, and run deployment-level actions.
- **Components tab.** The full configuration of each component. You can also run component-level actions.
- History tab. The complete history of the changes made to the deployment. You can also find more information about placement and what input values were provided for each change.
- **Monitor tab.** If you integrate with vRealize Operations Manager, the monitoring metric data and alerts appear for the deployment and the components.
- **Actions.** Using the details, you can also run deployment-level actions or component-level actions.

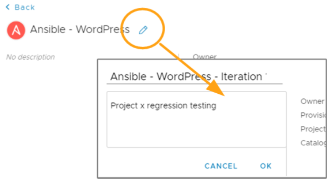
Using the Deployment Details

The deployment details provide more than the basic information that you find on the card. You can also change the deployment name and description, and run deployment and component-level actions.

Review the basic information about the deployment, including the blueprint it was deployed from and the cost.

Change the Deployment Name

The deployment takes its name from the blueprint. This name does not always have meaning to you as you work with your deployments. You can update the name and description to one that suits your needs.



Ansible - WordPress - Iteration '

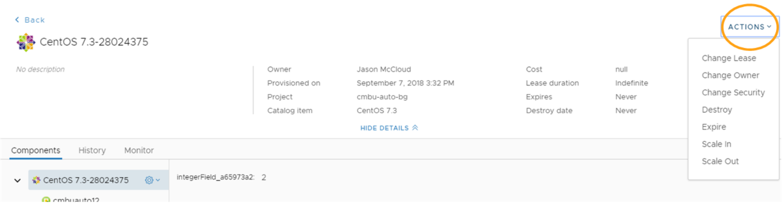
Project x regression testing

1 Point to the name and then click the pencil icon.

2 Update the name and description to something that is meaningful to you.

Run Deployment-Level Actions

Deployment-level actions are limited to changes that affect the entire deployment. The list of available actions depends on how your business group is entitled to use them.



CentOS 7.3-28024375

No description

Owner Jason McCloud
Provisioned on September 7, 2018 3:32 PM
Project cmbu-auto-bg
Catalog item CentOS 7.3

Cost null
Lease duration Indefinite
Expires Never
Destroy date Never

CentOS 7.3-28024375

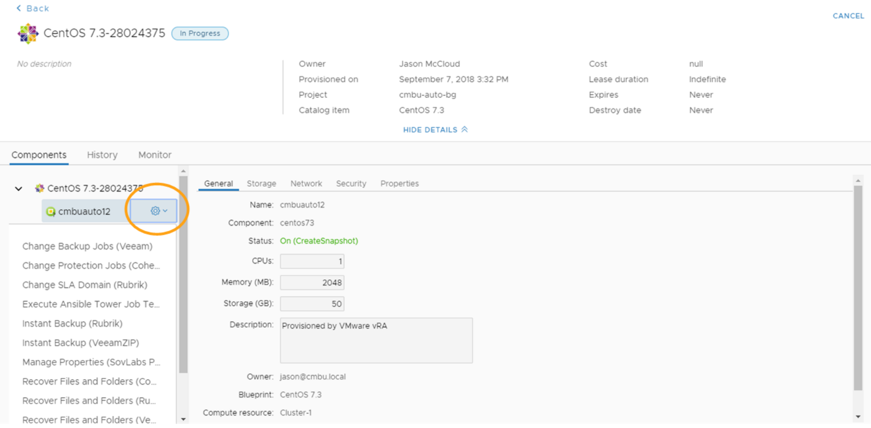
integerField_a65973a2: 2

Deployment Components

The Components tab in the deployment details provides the full configuration of all the deployment components. You can see how the machines and networks are configured. You can also run component-level actions to change the configuration.

Review the component details when you must understand the deployment that was provided to you or when you are troubleshooting a problem with the instance.

Any changes that you make using the actions are reflected in the details.


Back
CANCEL
In Progress
HIDE DETAILS
Components
History
Monitor
CentOS 7.3-28024375
cmbuauto12
Change Backup Jobs (Veeam)
Change Protection Jobs (Cohe...
Change SLA Domain (Rubrik)
Execute Ansible Tower Job Te...
Instant Backup (Rubrik)
Instant Backup (VeeamZIP)
Manage Properties (SovLabs P...
Recover Files and Folders (Co...
Recover Files and Folders (Ru...
Recover Files and Folders (Ve...
General
Storage
Network
Security
Properties

CentOS 7.3-28024375

No description

Owner Jason McCloud
Provisioned on September 7, 2018 3:32 PM
Project cmbu-auto-bg
Catalog item CentOS 7.3

Cost null
Lease duration Indefinite
Expires Never
Destroy date Never

Name: cmbuauto12
Component: centos73
Status: On (CreateSnapshot)
CPUs: 1
Memory (MB): 2048
Storage (GB): 50
Description: Provisioned by VMware vRA
Owner: jason@cmbu.local
Blueprint: CentOS 7.3
Compute resource: Cluster-1

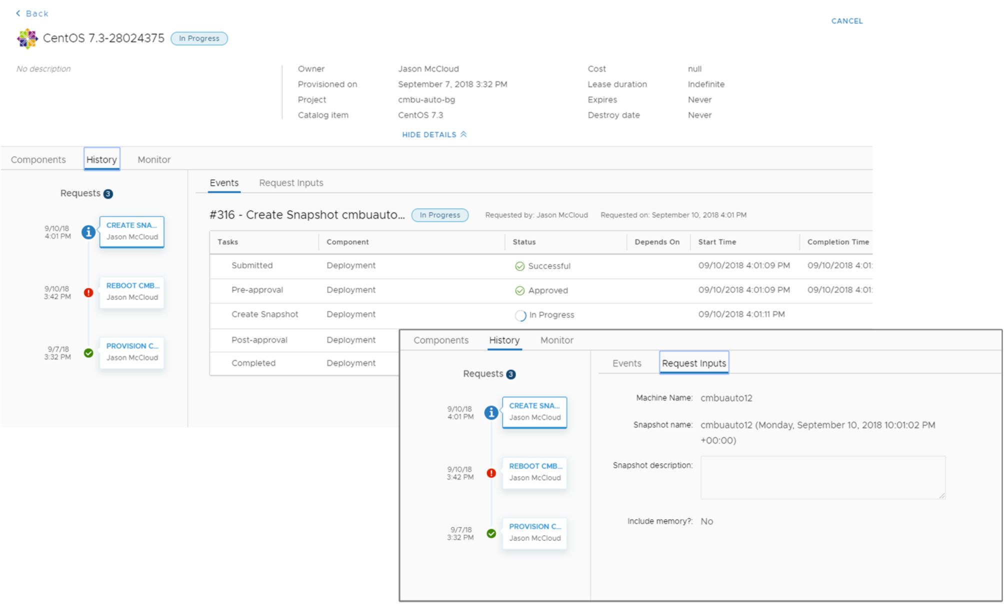
Run Component-Level Actions

Component-level actions are specific to the component. The available actions depend on how your business group is entitled to use them. If your administrator did not entitle you to run actions, you will not see the gear icon or the action list.

Deployment History

The History tab in the deployments details provides the full history of the deployment, from initial provisioning through any changes made using one or more actions. You can use the full provisioning history to learn when something changed and what values were provided.

Review the history details if you must determine when something changed or when you are investigating problems with the instance. You also use the history to troubleshoot failed deployments. See Testing and Troubleshooting Failed Provisioning Requests.



No description

Owner	Jason McCloud	Cost	null
Provisioned on	September 7, 2018 3:32 PM	Lease duration	Indefinite
Project	cmbu-auto-bg	Expires	Never
Catalog item	CentOS 7.3	Destroy date	Never

#316 - Create Snapshot cmbuauto... In Progress Requested by: Jason McCloud Requested on: September 10, 2018 4:01 PM

Tasks	Component	Status	Depends On	Start Time	Completion Time
Submitted	Deployment	✓ Successful		09/10/2018 4:01:09 PM	09/10/2018 4:01:
Pre-approval	Deployment	✓ Approved		09/10/2018 4:01:09 PM	09/10/2018 4:01:
Create Snapshot	Deployment	○ In Progress		09/10/2018 4:01:11 PM	
Post-approval	Deployment				
Completed	Deployment				

Machine Name: cmbuauto12

Snapshot name: cmbuauto12 (Monday, September 10, 2018 10:01:02 PM +00:00)

Snapshot description:

Include memory?: No

Deployment Monitoring Based on vRealize Operations Manager

vRealize Automation can show vRealize Operations Manager data about your deployments.

- Deployment-level alerts
- Machine-level metrics

Reviewing the filtered set of alerts and metrics directly in vRealize Automation saves you the task of accessing or searching vRealize Operations Manager. Although you cannot launch in context to vRealize Operations Manager, you are of course free to log in and use vRealize Operations Manager for additional data as needed.

Enable vRealize Operations Manager Data

For vRealize Automation to show vRealize Operations Manager data, you first configure settings and adapters.

Setup requires steps in both vRealize Operations Manager and vRealize Automation.

Prerequisites

Verify that you have vRealize Operations Manager version 6 or later.

Procedure

1 In vRealize Operations Manager, go to **Administration > Solutions**.

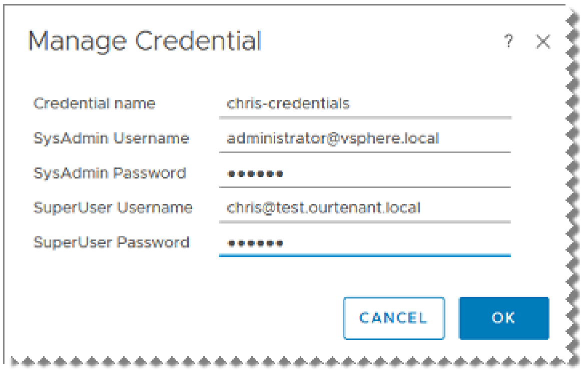
2 Under **Solutions**, verify that you have the vRealize Automation solution and that it is receiving data.

   a Select the vRealize Automation solution.

   b In the toolbar above the solutions, click the gears Configure icon.

c Under **Instance Settings**, go to **Credential**, and click the green plus to add credentials.

Credential Name	Description of this set of credentials
SysAdmin	Username and password of the vRealize Automation default tenant administrator, usually administrator@vsphere.local
SuperUser	Username and password of a high-access account for the vRealize Automation working tenant



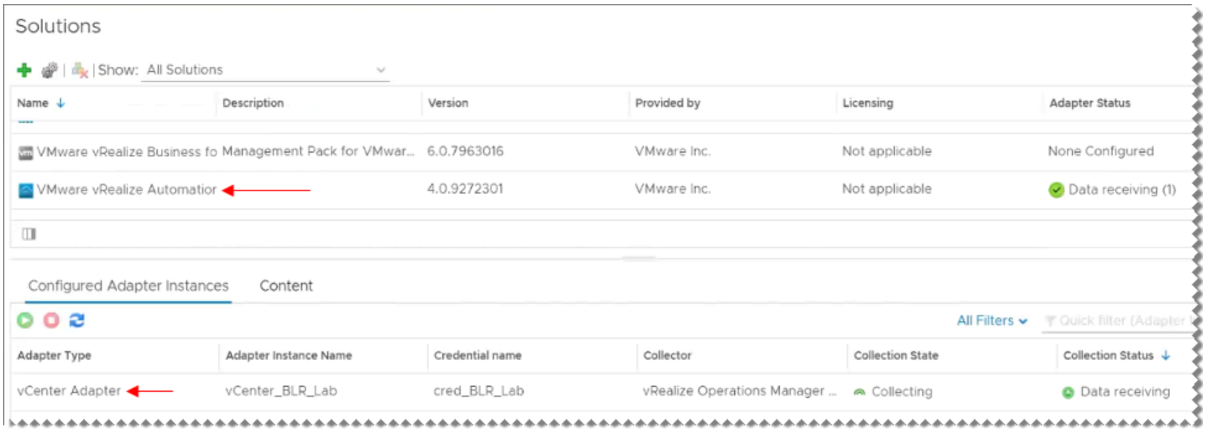
Manage Credential

Credential name: chris-credentials
SysAdmin Username: administrator@vsphere.local
SysAdmin Password: ••••••
SuperUser Username: chris@test.ourtenant.local
SuperUser Password: ••••••

d Save and test the credentials for proper connection.

3 Under **Configured Adapter Instances**, verify that you have a vCenter Adapter for the vSphere endpoint that vRealize Automation provisions to and that it is receiving data.

Figure 6-1. vRealize Operations Manager Solutions and Adapters



Solutions

Show: All Solutions

Name	Description	Version	Provided by	Licensing	Adapter Status
VMware vRealize Business fo	Management Pack for VMwar...	6.0.7963016	VMware Inc.	Not applicable	None Configured
VMware vRealize Automatior		4.0.9272301	VMware Inc.	Not applicable	✓ Data receiving (1)

Adapter Type	Adapter Instance Name	Credential name	Collector	Collection State	Collection Status
vCenter Adapter	vCenter_BLR_Lab	cred_BLR_Lab	vRealize Operations Manager ...	Collecting	Data receiving

4 In vRealize Operations Manager, go to **Alerts > Alert Settings**.

5 Verify that alert and symptom definitions will generate the vRealize Automation alerts that you want.

Most vRealize Automation users only need to ensure that a deployment stays healthy. Additional alerts from the virtual machine level can be overwhelming and contain details that cannot be managed using vRealize Automation.

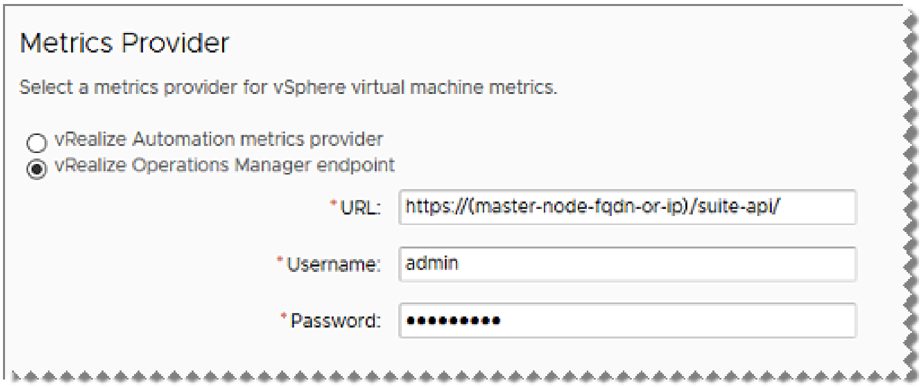
For vRealize Automation alerts, the overall deployment is the parent object. Virtual machines within the deployment are child objects. Alerting is at the deployment, parent level by default.

You are free to use vRealize Operations Manager to create deployment-level alerts that expose additional, specific symptoms. For example, you might want to show all SQL Server issues in a deployment.

6 In vRealize Automation, go to **Administration > Reclamation > Metrics Provider**.

7 Select **vRealize Operations Manager endpoint**.

8 Enter the vRealize Operations Manager URL https://*master-node-FQDN-or-IP*/suite-api/ and the username and password of an account with vRealize Operations Manager administrator rights.



Metrics Provider

Select a metrics provider for vSphere virtual machine metrics.

○ vRealize Automation metrics provider

● vRealize Operations Manager endpoint

*URL: https://(master-node-fqdn-or-ip)/suite-api/

*Username: admin

*Password: ●●●●●●●●●

Note When there is more than one authentication source, enter the username in user@domain@source format, where @source is the LDAP import source in vRealize Operations Manager. The user account requires a minimum of ReadOnly role, and Object rights to the vCenter Adapter and Cloud vCenter Server.

9 Test the connection and save it.

10 Click **Deployments**, select a deployment, and verify that the Monitor tab appears.

The Monitor tab only appears when vRealize Operations Manager is selected as metrics provider.

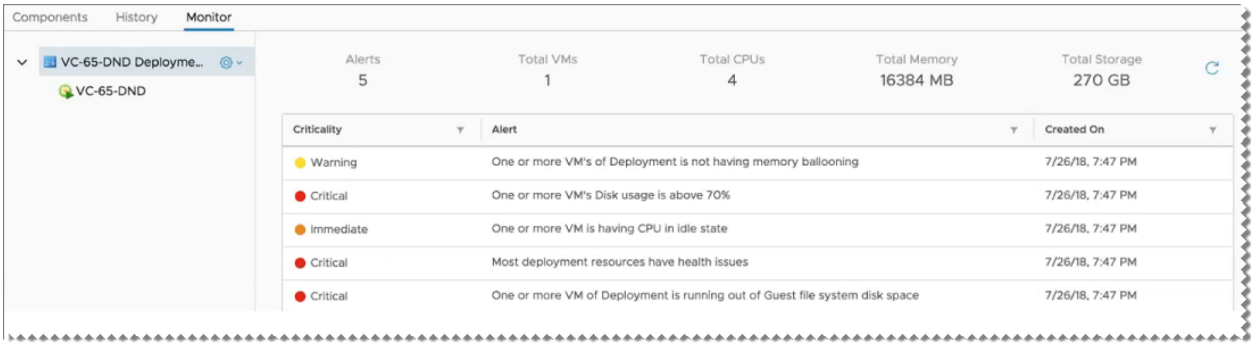
Alerts Provided by vRealize Operations Manager

When monitoring is enabled, vRealize Automation retrieves vRealize Operations Manager alerts about your deployments.

To access monitoring, click a deployment and select the **Monitor** tab. If the tab is missing, see Enable vRealize Operations Manager Data.

To see alerts, highlight the deployment name at the top of the component tree on the left.

- You can review the severity and text of the alerts.
- To focus on areas of concern, filter and sort on data in the columns.
- Only Health alerts appear. Other alert types such as Efficiency or Risk are not supported.



Alerts	Total VMs	Total CPUs	Total Memory	Total Storage
5	1	4	16384 MB	270 GB

Criticality	Alert	Created On
● Warning	One or more VM's of Deployment is not having memory ballooning	7/26/18, 7:47 PM
● Critical	One or more VM's Disk usage is above 70%	7/26/18, 7:47 PM
● Immediate	One or more VM is having CPU in idle state	7/26/18, 7:47 PM
● Critical	Most deployment resources have health issues	7/26/18, 7:47 PM
● Critical	One or more VM of Deployment is running out of Guest file system disk space	7/26/18, 7:47 PM

Metrics Provided by vRealize Operations Manager

When monitoring is enabled, vRealize Automation retrieves vRealize Operations Manager metrics about your deployments.

To access monitoring, click a deployment and select the **Monitor** tab. If the tab is missing, see Enable vRealize Operations Manager Data.

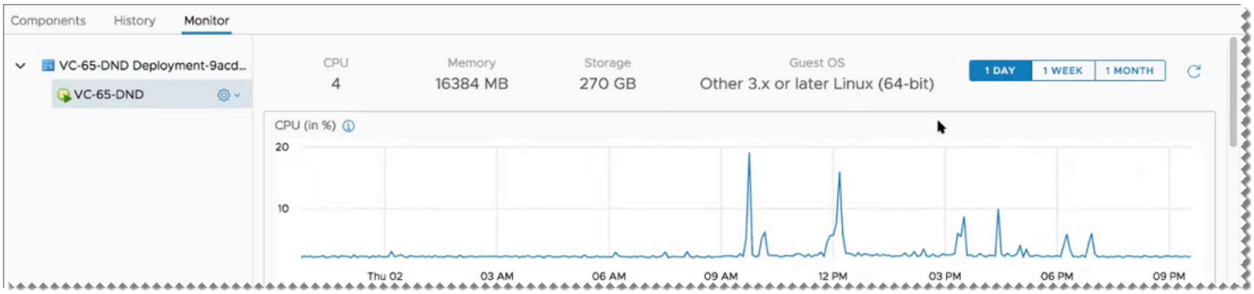
To see metrics, expand the component tree on the left, and highlight a virtual machine.

- Metrics are not cached. They come directly from vRealize Operations Manager and might take a few moments to load.
- Only virtual machine metrics appear. Metrics from other components such as vCloud Director, Software, or XaaS are not supported.
- Only vSphere virtual machine metrics appear. Other cloud providers such as AWS or Azure are not supported.

Metrics appear as timeline graphs that show highs and lows for the following measures.

- CPU
- Memory
- Storage IOPS
- Network MBPS

To reveal the specific metric name, click the blue information icon at the upper left corner of the timeline.



CPU	Memory	Storage	Guest OS
4	16384 MB	270 GB	Other 3.x or later Linux (64-bit)

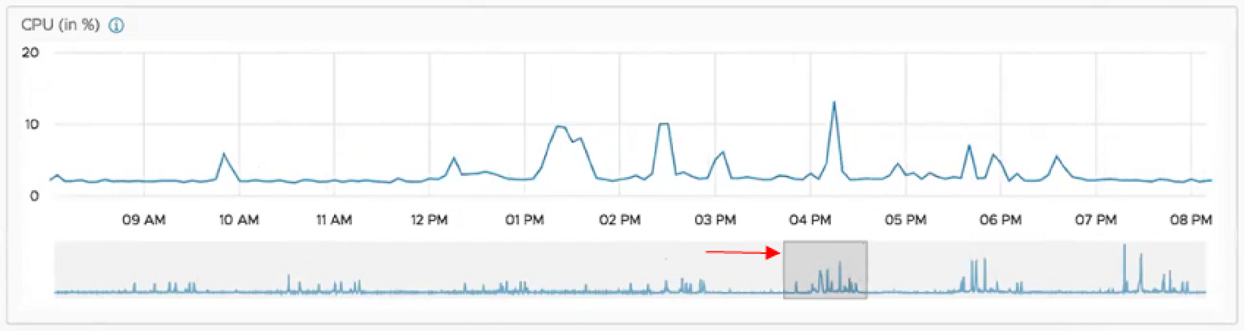
Acting on Data Provided by vRealize Operations Manager

When metrics provided by vRealize Operations Manager expose a problem, you can take some corrective actions directly in vRealize Automation.

To see metrics provided by vRealize Operations Manager, click a deployment and select the **Monitor** tab. If the tab is missing, see Enable vRealize Operations Manager Data.

Locating Problems

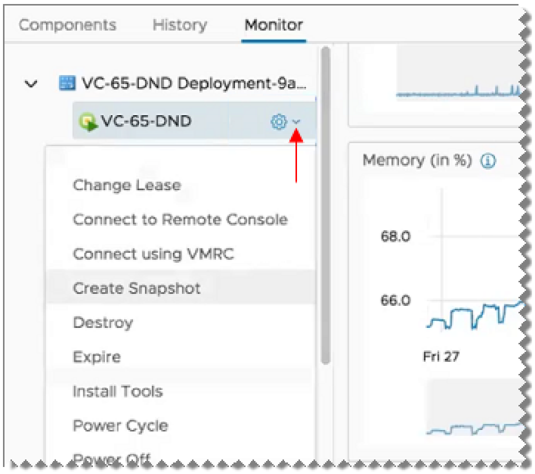
Metrics for the past day, week, or month are available. To zoom in on an area of concern, select a small area in the lower, shaded part under any metric timeline:



Making Changes

When a problem occurs, you can take some corrective actions directly in the same interface.

For example, if memory shows consistent usage spikes, you might decide to add memory. In the component tree on the left, click the drop-down for the virtual machine, and use the context menu options to perform maintenance or reconfigurations.



Running Actions on Deployed Resources

The actions that are available for a deployed resource depend on the type of resource, how the action was configured and made available for provisioned items, and the operational state of the item.

The configured actions that are available for a deployment or deployment component appear in the **Actions** menu for the selected deployment or component.

The list of available actions is determined by what your business group is entitled to run for the deployment and resource or machine type component. Whether an action is available depends on the machine type or state.

If the item was provisioned using an XaaS blueprint, the resource actions must be created, published, and entitled in the same service that is used to provision the item. The list of available actions is determined by the item type and the current state of the item.

The available actions for an item that was provisioned as an IaaS machine might also include XaaS resource actions if the actions are mapped to the item.

Action Menu Commands for Provisioned Resources

Actions are changes that you can make to provisioned resources. The vRealize Automation actions are used to manage the lifecycle of the resources.

The available commands on the **Action** menus depend on how your business group manager or tenant administrator configured the entitlement that contains the resource on which the actions run. The availability of a menu option also depends on the type of resource and the operational state of the item.

You can only run one action at a time. To run a second action on a resource, wait for the first to complete the requested change.

Table 6-2. Action Menu Commands

Action	Resource Type	Description
Associate Floating IP	Machine (OpenStack)	Associate a floating IP address with an OpenStack machine.
Cancel	Machine	Cancel a running reconfiguration action. Only actions that can roll back to a previous state can be cancelled by users. If an action does not support rolling back to a previous state, for example, Power Off, then only a user with tenant administrator privileges can cancel a the request.
Change Lease	Deployment and Machine	Change the number of days remaining in the lease for either a specific machine or for all resources included in a deployment. If you do not provide a value, the lease does not expire.
Change NAT Rules	NAT Network	Add new NAT port forwarding rules, reorder rules, edit existing rules, or delete rules.

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
Change Owner	Deployment	Change the owner of the deployment and all the included resources. Only Business group managers and support users can change the ownership of a deployment. The machine must be in the On, Off, or Active state when you initiate the change owner action or the action fails with the following message: `The action is invalid for the machine.`
Change Security	Deployment	You can add or remove existing NSX security groups and security tags. You can also remove on-demand security groups. For more information, see Add or Remove Security Items in a Deployment.
Connect using VMRC	Machine	Connect to the virtual machine using a VMRC 8.x application. To use this action, the VMRC application must be installed on the local system of the service catalog user who is running the action. For installation and user instructions, see VMware Remote Console Documentation. To download, see Download VMware Remote Console. The VMRC 8.x replaces the previous VMware Remote Console.
Connect to Remote Console	Machine	Connect to the selected machine using VMware Remote Console. The virtual machine console appears in the browser. The VMRC 8.x replaces the VMware Remote Console.
Connect using Console Ticket	Machine (OpenStack and KVM)	Connect to the OpenStack or KVM virtual machine using a console ticket for a VMware Remote Console connection.
Connect using ICA	Machine (Citrix)	Connect to the Citrix machine using the Independent Computing Architecture.
Connect using RDP	Machine	Connect to the machine by using Microsoft Remote Desktop Protocol.

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
Connect using SSH	Machine	Connect to the selected machine by using SSH. The **Connect Using SSH** option requires that your browser has a plug-in that supports SSH, for example the FireSSH SSH terminal client for Mozilla Firefox and Google Chrome. When the plug-in is present, selecting **Connect Using SSH** displays an SSH console and prompts for your administrator credentials. To use this action, the `Machine.SSH` custom property must be included and set to true in the blueprint's machine component in either a property group or individual custom property.
Connect using Virtual Desktop	Machine	Connect to the selected machine using Microsoft virtual desktop.
Create Snapshot	Virtual Machine	Create a snapshot of the virtual machine. If you are allowed only two snapshots and you already have them, this command is not available until you delete a snapshot.
Delete Snapshot	Virtual Machine	Delete a snapshot of the virtual machine.

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
Destroy	Deployment, Machine, and On-demand Security Group	Immediately destroy a provisioned resource. Except for XaaS, destroying components of a deployment is not a best practice. Use the scale in action to reduce the number of machines in your deployment, or destroy the entire deployment. You must run this action to destroy XaaS resources, even if they are part of a deployment you are destroying. Other resources are destroyed when their lease or their archival period ends. The Destroy action is not available for the following deployment situations: ■ physical machine deployments ■ deployments with an NSX existing network or NSX existing security resource ■ deployments with an NSX on-demand load balancer resource Because an NSX load balancer belongs to an NSX edge, when an NSX edge is destroyed, the load balancer resource is also destroyed and resources are released. When a machine tier that is load balanced is destroyed, it is removed from the load balancer pool on the respective NSX edge. **Note** The Destroy action might return a success message even if it cannot remove a machine deployment from its endpoint. For example, if a vSphere machine is on a non-vSAN datastore and its VMX file contains corrupted or otherwise invalid data. You can review the request log for additional information even if the Destroy message indicates that it was successful. Force-destroying a machine in this state might leave it running on the endpoint and cause IP conflicts. If the corruption is corrected on the endpoint (outside of vRealize Automation), you can retry the Destroy action. Business group administrators can forcibly destroy a deployment after a failed destroy request. Force destroy instructs vRealize Automation to ignore failures to destroy

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
		individual resources while destroying the deployment. For more information on using force destroy, see Force Destroy a Deployment After a Failed Destroy Request. **Note** Storage and memory that are assigned to a provisioned machine by a reservation are released when the machine to which they are assigned is deleted in vRealize Automation by the Destroy action. The storage and memory are not released if the machine is deleted in the vCenter Server. When destroying a deployment that contains an Amazon machine component, you can destroy more than one EBS volume at a time, depending on how the **Delete Volumes** setting was configured in the blueprint. For more information, see Amazon Machine Component Settings. When destroying a deployment that contains an Amazon machine component, all EBS volumes that were added to the machine during its life cycle are detached, rather than destroyed. vRealize Automation does not provide an option for destroying the EBS volumes.
Disassociate Floating IP	Machine (Openstack)	Remove the floating IP from the Openstack machine.
Dismiss	No resource type. Failed initial provisioning request or a failed action.	You dismiss a failed request. You cancel an in-progress request. ■ If the dismissed request is a deployment request, dismiss removes the failed deployment from your deployments list. ■ If the dismissed request is an action, dismiss removes the failed action request from the card, leaving the deployment in the previous state. You must dismiss a failed action request so that you can see run other actions on the associated deployment. You must also dismiss failed actions so that the deployment users can see the machine history. You cannot run dismiss on requests submitted by the API and it does not block actions submitted by the API. This action is available for all initial provisioning failed requests. It does not require entitlement.

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
Execute Reconfigure	Machine	Immediately reconfigure the machine or schedule the reconfiguration action for a later time.
Expire	Deployment and Machine	End the deployment or machine lease for all resources included in the deployment.
Export Certificate	Machine	Export the certificate from a Cloud machine.
Get Expiration Reminder	Machine	Downloads a calendar event file for the current lease expiration date.
Install Tools	Machine	Install VMware Tools on a vSphere virtual machine.
Power Cycle	Machine	Power off the machine, then power it back on.
Power Off	Machine	Power off the machine without shutting down the guest operating system.
Power On	Machine	Power on the machine. If the machine was suspended, normal operation resumes from the point at which the machine was suspended.
Reboot	Machine	Reboot the guest operating system on a vSphere virtual machine. VMware Tools must be installed on the machine to use this action.

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
Reconfigure	Machine	A business group manager, support user, or machine owner can perform the following reconfigure actions for the selected vSphere virtual machine: ■ Change description ■ Change CPU, memory, network, and disk settings ■ Add, edit, and delete custom properties and property groups ■ Add, edit, reorder, or delete a network adapter for NAT port forwarding rules ■ Reconfigure shutdown ■ Change machine owner (available for business group managers and support users only) You cannot change a storage reservation policy if doing so might change the storage profile on a disk. For more information, see Specify Machine Reconfiguration Settings and Considerations for Reconfiguration. If you selected the **Propagate updates to existing deployments** option on the **Blueprints Settings** page in the source blueprint, any increase or broadening in the CPU, Memory, or Storage minimum and maximum settings in the blueprint are pushed to active deployments that were provisioned from that blueprint. For more information, see Blueprint Properties Settings. Do not manage vRealize Automation-administered NSX objects outside of vRealize Automation. For example, if you modify the member port of a deployed load balancer in NSX, rather than in vRealize Automation, then NSX data collection breaks. Scale in and scale out operations also produce unexpected results.

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
Reconfigure	Load Balancer	An entitled machine owner, support user, tenant administrator, or business group manager can change any of the settings in a virtual server and can add or remove virtual servers in the NSX load balancer: For more information, see Reconfigure a Load Balancer in a Deployment. For information about virtual server settings in the load balancer, see Add an On-Demand Load Balancer Component. Do not manage vRealize Automation-administered NSX objects outside of vRealize Automation. For example, if you modify the member port of a deployed load balancer in NSX, rather than in vRealize Automation, then NSX data collection breaks. Scale in and scale out operations also produce unexpected results.
Register VDI	Virtual Machine (XenServer)	Register the virtual disk image on XenServer items.
Remove from Catalog	Deployments	Remove XaaS provisioned resources from catalog. You can perform this operation on existing objects and objects that are no longer in the Orchestrator inventory.
Reprovision	Machine	Destroys the machine, then initiates the provisioning workflow to create a machine with the same name. When you request machine reprovisioning, a known issue might cause vRealize Automation to display the reprovisioning status as Complete in the catalog, when the actual state is In Progress. After you submit a request to reprovision a machine, you can use any of the following sequences to check the status of the reprovisioned machine: ■ **Infrastructure > Managed Machines** ■ **Deployments** tab ■ **Administration > Events > Event Logs** **Note** You cannot reprovision an Amazon machine. For related information, see the VMware Knowledge Base article Reprovisioned machine tasks ... (2065873) at http://kb.vmware.com/kb/2065873.

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
Resubmit	No resource type. Failed initial provisioning request.	Resubmit a failed provisioning request. The resubmitted request starts at the beginning of the provisioning process with the already entered values. If a request fails and you can resolve the problem, you can resubmit the request rather than creating a new request. If the error is due to incorrect values, for example a datastore that does not support your request, you must create a new request with the new values. This action is available for all initial provisioning failed requests. It does not require entitlement.
Resume	Deployment	Resume partially successful provisioning request. Resume continues from the point of failure. If a deployment fails during the provisioning process due to temporary environmental or infrastructure problems, timeouts, or other issues that can be corrected outside of the request, you can resume the provisioning process rather than creating a new provisioning request. If errors in the blueprint caused the failure, resume does not work. You must request a new deployment rather than trying to resume. If a deployment request has only a partial success, and you can resolve the problem, you can use the resume action. The resumed request continues from the point of failure. For more information, see How the Resume Action Works.
Revert Snapshot	Virtual Machine	Revert to a previous snapshot of the machine. You must have an existing snapshot to use this action.

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
Scale In	Deployment	Destroys unneeded instances of machines in your deployment to adjust to reduced capacity requirements. Machine components and any software components installed on them are destroyed. Dependent software components and networking and security components are updated for the new deployment configuration. XaaS components are not scalable and are not updated during scale operations. You can try to repair partially successful scale operations by attempting to scale the deployment again. However, you cannot scale a deployment to its current size, and fixing a partially successful scale this way does not deallocate the dangling resources. You can view the request execution details screen and find out which tasks failed on which nodes to help you decide whether to fix the partially successful scale with another scale operation. Failed and partially successful scale operations do not impact the functionality of your original deployment, and you can continue to use your catalog items while you troubleshoot any failures.

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
Scale Out	Deployment	Provision additional instances of machines in your deployment to adjust to expanding capacity requirements. Machine components and any software components installed on them are provisioned. Dependent software components and networking and security components are updated for the new deployment configuration. XaaS components are not scalable and are not updated during scale operations. You can try to repair partially successful scale operations by attempting to scale the deployment again. However, you cannot scale a deployment to its current size, and fixing a partially successful scale this way does not deallocate the dangling resources. You can view the request execution details screen and find out which tasks failed on which nodes to help you decide whether to fix the partially successful scale with another scale operation. Failed and partially successful scale operations do not impact the functionality of your original deployment, and you can continue to use your catalog items while you troubleshoot any failures. If you selected the **Propagate updates to existing deployments** option on the **Blueprints Settings** page in the source blueprint, any increase in the CPU, Memory, or Storage minimum and maximum settings in the blueprint are pushed to active deployments that were provisioned from that blueprint. For more information, see Blueprint Properties Settings.
Shutdown	Machine	Shut down the guest operating system and power off the machine. VMware Tools must be installed on the machine to use this action.
Suspend	Machine	Pause the machine so that it cannot be used and does not consume any system resources other than the storage it is using.
Unregister	Machine	Remove the machine from the inventory without destroying it. Unregistered machines are not usable.

Table 6-2. Action Menu Commands (continued)

Action	Resource Type	Description
Unregister	Network	Remove the network from the inventory without destroying it. Unregistered networks are not usable.
Unregister VDI	Virtual Machine (XenServer)	Unregister the virtual disk image on XenServer items.

Troubleshooting Missing Actions in the Resource Actions Menu

As a machine or resource owner, you do not see all entitled actions for a provisioned item.

Problem

In an environment where you know that an action was entitled for your user or business group, you expect to see all actions when you select an item in your **Deployment** list.

Cause

The availability of actions depends on the type of provisioned resource, operational state of the resource, and how it was configured and made available. The following list provides some reasons why you do not see all configured actions.

- The action is not applicable based on the current state of the provisioned resource. For example, Power Off is available only when the machine is powered on.
- The action is not applicable to the selected item type. If the item does not support the action, it does not appear in the list. For example, the Create Snapshot action is not available for a physical machine, and the Connect by Using RDP action is not available if the selected item is a Linux machine.
- The action is applicable for the provisioned resource type, but the action is deactivated in the Infrastructure blueprint. If the action is deactivated, it never appears as an available action for any of the items that were provisioned using the blueprint.
- The action is not included in the entitlement used to provision the item on which you need to run the action. Only entitled actions, either as part of an IaaS blueprint or as an XaaS resource action, can appear in the Actions menu.
- The action is created as an XaaS resource action but was not included in the entitlement used to provision the item on which you need to run the action. Only entitled actions appear in the Actions menu.
- The action might be limited based on the configured target criteria for XaaS resource actions or resource mappings to provisioned IaaS machines.

Solution

- ◆ Verify that the action is applicable to the provisioned item or the state of the provisioned item.
- ◆ Verify that the action is configured and included in the entitlement used to provision the item.

Create a Snapshot of Your Machine

Depending on how your administrators have configured your environment, you might be able to create a snapshot of your virtual machine. A snapshot is an image of a virtual machine at a specific time. It is a space-efficient copy of the original VM image. Snapshots are an easy way to recover a system from damage, data loss, or security threats. After you create a snapshot of your virtual machine, you can apply it and reset your system back to the point where the snapshot was taken.

When you create a memory snapshot, the snapshot captures the state of the virtual machine power settings and, optionally, the virtual machine's memory. When you capture the virtual machine's memory state, the snapshot operation takes longer to complete. You might also see a momentary lapse in response over the network.

Prerequisites

- An existing virtual machine that is powered on, off, or suspended.
- If your virtual machine is configured for one or more independent disks, power off the machine before creating a snapshot. You cannot create a snapshot when it is powered on. For disk configuration information, see *Custom Properties V Table*.
- Your tenant administrator or business group manager entitled you to the snapshot action.

Procedure

1. Click **Deployments**.
2. Locate the deployment that includes the machine that you want to snapshot and click the deployment name.
3. On the **Components** tab, click the virtual machine and click actions gear icon.

   The component actions menu appears.
4. Click **Create Snapshot** in the Actions menu.
5. Enter a name and, optionally, a description.
6. If you want to capture the memory and power settings of the machine, select **Include memory**.
7. Click **Submit**.

Connect Remotely to a Machine

You can connect remotely to a machine from the vRealize Automation console.

If you are using VMware Remote Console to connect, see Knowledge Base article Troubleshooting VMRC connectivity in vRealize Automation (2114235).

Prerequisites

- Log in to vRealize Automation as a **machine owner**, **tenant administrator**, or **business group manager**.

- Verify that VMware Tools is installed.

  VMware Tools must be installed on your vRealize Automation client to support fully functioning access when connecting with VMware Remote Console. If VMware Tools is not installed, problems occur, such as the mouse pointer and mouse keys not working after connecting to the target machine. For information about supported VMware Tools versions, see the *vRealize Automation Support Matrix* in vRealize Automation product documentation.

- Verify that the provisioned machine is powered on.
- Allow network traffic between the vRealize Automation appliance(s) and the ESXi server over port 902.
- Allow network traffic between the vRealize Automation appliance(s) and the client browser over port 8444.
- Allow network traffic between the IaaS web component Windows server(s) and associated vSphere endpoint(s) over port 443.

Procedure

1. Click **Deployments**.
2. Locate the deployment that includes the machine to which you must connect and click the deployment name.
3. On the **Components** tab, locate the machine and click actions gear icon.

   The component actions menu appears.

4. Select the remote connection method.

   - Select **Connect Using RDP** to connect by using RDP.
   - Select **Connect to remote console** to connect by using VMware Remote Console.

   Respond to any prompts.

5. Click **Connect** and log in to the machine as directed.
6. When finished, log out and close the browser window.

Configuring Remote Consoles for vSphere with Untrusted SSL Certificates

If your vRealize Automation deployment uses untrusted certificates, before you can use remote consoles with VMware Remote Console, you must configure your client browser to trust the certificate, The steps to do this vary by browser.

If vRealize Automation is configured with a trusted SSL certificate for your environment, then VMware Remote Console does not require additional configuration on client browsers. When a vRealize Automation appliance certificate is replaced and is a trusted certificate, there is no need to update certificate information for the Web browser client.

If you want to replace the certificate, see the topic on replacing a vRealize Automation appliance certificate in the *System Administration* guide for vRealize Automation.

Remote connections using VMware Remote Console for machines provisioned on vSphere are secured by vRealize Automation appliance certificates through a proxy console. VMware Remote Console requires WebSockets support in the browser and browsers must trust the vRealize Automation appliance certificate. The certificate can be obtained by going to the root-level virtual appliance at an address of the form https://*vra-va.eng.mycompany.com/*.

For information about support requirements for browsers and vSphere, see the *vRealize Automation Support Matrix*.
Configure Firefox to Trust a Certificate for vRealize Automation
Untrusted vRealize Automation appliance certificates must be manually imported to client browsers to support VMware Remote Console on clients provisioned on vSphere.

For information about supported versions of Firefox, see the *VMware vRealize Support Matrix* in the vRealize Automation Information Center.

Note If vRealize Automation is configured with a trusted SSL certificate for your environment, then VMware Remote Console does not require additional configuration on client browsers.

Procedure

1 In a Firefox browser, log in to the vRealize Automation appliance.

A message appears saying that the certificate is not trusted.

2 Select **Open Menu > Options**.

3 Click **Privacy and Security**, and then click **View Certificates**.

4 In the Certificates Manager dialog box, click **Servers**, and then click **Add Exception**.

5 Add the URL for your vRealize Automation appliance with the 8444 port.

For example, `https://your-vra-fqdn-domain:8444`.

6 Click **Get Certificate**, and then click **Confirm Security Exception**.

7 Click **OK**.

Results

You can connect to the remote console without certificate errors.
Configure Internet Explorer to Trust a Certificate for vRealize Automation Appliance
Untrusted vRealize Automation appliance certificates must be manually imported to client browsers to support VMware Remote Console on clients provisioned on vSphere.

Note If vRealize Automation is configured with a trusted SSL certificate for your environment, then VMware Remote Console does not require additional configuration on client browsers.

The steps in this procedure apply for self-signed certificates and certificates issued by a Certificate Authority.

For information about supported versions of Internet Explorer, see the *VMware vRealize Support Matrix* on the VMware Web site.

Procedure

1. In an Internet Explorer browser, log in to the vRealize Automation appliance.
2. Click **View Certificate** on the certificate error message that appears in the browser address bar.
3. Click the **General** tab of the Certificate Information window..
4. Verify that the information about the certificate is correct and click **Install Certificate**.
5. Select **Place all certificates in the following store** in the Certificate Store dialog box.
6. Click **Browse** to locate the certificate store.
7. Select **Trusted Root Certification Authority** and click **OK**.
8. Click **Next** on the Certificate Store dialog box.
9. Click **Yes** in the Security Warning dialog box to install the certificate.
10. Restart the browser.

Results

You can connect to the remote console without certificate errors.
Configure Chrome to Trust a Certificate for vRealize Automation Appliance
Untrusted vRealize Automation appliance certificates must be manually imported to client browsers to support VMware Remote Console on clients provisioned on vSphere.

For information about supported versions of Chrome, see the *vRealize Automation Support Matrix* in vRealize Automation product documentation.

Note If vRealize Automation is configured with a trusted SSL certificate for your environment, then VMware Remote Console does not require additional configuration on client browsers.

On Windows, Chrome and Internet Explorer use the same certificate store. This means that certificates that are trusted by Internet Explorer are also trusted by Chrome. To establish trusted certificates for Chrome, import them through Internet Explorer. For information about this procedure, see Configure Internet Explorer to Trust a Certificate for vRealize Automation Appliance.

When you complete the procedure, restart Chrome.

To permanently trust a certificate on the Macintosh operating system, download the certificate file and install the certificate as trusted in your certificate management tool.

Procedure

1. In a Chrome browser, log in to the vRealize Automation appliance.
2. Click the *View site information* icon next to the browser address bar and click the **Certificate** icon to show the certificate information.
3. Save the certificate.

4 Start the Keychain Access application, which is typically located in the utilities folder of your applications folder.

5 Select **File > Import Items**.

6 On the Keychain Access screen, select the certificate file you saved earlier.

   Set the value of **Destination Key** to **System**.

7 Click **Open** to import the certificate.

8 Restart the browser.

Specify Machine Reconfiguration Settings and Considerations for Reconfiguration

vSphere, vCloud Air, and vCloud Director platforms support reconfiguration of existing machines in a deployment to modify specifications such as CPU, memory, and storage.

Reconfiguration requests are subject to approval based on entitlements, policies, and the actions enabled for the machine component in the blueprint.

Reconfiguring a virtual machine that is assigned to an on-demand network is not supported. You cannot reconfigure a NIC that is attached to an on-demand network. If you attempt to reconfigure an on-demand NAT or routed network, the error `Original network [<network>] is not selected in the machine's reservation.` is displayed, the networks on the machine remains intact, and IP addresses on the machine are unchanged.

If you are entitled to the Cancel Reconfigure (Machine) and Execute Reconfigure (Machine) actions, you can cancel a reconfiguration or retry a failed reconfiguration.

Expanding a disk on a VM that was provisioned from a linked clone blueprint is not supported.

You cannot reconfigure machines by using the `Size` or `Image` component profiles. The range of CPU, memory, and storage is calculated from the profile remains available for reconfigure actions. For example, use a small (1 CPU, 1024 MB memory, and 10 GB storage), medium (3 CPUs, 2048 MB memory, 12 GB storage) and large (5 CPUs, 3072 MB memory, 15 GB storage) `Size` value set. The available ranges during machine reconfiguration are 1-5 CPUs, 1024-3072 memory, and 1-15 GB storage.

vRealize Automation takes a blueprint snapshot at deployment. If you encounter reconfigure problems when updating machine properties such as CPU and RAM in a deployment, see Knowledge Base article 2150829 vRA 7.x Blueprint Snapshotting.

Prerequisites

- Log in to vRealize Automation as a **machine owner**, **support user**, **business group user with a shared access role**, or **business group manager**.
- The machine you want to reconfigure must have the status On or Off with no active reconfigure status.
- The machine type must be vSphere, vCloud Air, or vCloud Director although the NSX settings apply only to vSphere.

- Verify that you are entitled to reconfigure a machine.

Procedure

1. Click **Deployments**.
2. Locate the deployment that includes the machine that you must reconfigure and click the deployment name.
3. On the **Components** tab, click the virtual machine and click actions gear icon.

   The component actions menu appears.
4. Select **Reconfigure**.
5. Select the tab appropriate to the settings that you want to reconfigure.

   Table 6-3. Request Reconfiguration Changes

Tab	Topic
General	Reconfigure CPUs and Memory
Storage	Edit Storage Settings
Network	Change Network Settings To change NAT rules, see Change NAT Rules in a Deployment.
Security	To reconfigure security settings, see Add or Remove Security Items in a Deployment.
Properties	Change Custom Property and Property Group Settings

What to do next

Run the Requested Machine Reconfiguration .

Reconfigure CPUs and Memory

You can change the number of CPUs or the amount of memory and storage used by the provisioned machine, within the limits set by the provisioning blueprint.

For provisioned Amazon deployments, you can reconfigure all storage volumes in the deployment except for the root volume.

Expanding a disk on a VM that was provisioned from a linked clone blueprint is not supported.

Prerequisites

Specify Machine Reconfiguration Settings and Considerations for Reconfiguration.

Procedure

1. Click the **General** tab.
2. Enter the number of CPUs in the **# CPUs** text box.
3. Enter the amount of memory in the **Memory (MB)** text box.

4 Enter the amount of storage in the **Storage (GB)** text box.

What to do next

Specify additional machine reconfiguration settings. If you have finished changing machine settings, start the machine reconfiguration request. See Run the Requested Machine Reconfiguration .

Edit Storage Settings

You can add, delete, or change the size of a storage volume on a provisioned virtual machine.

You cannot reconfigure storage for the IDE disk type.

Storage and memory that are assigned to a provisioned machine by a reservation are released when the machine to which they are assigned is deleted in vRealize Automation by the Destroy action. The storage and memory are not released if the machine is deleted in the vCenter Server.

For example, you cannot delete a reservation that is associated with machines in an existing deployment. If you move or delete deployed machines manually in the vCenter Server, vRealize Automation continues to recognize the deployed machines as live and prevents you from deleting associated reservations.

You can change some settings, such as capacity and storage reservation policy, after machine provisioning and deployment.

The **Drive Letter / Mount Path** and **Label** values are applied to the guest agent at the time of provisioning. These values are not updated after provisioning and thus might not be current. To data-collect and display their current values, you can create and run a custom vRealize Orchestrator workflow.

Prerequisites

Specify Machine Reconfiguration Settings and Considerations for Reconfiguration.

For provisioned Amazon deployments, you can reconfigure all storage volumes in the deployment except for the root volume.

Procedure

1 Click the **Storage** tab.

2 View or edit storage options as needed.

   - If available, add a new volume.
   - If available, delete a volume.

     An unselectable icon indicates an undeletable volume, such as one from a linked clone.

   - If available, change a volume size.

     You cannot reduce the size of existing volumes. Volume size is limited by the total amount of storage specified in the blueprint, less the amount allocated to other volumes.

What to do next

Specify additional machine reconfiguration settings. If you have finished changing machine settings, start the machine reconfiguration request. See Run the Requested Machine Reconfiguration .

Change Network Settings

You can add, remove, or edit a network adapter.

You can change the following network settings during the machine reconfiguration process:

- Add or remove NICs.
- Allocate or release IP addresses for existing NICs.
- Assign new IP addresses to NICs, provided that the network is not an on-demand NAT or on-demand routed network.

  You cannot reconfigure an on-demand routed or on-demand NAT network.

  Network reconfiguration requires that the source and target networks be selected in the reservation.

When you add NICs, IP addresses are allocated. When you remove NICs, IP addresses are released.

When you change network settings based on reservation and network profile information, the new network IP is assigned in vRealize Automation but the deployed machine is not updated at the endpoint with the new IP information. You must manually assign the IP to the machine after the reconfiguration process is finished.

Reconfiguring a virtual machine that is assigned to an on-demand network is not supported. You cannot reconfigure a NIC that is attached to an on-demand network. If you attempt to reconfigure an on-demand NAT or routed network, the error `Original network [<network>] is not selected in the machine's reservation.` is displayed, the networks on the machine remains intact, and IP addresses on the machine are unchanged.

Changing NSX network settings is not supported for deployments that were upgraded or migrated from vRealize Automation 6.2.x to this vRealize Automation release.

Prerequisites

Specify Machine Reconfiguration Settings and Considerations for Reconfiguration.

Procedure

1. Click the **Network** tab.
2. (Optional) Add a network adapter.
   a. Click **New Network Adapter**.
   b. Select a network from the **Network Path** drop-down menu.

      All networks selected on the machine’s reservation are available.

   c Type a static IP address for the network in the **Address** text box.

   The IP address must be unallocated in the network profile assigned in the reservation.

   d Click the **Save** icon (✓).

3 (Optional) Remove a network adapter.

   a Locate the network adapter.

   b Click the **Delete** icon (🗑).

   You cannot remove network adapter 0.

4 (Optional) Edit a network adapter.

   a Locate the network adapter.

   b Click the **Edit** icon (✎).

   c Select a network from the **Network Path** drop-down menu.

   d Click the **Save** icon (✓).

What to do next

Specify additional machine reconfiguration settings. If you have finished changing machine settings, start the machine reconfiguration request. See Run the Requested Machine Reconfiguration .

Change Custom Property and Property Group Settings

You can edit, add, or delete custom properties in the deployed machine.

You cannot use custom properties to enter values for volume disk number, capacity, label, or storage reservation policy. You must enter these values by adding or editing a volume in the Storage volumes table. See Edit Storage Settings.

Prerequisites

Specify Machine Reconfiguration Settings and Considerations for Reconfiguration.

Procedure

1. Click the **Properties** tab.
2. To add a property, click **New Property**.
3. Enter the property name in the **Name** text box.
4. Enter the property value in the **Value** text box.
5. Select the **Encrypted** check box to encrypt the value.
6. Select the **Prompt user** check box to prompt users for the value when they request the machine.
7. Add another property, edit an existing property, or delete a property.

What to do next

Specify additional machine reconfiguration settings. If you have finished changing machine settings, start the machine reconfiguration request. See Run the Requested Machine Reconfiguration .

Run the Requested Machine Reconfiguration

You can start the requested machine reconfiguration immediately or schedule it to start at a particular day and time. You can also specify the power option for the machine before reconfiguring it.

Prerequisites

Specify Machine Reconfiguration Settings and Considerations for Reconfiguration.

Procedure

1 If the **Execution** tab is visible, you can select it to specify additional reconfiguration settings. If it is not visible, click **Submit** to start machine reconfiguration.

2 If the **Execution** tab is visible, click **Execution** to schedule the reconfiguration action.

3 (Optional) Select an option from the **Execute request** drop-down menu.

Option	Description
Immediate	Start reconfiguration as soon as possible after approval.
Scheduled	Start reconfiguration at the specified date and time. Enter the date and time in the text boxes that appear.

The scheduled time is the local time where the vRealize Automation web server is located. If **Execute request** is not available, reconfiguration starts immediately.

4 (Optional) Select a power action from the **Power action** drop-down menu.

Option	Description
Reboot if required	(Default) If required, restart the machine before reconfiguring it.
Reboot	Restart the machine before you reconfiguring it, regardless of whether a restart is required.
Do not reboot	Do not restart the machine before reconfiguring it, even if a restart is required.

The following conditions require that the machine be restarted before reconfiguration:

- CPU change where hot add is not supported or is deactivated
- Memory change where hot memory is not supported or is deactivated
- Storage change where hot storage is deactivated

If the machine is in the shutdown state, it is not restarted.

Note You can deactivate the vSphere hot add option by using the `VirtualMachine.Reconfigure.DisableHotCpu` custom property.

5 Click **OK**.

What to do next

You can monitor the progress of the reconfiguration by observing the workflow states shown in the user interface. See Workflow States of Reconfigure Operations.

Workflow States of Reconfigure Operations
When reconfiguration starts and as it progresses through the workflow, you can monitor the progress from the Edit page.

Table 6-4. Workflow States of Reconfigure Operations

State	Description
Reconfigure pending	The state operation was created.
Scheduled	A scheduled workflow has been created for the Distributed Execution Manager (DEM).
Reconfiguring	The interface-specific workflow is being run.
Reconfigure failed, waiting to retry	The reconfigure failed, waiting for the owner to request a retry. If the machine owner is entitled to the reconfigure or cancel reconfigure actions, the owner can retry or cancel a reconfiguration.
ReconfigureFailed	The reconfigure failed, waiting for the workflow to perform the next action.
ReconfigureSuccessful	The reconfigure was successful, waiting for the workflow to perform the next action.
Canceled	The user has canceled the reconfiguration. Machine owners who are entitled can cancel a reconfiguration.
Complete	The completion workflow sets this state after completing the cleanup, so that the workflow can proceed to clean up the state operations and approvals. A status of complete indicates that the request from vRealize Automation is finished, but it does not indicate that the machine reconfiguration completed successfully.

Reconfigure a Load Balancer in a Deployment

You can add, edit, or delete a virtual server in a deployed NSX load balancer.

The following considerations apply to deployments that originated in vRealize Automation 7.2 or earlier:

- Load balancer reconfiguration is limited to deployments that contain a single load balancer.
- The Items detail page for any load balancer in a deployment displays the virtual servers that are used by all the load balancers in the deployment. For more information, see Knowledge Base Article 2150276.

- The Reconfigure Load Balancer operation is not supported for deployments that were upgraded or migrated from vRealize Automation 6.2.x to this vRealize Automation release.

For upgraded load balancers and load balancers deployed in the current vRealize Automation release, do not edit a virtual server and add a virtual server in the same request. For more information, see Knowledge Base Article 2150240.

Note The **Reconfigure** action is not supported for NSX-T load balancers.

If you submit a request to reconfigure a load balancer while another action is being performed on the deployment, for example when a scale out operation on the deployment is in progress, reconfiguration fails with a supporting message. In this situation, you can wait until the action is finished and then submit the reconfiguration request.

Note If the blueprint associated with the deployment is imported from a YAML file that contains an on-demand load balancer with a value in the name field that is different from the value in the ID field, the **Reconfigure** action fails. To enable the load balancer reconfiguration option for a deployment that is based on an imported blueprint, perform the following steps in the blueprint to allow post-provisioning actions for load balancer components in future deployments.

1. In the vRealize Automation consol, select the blueprint.
2. Click **Edit** and change the blueprint name. This sets the name and embedded ID to the same value.
3. Select the load balancer component in the blueprint.
4. Click **Edit** and re-enter the component name. This sets the name and embedded ID to the same value.
5. Repeat for all load balancer components in the blueprint.
6. Save the blueprint.

When you provision a new deployment using the edited blueprint, the reconfigure load balancer action works. To avoid this issue, ensure that all YAML files have identical name and ID values for all load balancer, network, and security components prior to importing them.

Do not manage vRealize Automation-administered NSX objects outside of vRealize Automation. For example, if you modify the member port of a deployed load balancer in NSX, rather than in vRealize Automation, then NSX data collection breaks. Scale in and scale out operations also produce unexpected results.

For information about the settings that are available when you add or edit a virtual server, see Add an On-Demand Load Balancer Component.

When you reconfigure a load balancer in vRealize Automation, some of the settings that were configured in NSX and that are not available as settings in vRealize Automation, are reverted back to their default value. After you run the load balancer reconfigure action in vRealize Automation, verify and update as needed the following settings in NSX:

- Insert-X-Forwarded for HTTP Header

- HTTP Redirect URL
- Service Monitor Extension

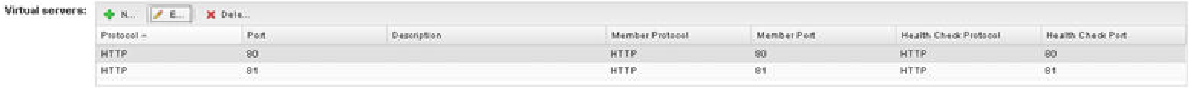
Prerequisites

- Log in to vRealize Automation as a **machine owner**, **support user**, **business group user with a shared access role**, or **business group manager**.
- Verify that you are entitled to reconfigure load balancers in a deployment. The required catalog entitlement is Reconfigure (Load Balancer).

Procedure

1. Click **Deployments**.
2. Locate the deployment that includes the load balancer that you must reconfigure and click the deployment name
3. On the **Components** tab, click the load balancer and click actions gear icon.

   The component actions menu appears.
4. Select **Reconfigure**.
5. Add, edit, or remove virtual servers.

Protocol ▲	Port	Description	Member Protocol	Member Port	Health Check Protocol	Health Check Port
HTTP	80		HTTP	80	HTTP	80
HTTP	81		HTTP	81	HTTP	81

6. Click **Submit**.

Change NAT Rules in a Deployment

You can add, edit, and delete existing NSX NAT rules in a deployed NAT one-to-many network.

You can also change the order in which the NAT rules are processed.

Note If the deployment's source blueprint is imported from a YAML file that contains a NAT network component, and the NAT network component's name and ID values are not identical, the **Change NAT Rules** action fails. To allow the **Change NAT Rules** action for a deployment that is based on an imported blueprint, perform the following steps in the blueprint before you provision a deployment.

1. Start vRealize Automation, click the Design tab, and open the blueprint.
2. Click **Edit** and change the blueprint name. This sets the name and embedded ID to the same value.
3. Select the NAT network component in the blueprint.
4. Click **Edit** and re-enter the component name. This sets the name and embedded ID to the same value.
5. Repeat for all NAT network components in the blueprint.
6. Save the blueprint.

To avoid this issue, ensure that all YAML files have identical name and ID values for all blueprints and load balancer, network, and security components prior to importing them.

For related information, see Creating and Using NAT Rules for NSX for vSphere and Add an On-Demand NAT or On-Demand Routed Network Component in vRealize Automation.

Prerequisites

- Log in to vRealize Automation as a **machine owner**, **support user**, **business group user with a shared access role**, or **business group manager**.
- Verify that you are entitled to change NAT rules in a network.
- Verify that the NAT network is configured as a NAT one-to-many network. The action is not available for NAT one-to-one networks.

  NSX for vSphere supports NAT one-to-one and NAT one-to-many networks, but NSX-T only supports NAT one-to-many.

Procedure

1. Click **Deployments**.
2. Locate the deployment that includes the network component that you must change and click the deployment name
3. On the **Components** tab, click the NAT network component.

   For an on-demand NAT network associated with a third-party IPAM provider, you cannot edit the component. However, you can manually add a new a destination IP address. When you add a new destination IP address, the component value is nulled. The new destination IP address and the null machine ID are processed when you submit the reconfiguration request.

4 Click the actions gear icon.

   The component actions menu appears.

5 Click **Change NAT Rules**.

6 Add new NAT port forwarding rules, reorder rules, edit existing rules, or delete rules.

7 Click **Submit**.

Display All NAT Rules for an Existing NSX Edge

You can display NAT rule information about the NSX Edges that are used in active deployments.

The NAT rules are displayed in the Edge view as an aggregate of all the NAT rules that are used in the deployment. In the Edge view, the rules are not necessarily displayed in the order in which they are processed.

To see and optionally change the order in which the NAT rules are processed in a NAT one-to-many network, see Change NAT Rules in a Deployment.

Prerequisites

- Log in to vRealize Automation as a **machine owner**, **support user**, **business group user with a shared access role**, or **business group manager**.

Procedure

1 Click **Deployments**.

2 Locate the deployment that includes the NSX Edge that your are viewing and click the deployment name.

3 On the **Components** tab, locate the NSX Edge component.

4 Select the NSX Edge that you want to view.

5 Click **Close** when you are finished.

Add or Remove Security Items in a Deployment

You can add or remove existing NSX security groups and security tags in a machine deployment. You cannot add on-demand security groups but you can remove them.

The change security action is based on a machine component or cluster. For example, if security is associated to a cluster named AppTier2 which consists of 2 machines, you perform the change security operation on the AppTier2 cluster, not the individual machines within the cluster.

The Change Security operation is not supported for deployments that were upgraded or migrated from vRealize Automation 6.2.x to this vRealize Automation release.

Prerequisites

- Log in to vRealize Automation as a **machine owner**, **support user**, **business group user with a shared access role**, or **business group manager**.

- Verify that you are entitled to change security in a deployment. The required catalog entitlement is Change Security (Deployment).

Procedure

1. Click **Deployments**.
2. Locate the deployment that includes the security groups and tags and click the deployment name.
3. On the **Components** tab, click the security component and click actions gear icon.

   The component actions menu appears.
4. Click **Change Security**.
5. Select the deployed machine component or cluster in which to add or remove security items.
6. Add or remove existing security groups and security tags for each machine component or cluster in the deployment as required.
7. Remove on-demand security groups for each machine component or cluster in the deployment as required.
8. (Optional) Click the **Reason** tab and enter a reason for the request.
9. Click **Submit**.

Additional Deployment Management Methods

Deployed resources can be managed using entitled actions, buy there are additional methods that are not included as actions.

These methods are not available on the Deployments tab, but you use them to make changes to provisioned resources.

Reclaiming Resources Based on vRealize Operations Manager Metrics

Reclamation helps you use your resources efficiently. If you also use vRealize Operations Manager to manage resources in your environment, you can configure vRealize Automation to use the metrics to calculate where you can reclaim deployment resources.

Procedure

1. Configure a Metrics Provider

   You can configure vRealize Automation to use vRealize Operations Manager health and resource metrics for vSphere virtual machines.
2. Send Reclamation Requests

   You can view and manage deployments and send reclamation requests to deployment owners. A reclamation request specifies a new lease length in days, the amount of time given for a deployment owner’s response, and which machines to target for reclamation.

3 Track Reclamation Requests

You can track the current state of reclamation requests and other details.

Configure a Metrics Provider

You can configure vRealize Automation to use vRealize Operations Manager health and resource metrics for vSphere virtual machines.

For more information about vRealize Operations Manager health badges and metrics, see the vRealize Operations Manager documentation.

Prerequisites

- Log in to the vRealize Automation console as a **tenant administrator**, **business group manager**, or **machine owner**.

  Reclamations—Users who create reclamation requests need the tenant administrator role, and the same tenant administrator account must be a member of at least one business group in the tenant.

  Failure to add the tenant administrator account to a business group causes a system exception when opening the **Reclamation > Deployments** tab.
- Create a vRealize Operations Manager user account with view and resource metrics query privileges for all vSphere servers that you integrate with vRealize Automation.
- Create vRealize Operations Manager adapter instances for all vSphere servers you add as endpoints in vRealize Automation. For information about creating adapter instances, see the vRealize Operations Manager documentation.

Procedure

1 Select **Administration > Reclamation > Metrics Provider**.

2 Select a metrics provider.

Option	Description
(Default) vRealize Automation metrics provider	If you do not have a vRealize Operations Manager instance, vRealize Automation provides basic machine metrics.
vRealize Operations Manager endpoint	Provide connection information for the vRealize Operations Manager instance you want to use as your metrics provider for vSphere virtual machines.

3 Click **Test Connection**.

4 Click **Save**.

Results

Tenant administrators, machine owners, and business group managers of the group in which the machine resides can view health badges and health alerts on the item details pages for vSphere virtual machines. They can also view vRealize Operations Manager metrics and health badges when they filter by the platform type vSphere on the reclamations page.

What to do next

Send Reclamation Requests.

Send Reclamation Requests

You can view and manage deployments and send reclamation requests to deployment owners. A reclamation request specifies a new lease length in days, the amount of time given for a deployment owner’s response, and which machines to target for reclamation.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.
- (Optional) To see health badges or view metrics provided by vRealize Operations Manager, see Configure a Metrics Provider.

Procedure

1 Select **Administration > Reclamation > Deployments**.

2 Find virtual machine deployments that match your search criteria.

You must select platform type vSphere to view metrics provided by vRealize Operations Manager.

a Click the **Advanced Search** down arrow to open the search box.

b Enter or select one or more search values.

Option	Action
Virtual Machine name contains	Enter one or more characters in the text box to find virtual machine names that match.
Owner name contains	Enter a name in the text box to find owner names that match.
Business group names contains	Enter a name in the text box to find business group names that match.
Platform Type	Select a platform type from the drop-down menu. Select vSphere to view metrics provided by vRealize Operations Manager. Required for vRealize Operations Manager.
Power State	Select a power state value from the drop-down menu to find virtual machines with a matching power state.
Expiration date between	Click the calendar icons and select start and end dates to find expiration dates inside the range.
CPU usage	Select a value from the drop-down menu to find virtual machines with High CPU use, above 80%, Low CPU use below 5%, or None, no value. If you are querying vRealize Operations Manager metrics, you cannot use this filter to query, and you cannot sort results by CPU usage.
Memory usage	Select a value from the drop-down menu to find virtual machines with High Memory use, above 80%, Low Memory use, below 10%, or None, no value. If you are querying vRealize Operations Manager metrics, you cannot use this filter to query, and you cannot sort results by memory usage.
Disk usage	Select a value from the drop-down menu to find virtual machines with Low Hard Disk use, less than 2 KBs per second or None, no value. If you are querying vRealize Operations Manager metrics, you cannot use this filter to query, and you cannot sort results by disk usage.
Network usage	Select a value from the drop-down menu to find virtual machines with Low Network use, less than 1 KB per second, or None, no value. If you are querying vRealize Operations Manager metrics, you cannot use this filter to query, and you cannot sort results by network usage.
Complex metric	Select a value from the drop-down menu to find virtual machines based on complex metrics. For example, select idle to find machines that have CPU, network, memory, and disk usage values all under 20%. You cannot use this filter if you are querying vRealize Operations Manager metrics.

c Click the search icon (🔍).

3. From the Deployments page, select one or more machines whose parent deployment is to be reclaimed.

   Only selected machines that are visible on the current results page are reclaimed.

4. Click **Reclaim**.

   The deployments that contain virtual machines that are selected on the current page are included in the request.

   Note The Reclaim Deployment page can list machines that are not available for reclamation, such as machines for which the lease has expired. If you specify a machine that is not available for reclamation, you receive the following error:

   ```
   Selection Error: Virtual machine name is not in valid state for reclamation.
   ```

5. Enter the duration of the new lease in the **New lease length (days)** text box.

   The minimum is 1 day, the maximum is 365 days, and the default is 7 days.

6. Enter how many days the deployment owner has to respond to the reclamation request in the **Wait before forcing lease (days)** text box

   At the end of that time, the deployment gets a new lease with the new lease length. The minimum waiting period is 1 day, the maximum is 365 days, and the default is 3 days.

7. Enter a reason for the request in the **Reason for request** text box.

8. Click **Submit**.

9. Click **OK**.

Results

When you send a reclamation request, it appears in the Inbox of the deployment owner. If the owner does not respond to the request in the required number of days, the deployment gets a new lease of the specified length, unless its current lease is shorter. If the owner clicks **Item in Use** on the reclamation request, the deployment's lease remains unchanged. If the owner clicks **Release for Reclamation**, the deployment lease expires immediately.

What to do next

Track Reclamation Requests.

Track Reclamation Requests

You can track the current state of reclamation requests and other details.

The following alternative methods are available for checking a recent reclamation request:

- Click the **Inbox** tab and select **Reclamation Requests** to view reclamation request information.
- Click the **Reclamation Requests** tab and view the list of recent requests
- Click the **Deployments** to view recent deployment changes.

Prerequisites

Log in to vRealize Automation as a **tenant administrator**.

Procedure

1. Select **Administration > Reclamation > Reclamation Requests**.
2. Find the virtual machines that match your search criteria.
   a. Click the **Advanced Search** down arrow to open the search box.
   b. Type or select one or more search values.

Option	Action
Virtual Machine name contains	Type one or more characters in the text box to find virtual machine names that match.
Owner name contains	Type one or more characters in the text box to find owner names that match.
Request Reason contains	Type one or more characters in the text box to find a request reason that matches.
Request State	Select a request state value from the drop-down menu to find virtual machines with a matching request state.

   c. Click the **Search** icon (🔍) or press Enter to start the search.
   d. Click the **Advanced Search** up arrow to close the search box.
3. (Optional) Click **Refresh Data** to update the display of reclamation requests.

Change the Reservation of a Managed Machine

You can change the reservation or storage setting for a managed machine. This ability is useful when a machine moves to a new storage path that is not available in its current reservation. For a single machine deployment, you can also change the business group for the machine.

You can move a machine in a single machine deployment to a different business group if the machine owner is a member of the target business group. You must be a business group manager of the original and the target business group to change the business group setting.

Note If there is a reservation policy assigned to the machine, you cannot change its business group.

You can create additional reservations for the associated compute resource by using the **Administration > Compute Resource** menu options.

Storage and memory that are assigned to a provisioned machine by a reservation are released when the machine to which they are assigned is deleted in vRealize Automation by the Destroy action. The storage and memory are not released if the machine is deleted in the vCenter Server.

For example, you cannot delete a reservation that is associated with machines in an existing deployment. If you move or delete deployed machines manually in the vCenter Server, vRealize Automation continues to recognize the deployed machines as live and prevents you from deleting associated reservations.

If changing the reservation will move a machine in vCenter Server to a new storage path that is not part of that machine's reservation in vRealize Automation, verify that the target or new storage path is selected in the machine's target reservation before you change the machine's reservation.

Prerequisites

Log in to vRealize Automation as a **fabric administrator**.

Procedure

1 Select **Infrastructure > Managed Machines**.

2 Locate the machine with the reservation to change.

3 Click **Change Reservation** in the drop-down menu.

   You can view information about the managed machine, such as its associated blueprint and compute resource, by clicking **View** in the drop-down menu.

4 (Optional) Select a business group from the **Business group** drop-down menu.

5 (Optional) Select a reservation from the **Reservation** drop-down menu.

6 (Optional) Select a storage policy from **Storage** drop-down menu.

7 Click **OK**.

Working with the Inbox

The Inbox provides in-product notifications regarding catalog request approvals, interactions requested during the provisioning process, and the status of reclamations requests based on any vRealize Operations Manager metrics.

You can review each tab to see whether you have any pending notifications that require action.

- **Approvals.** You can track your catalog requests that require approval. If you are designated as an approver on a catalog request, you can respond to an approval request. See Add Level Information to Approval Policy Settings .
- **Manual User Action.** Some catalog requests require interaction during the provisioning process. You can respond to the interaction request. See vRealize Orchestrator Integration in vRealize Automation.
- **Reclamation Requests.** If you use vRealize Operations Manager to determine where you can reclaim resources, you can track the reclamation requests. See Track Reclamation Requests.

Life Cycle Extensibility

7

Using vRealize Orchestrator with vRealize Automation, you can extend how you the manage the life cycle of IaaS machines.

Extending vRealize Automation requires you to use provided vRealize Orchestrator workflows and to create custom workflows.

This chapter includes the following topics:

Machine Extensibility Overview

Provisioning or decommissioning a new machine, especially for mission-critical systems, typically requires interacting with a number of different management systems, including DNS servers, load balancers, CMDBs, IP Address Management and other systems.

Machine Life Cycle Extensibility

You can inject custom logic at various predetermined IaaS life cycle stages by leveraging IaaS state change workflows, known as workflow stubs. You can use the workflow stubs to call out to vRealize Orchestrator for bi-directional integration with external management systems.

Creating a state change workflow enables you to trigger the execution of a workflow before the IaaS main workflow enters a specific state. For example, you can create custom workflows to integrate with an external database and record information at different stages of the machine life cycle.

- Create a custom workflow that runs before the main workflow enters the MachineProvisioned state to record such information as machine owner, approvers, and so on.
- Create a custom workflow that runs before a machine enters the MachineDisposing state to record the time at which the machine was destroyed and data such as its resource utilization at last data collection, last logon, and so on.

The main workflow illustrations show the main states of the primary workflow, highlighting in yellow the states you can customize by using IaaS workflow stubs. The **Customizable State Change Workflows** table lists the workflow stubs available, their corresponding place in the main workflow state, and examples of custom logic you could use at each state to extend the machine life cycle.

Figure 7-1. Main workflow states for provisioning machines

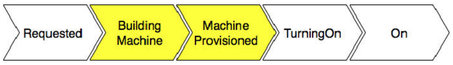


Figure 7-2. Main workflow states for importing machines

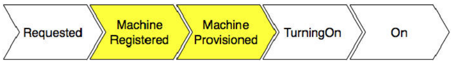


Figure 7-3. Main workflow states for machine lease expiration

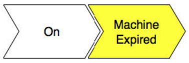


Figure 7-4. Main workflow states for disposing a machine

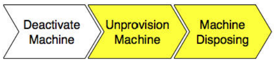


Table 7-1. Customizable State Change Workflows

Main Workflow State	Customizable Workflow Name	Extensibility Examples
BuildingMachine	WFStubBuildingMachine	Prepare for the machine to be created on the hypervisor.Create a configuration management database (CMDB) record, call out to an external system to assign an IP address to a machine, and then during machine disposal, use WFStubMachineDisposing to return the IP address to the pool.
RegisterMachine	WFStubMachineRegistered	Add an imported machine to an application provisioning tool to receive updates and undergo compliance checks.

Table 7-1. Customizable State Change Workflows (continued)

Main Workflow State	Customizable Workflow Name	Extensibility Examples
MachineProvisioned	WFStubMachineProvisioned	The machine exists on the hypervisor, and any additional customizations are completed at this point, for example guest agent customizations. Use this workflow stub to update a configuration management database (CMDB) record with DCHP IP address and storage information. Customizations made by using the WFStubMachineProvisioned are typically reversed by using WFStubUnprovisionMachine.
Expired	WFStubMachineExpired	Move an expired machine to low cost storage to reduce archival costs and update the CMDB record and billing system to reflect storage and cost changes.
UnprovisionMachine	WFStubUnprovisionMachine	Remove machines from active directory accounts. Customizations made by using the WFStubMachineProvisioned are typically reversed by using WFStubUnprovisionMachine.
Disposing	WFStubMachineDisposing	Return IP addresses to the pool.

Choosing a Life Cycle Extensibility Scenario

You can use vRealize Orchestrator or vRealize Automation Designer to extend machine lifecycles.

You can extend machine lifecycles by using vRealize Automation Designer to call out to vRealize Orchestrator, or by using vRealize Orchestrator directly. Both approaches allow you to add custom logic to predetermined stages of the IaaS machine lifecycle by creating custom vRealize Orchestrator workflows and then insert the custom workflows into the state change workflow stubs. However, you can only restrict custom state change logics to particular blueprints if you are using vRealize Orchestrator directly, and you can only restrict the running of workflows to specific Distributed Execution Managers (DEMs) by vRealize Automation Designer.

Note The workflow stubs are replaced by the event broker workflow subscriptions. They are still available, supported, and they can be used, but expect them to be removed in a future version of vRealize Automation. To ensure future product compatibility, you should use the workflow subscriptions to run custom workflows based on state changes. See Configuring Workflow Subscriptions to Extend vRealize Automation.

Table 7-2. Choosing a Lifecycle Extensibility Scenario

Scenario	Procedure
Add custom logic to predetermined stages of the IaaS machine lifecycle and apply that custom logic to specific blueprints.	Extending Machine Lifecycles by Using vRealize Orchestrator Checklist
Add custom logic to predetermined stages of the IaaS machine lifecycle and apply that custom logic globally to all of your blueprints.	Extending Machine Life Cycles By Using vRealize Automation Designer Checklist
Restrict the running of workflows to specific Distributed Execution Managers by using skills in vRealize Automation Designer. Skills are similar to a tag that you can apply to both workflows and DEM Worker instances. For example, you might want to restrict cloud provisioning workflows to a specific DEM running on a host with the required network access to Amazon URLs.	Associate Workflows and DEM Workers by Using Skills

Extending Machine Lifecycles By Using vRealize Orchestrator

You can inject custom logic into predetermined stages of the IaaS machine lifecycle by creating custom vRealize Orchestrator workflows and then using vRealize Orchestrator to insert the custom workflows into the lifecycle of machines built from specific blueprints.

Extending Machine Lifecycles by Using vRealize Orchestrator Checklist

The extending machine lifecycles by using vRealize Orchestrator checklist provides a high-level overview of the steps required to install and configure vRealize Orchestrator to customize IaaS machine lifecycles.

Table 7-3. Extending Machine Lifecycles by Using vRealize Orchestrator Checklist

Task	Details
❑ Configure a vRealize Automation host for your vRealize Orchestrator.	Add a vRealize Automation Host
❑ Configure an IaaS host for your vRealize Orchestrator.	Add an IaaS Host
❑ Install the vRealize Orchestrator customizations for extending IaaS machine lifecycles.	Install vRealize Orchestrator Customization
❑ Create a vRealize Automation endpoint for your vRealize Orchestrator instance.	Create a vRealize Orchestrator Endpoint

Table 7-3. Extending Machine Lifecycles by Using vRealize Orchestrator Checklist (continued)

Task	Details
❑ Use the workflow template provided in the Extensibility subdirectory of the vRealize Automation plug-in library to create a custom vRealize Orchestrator workflow to run during the machine lifecycle. You can run multiple workflows in the same state for the same blueprint as long as you nest them under a single wrapper workflow.	For information about developing workflows with vRealize Orchestrator, see the vRealize Orchestrator documentation. For training in vRealize Orchestrator development for vRealize Automation integrations, see the training courses available from VMware Education and the instructional material provided by VMware Learning.
❑ Run the provided workflow that inserts your custom workflow into an IaaS workflow stub and configures a blueprint to call the IaaS workflow stub. **Note** The workflow stubs are replaced by the event broker workflow subscriptions. They are still available, supported, and they can be used, but expect them to be removed in a future version of vRealize Automation. To ensure future product compatibility, you should use the workflow subscriptions to run custom workflows based on state changes. See Configuring Workflow Subscriptions to Extend vRealize Automation.	Assign a State Change Workflow to a Blueprint and Its Virtual Machines

Configuring the vRealize Automation Plug-in for Machine Extensibility

You configure your vRealize Automation and IaaS hosts, install the customizations for machine extensibility, and create a vRealize Automation endpoint for your vRealize Orchestrator instance.

Add a vRealize Automation Host

You can run a workflow to add a vRealize Automation host and configure the host connection parameters.

Procedure

1 From the drop-down menu in the Orchestrator client, select **Run** or **Design**.

2 Click the **Workflows** view.

3 Expand **Library > vRealize Automation > Configuration**.

4 Right-click the **Add a vRA host** workflow and select **Start workflow**.

5 Enter a unique name for the host in the **Host Name** text box.

6 Enter the URL address of the host in the **Host URL** text box.

   For example: *https://hostname*.

7. (Required) Enter the name of the tenant in the **Tenant** text box.

   To use the full functionality of the plug-in for a tenant, create a dedicated vRealize Automation host for each tenant.

8. Select whether to install the SSL certificates automatically without user confirmation.
9. (Optional) To configure the length of time vRealize Orchestrator waits for a connection or response from vRealize Automation, enter timeout intervals in the **Connection timeout (seconds)** and **Operation timeout (seconds)** text boxes.
10. Select the type of connection to the host from the **Session mode** drop-down menu.

Option	Actions
Shared Session	Enter the credentials for a vRealize Automation user in the **Authentication username** and **Authentication password** text boxes.
Per User Session	Connect using the credentials of the user that is currently logged in. You must be logged in to the Orchestrator client with the credentials of the vRealize Automation system administrator. To use this option with an external vRealize Orchestrator server, you must register the Orchestrator server in the vRealize Automation component registry. **Note** To register an external vRealize Orchestrator server in the component registry, you must configure Orchestrator to use vRealize Automation as an authentication provider. For more information, see *Installing and Configuring VMware vRealize Orchestrator*.

11. Click **Submit**.

What to do next

Add a vRealize Automation Infrastructure Administration host.

Add an IaaS Host

You can run a workflow to add the IaaS host of a vRealize Automation host and configure the connection parameters.

Procedure

1. From the drop-down menu in the Orchestrator client, select **Run** or **Design**.
2. Click the **Workflows** view.
3. Expand **Library > vRealize Automation > Infrastructure Administration > Configuration**.
4. Right-click **Add an IaaS host** and select **Start workflow**.
5. Select the vRealize Automation host for which you want to configure an IaaS host from the **vCAC host** drop-down menu.
6. Enter a unique name for the host in the **Host Name** text box.

7 Enter the URL of the machine on which your Model Manager is installed.

For example: https://*model_manager_machine.com*.

8 To install the SSL certificates, select **Yes**.

9 To use a proxy to access your model manager machine, select **Yes**.

If you select this option, you must provide the proxy host and the proxy port on the following page.

10 Click **Next**.

11 If you are configuring an explicit proxy, provide the proxy host URL and the port.

12 Click **Next**.

13 To configure your own timeout values, click **No**.

14 (Optional) To configure the length of time vRealize Orchestrator waits for a connection or response from vRealize Automation, enter timeout intervals in the **Connection timeout (seconds)** and **Operation timeout (seconds)** text boxes.

15 Click **Next**.

16 Select the host's authentication type.

Option	Description
SSO	Select this to use vCenter Single Sign-On.
NTLM	Select this to enable NT LAN Manager (NTLM) protocol-based authentication only if your Active Directory infrastructure relies on NTLM authentication. If you select this option, you must the additional NTLM credentials and authentication options.

17 If you selected NTLM, click **Next** and enter the name of the Workstation machine and the NetBIOS domain name.

18 Click **Submit**.

Install vRealize Orchestrator Customization

You can run a workflow to install the customized state change workflow stubs and Orchestrator menu operation workflows.

Note The workflow stubs are replaced by the event broker workflow subscriptions. They are still available, supported, and they can be used, but expect them to be removed in a future version of vRealize Automation. To ensure future product compatibility, you should use the workflow subscriptions to run custom workflows based on state changes. See Configuring Workflow Subscriptions to Extend vRealize Automation.

Procedure

1. From the drop-down menu in the Orchestrator client, select **Run** or **Design**.
2. Click the **Workflows** view.
3. Select **Library > vCloud Automation Center > Infrastructure Administration > Extensibility > Installation**.
4. Right-click the **Install vCO customization** workflow and select **Start workflow**.
5. Select an IaaS host.
6. Click **Next**.
7. Choose the lifecycle stages to which you want to add custom logic by selecting one or more state change workflow stubs to install.
8. Click **Submit**.

Create a vRealize Orchestrator Endpoint

You can create a vRealize Orchestrator endpoint to connect to a vRealize Orchestrator server.

You can configure multiple endpoints to connect to different vRealize Orchestrator servers, but you must configure a priority for each endpoint.

When executing vRealize Orchestrator workflows, vRealize Automation tries the highest priority vRealize Orchestrator endpoint first. If that endpoint is not reachable, then it proceeds to try the next highest priority endpoint until a vRealize Orchestrator server is available to run the workflow.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1. Select **Infrastructure > Endpoints > Endpoints**.
2. Select **New > Orchestration > vRealize Orchestrator**.
3. Enter a name and, optionally, a description.
4. Enter a URL with the fully qualified name or IP address of the vRealize Orchestrator server and the vRealize Orchestrator port number.

   The transport protocol must be HTTPS. If no port is specified, the default port 443 is used.

   To use the default vRealize Orchestrator instance embedded in the vRealize Automation appliance, type `https://`*`vrealize-automation-appliance-hostname`*`:443/vco`.
5. Provide your vRealize Orchestrator credentials in the **User name** and **Password** text boxes to connect to the vRealize Orchestrator endpoint.

   The credentials you use should have Execute permissions for any vRealize Orchestrator workflows to call from IaaS.

To use the default vRealize Orchestrator instance embedded in the vRealize Automation appliance, the user name is `administrator@vsphere.local` and the password is the administrator password that was specified when configuring SSO.

6 Enter an integer greater than or equal to 1 in **Priority** text box.

   A lower value specifies a higher priority.

7 (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.

8 Click **OK**.

Customizing IaaS Workflows By Using vRealize Orchestrator

You use a single workflow in vRealize Orchestrator to inject your custom logic into the IaaS workflow stubs and assign your customized life cycles to machine blueprints.

Note The workflow stubs are replaced by the event broker workflow subscriptions. They are still available, supported, and they can be used, but expect them to be removed in a future version of vRealize Automation. To ensure future product compatibility, you should use the workflow subscriptions to run custom workflows based on state changes. See Configuring Workflow Subscriptions to Extend vRealize Automation.

You must design your custom vRealize Orchestrator workflows to accept string inputs. If your custom workflow expects a complex data type, create a wrapper workflow that looks up this complex value and translates it to a string. For an example wrapping workflow, see the sample Workflow template, provided in **Library > vRealize Automation > Infrastructure > Extensibility**.

Assign a State Change Workflow to a Blueprint and Its Virtual Machines

You configure custom vRealize Orchestrator workflows to run at specific stages in the main machine workflow by associating your custom workflow with a state change workflow stub and assigning the workflows to a blueprint.

Note The workflow stubs are replaced by the event broker workflow subscriptions. They are still available, supported, and they can be used, but expect them to be removed in a future version of vRealize Automation. To ensure future product compatibility, you should use the workflow subscriptions to run custom workflows based on state changes. See Configuring Workflow Subscriptions to Extend vRealize Automation.

Prerequisites

Use the workflow template provided in the Extensibility subdirectory of the vRealize Automation plugin library to create a custom workflow to run during the machine lifecycle.

Procedure

1 From the drop-down menu in the Orchestrator client, select **Run** or **Design**.

2 Click the **Workflows** view.

3 Select **Library > vRealize Automation > Infrastructure > Extensibility**.

4 Right-click the **Assign a state change workflow to a blueprint and its virtual machines** workflow and select **Start workflow**.

5 Choose the lifecycle stage at which to run the workflow by selecting a stub from the **vCAC workflow stub to enable** drop-down menu.

6 Select an IaaS host.

7 Click **Next**.

8 Select the blueprint to which you want to assign the workflow.

9 Choose whether or not to apply these workflows to existing machines provisioned by using this blueprint.

10 Select the workflow you want to run during the machine lifecycle.

11 Configure which workflow input values are added as custom properties to the machine.

   a Add vCO workflow inputs as blueprint properties.

   b Add last vCO workflow run input values as blueprint properties.

12 Click **Submit**.

Configuring Workflow Subscriptions to Extend vRealize Automation

You create workflow subscriptions that use the event broker service to monitor the registered services for event messages in vRealize Automation, and then run a specified vRealize Orchestrator workflow when the conditions in the subscription are met. To configure the subscription you specify the event topic, the triggering conditions, and the workflow that runs when triggered.

Tenant administrators can create and manage the workflow subscriptions that are specific to their tenant.

The system administrator can create and manage system workflow subscriptions. The created system workflow subscriptions are active for events in any tenant and for system events.

Event Topics Provided With vRealize Automation

Event topics describe the type of event message that is sent to the event broker service by the other services. You select an event topic and configure the workflow subscription based on the topic.

Table 7-4. Event Topics

Event Topic Name	Description	Service
Blueprint component completed	A blueprint component that is part of a composite blueprint finishes provisioning. The component is any blueprint that is part of a composite blueprint.	composition-service
Blueprint component requested	A blueprint component that is part of a composite blueprint is requested. The component is any blueprint that is part of a composite blueprint.	composition-service
Blueprint configuration	A blueprint is created, updated, or deleted.	composition-service
Blueprint request completed	A composite blueprint finished provisioning. This event topic includes all blueprint components. It does not include standalone XaaS blueprints.	composition-service
Blueprint requested	A composite blueprint is requested. This event topic does not include XaaSblueprints.	composition-service
Business group configuration	A business group is created, updated, or deleted.	identity
Component action completed	An action ran on a deployed blueprint component when a deployment action was requested.	composition-service
Component action requested	An action to run on a deployed blueprint component is requested when a deployment action was requested.	composition-service
Deployment action completed	An action on a deployed blueprint finished running, including running all the component actions.	composition-service
Deployment action requested	An action on a deployed blueprint is requested.	composition-service
EventLog default event	A standard entry is added to the event log. The log entry is not distributed to subscribers.	eventlog-service
IPAM IP lifecycle event completion	An IP allocation or deallocation request is finished.	ipam-service
Machine lifecycle	A provided IaaS action is run on a provisioned machine.	iaas-service
Machine provisioning	An IaaS machine is in the process of being provisioned.	iaas-service

Table 7-4. Event Topics (continued)

Event Topic Name	Description	Service
Orchestration server configuration	A vRealize Orchestrator server configuration is created, updated, deleted, or modified to use a different default instance.	o11n-gateway-service
Orchestration server configuration (XaaS) - Obsolete	A vRealize Orchestrator server configuration is created, updated, deleted, or modified to use a different default instance.	advanced-designer-service
Post Approval	A post-approval policy level is configured to use the event subscription option.	approval-service
Pre Approval	A pre-approval policy level is configured to use the event subscription option.	approval-service
Resource reclamation completion event	A resource lease expired and the resources are reclaimed.	management-service

Workflow Subscriptions and Event Broker Terminology

As you work with the workflow subscriptions and the event broker service, you might encounter some terminology that is specific to the subscriptions and event broker service.

Table 7-5. Workflow Subscription and Event Broker Terminology

Term	Description
Event Topic	Describes a set of events that have same logical intent and the same structure. Every event is an instance of an event topic.
Event	Indicates a change in the state in the producer or any of the entities managed by it. The event is the entity that records information about the event occurrence.
Message	Transports information about the event between the various services and components. For example, from the producer to the event broker service, or from the event broker service to the subscribers.
Event Broker Service	The service that dispatches messages that are published by a producer to the subscribed consumers.
Payload	The event data.
Subscription	Indicates that a subscriber is interested in being notified about an event by subscribing to an event topic and defining the criteria that triggers the notification.
Subscriber	Consumes the events published to the event broker service based on the subscription definition. The subscriber might also be referred to as the consumer.

Table 7-5. Workflow Subscription and Event Broker Terminology (continued)

Term	Description
Provider	Registers event topics in the event broker service.
Producer	Publishes events to the event broker service.
System Administrator	A user with privileges to create, read, update, and delete tenant workflow subscriptions and system workflow subscriptions using the API or vRealize Automation plug-in. vRealize Automation does not include a user interface for the system administrator.
Tenant Administrator	The user with privileges to create, read, update, and delete tenant workflow subscriptions for their tenant.
Workflow Subscription	Specifies the event topic and conditions that trigger a vRealize Orchestrator workflow.
System Workflow Subscription	A specialized workflow subscription that reacts to system events and to events in all the tenants.
Tenant Workflow Subscription	A specialized workflow subscription that specifies which conditions trigger a vRealize Orchestrator workflow for events in the same tenant.

Blockable and Replyable Event Topics

Event topics might support blockable and replyable events. The behavior of a workflow subscription depends on whether the topic supports these event types and how you configure the workflow subscription.

Non-Blockable Event Topics

Non-blockable event topics only allow you to create non-blocking subscriptions. Non-blocking subscriptions are triggered asynchronously and you cannot rely on the order that the subscriptions are triggered. However, the triggering event is guaranteed to occur and the vRealize Orchestrator workflow associated with the subscription is run. Non-blocking subscriptions only return a response if the topic is replyable.

Blockable Event Topics

Some event topics support blocking. If a workflow subscription is marked as blocking, then all messages that meet the configured conditions are not received by any other workflow subscriptions with matching conditions until the first workflow is finished. If you have multiple blocking workflow subscriptions for the same event topic, you prioritize the subscriptions.

Blocking subscriptions run in priority order. The highest priority value is 0 (zero). If you have more than one blocking subscription for the same event topic with the same priority level, the subscriptions run in alphabetical order based on the name. After all blocking subscriptions are processed, the message is sent to all the nonblocking subscriptions at the same time. Because the blocking workflow subscriptions run synchronously, the changed event payload includes the updated event when the subsequent workflow subscriptions are notified.

You apply blocking to one or more workflow subscriptions depending on the selected workflow and your goals.

For example, you have two provisioning workflow subscriptions where the second workflow depends on the results of the first. The first one changes a property during provisioning, and a second record the new property, perhaps a virtual machine name, in a file system. The ChangeProperty subscription is prioritized as 0 and the RecordProperty is prioritized as 1 because it uses the results of the ChangeProperty subscription. When a virtual machine is provisioned, the ChangeProperty subscription begins running. Because the RecordProperty subscription conditions are based on a post-provisioning condition, a message triggers the RecordProperty subscription. However, because the ChangeProperty workflow is a blocking workflow, the message is not received until it is finished. When the name is changed and the first workflow is finished, the second workflow runs, recording the name in the file system.

Even if an event topic supports blocking, you can create a non-blocking workflow subscription if the workflow subscription does not have any dependant subsequent workflows. The workflow subscription is triggered and runs the vRealize Orchestrator workflow without further interaction from vRealize Automation or the outside system.

Replyable Event Topics

Some event topics support replies from the subscribed service. The service that registered the replyable event topic can accept a reply event that provides the workflow output, usually as a result of an interaction with a system or user. The reply output parameters must meet the criteria defined in the reply schema so that the vRealize Automation service that published the original replyable event can process it. For example, pre-approval and post-approval workflow subscriptions are replyable. If you create a workflow that sends an approval request to an external system, vRealize Automation processes the reply, approved or rejected, and the catalog item is provisioned or the user is notified that the request was .

The reply can be the output from the vRealize Orchestrator workflow or it can be a failure when the workflow times out or fails. If the reply is from the workflow output parameters, the reply must be in the correct reply schema format.

Best Practices for Creating vRealize Orchestrator Workflows for Workflow Subscriptions

A workflow subscription is based on a specific topic schema. To ensure that the subscriptions can start the vRealize Orchestrator workflows, you must configure them with the correct input parameters so that they work with the event data.

Workflow Input Parameters

The custom workflow that you create can include all the parameters or a single parameter that consumes all the data in the payload.

- To include individual parameters, configure one or more parameters. Ensure that the name and type match the name and type specified in the schema. Complex types from the schema should be defined as 'Properties' in the workflow.

- To use a single parameter, configure one parameter with a type of `Properties`. You can provide any useful name. For example, you can use `payload` as the parameter name.

Workflow Output Parameters

The custom workflow that you create can include output parameters that are relevant to subsequent events necessary for a reply event topic type.

If an event topic expects a reply, the workflow output parameters must match the reply schema.

Workflow Subscription Settings

The subscription options determine when a workflow runs based on event messages in vRealize Automation. Use the options to manage your subscriptions.

A subscription represents a user's intent to subscribe to events for a given event topic and to run a workflow when an event for the topic is received that matches defined conditions.

You must be a tenant administrator to create a workflow subscription. All workflow subscriptions are specific to your tenant.

To manage your workflow subscriptions, select **Administration > Events > Subscriptions**.

Table 7-6. Workflow Subscription options

Option	Description
New	Create a new subscription.
Edit	Modify the selected subscription. If the subscription is published, the saved changes are immediately active. You cannot edit the event topic or modify the blocking option for a published or unpublished subscription.
Publish	Make the subscription active. The events from the event broker service are processed and the subscription conditions are evaluated. If a configured condition is true, the workflow is triggered.
Unpublish	Return a subscription to a draft state. The subscription is no longer active in your environment and no longer receives events. If you republish a subscription, the subscription starts to receive new events. Past events are not received.
Delete	Delete the selected subscription.

Assign Event Topics to a Subscription

Event topics are classes of events provided in vRealize Automation. You select the event topic on which to define the subscription.

Event topics are the categories that group similar events together. When assigned to a subscription, event topics define which event triggers the subscription.

Procedure

1 Select **Administration > Events > Subscriptions**.

2 Click **New** and select an **Event Topic**.

Table 7-7. Event Topic Details

Event Topic Details	Description
Topic ID	Event topic identifier.
Name	Name of the event topic.
Description	Description of the event topic.
Publisher	Name of the service for which this event topic is registered.
Blockable	Indicates whether you can create a blocking subscription for this event topic. Blocking subscriptions are used to change the event's payload or run your custom logic when the results of a second workflow for the same event depend on the results of the first workflow.
Replyable	Indicates whether an event topic subscription can publish a reply event to the service that originally produced the event. If the value is yes, a reply is sent to the service that published the original event when the workflow finishes. The reply contains the output of the vRealize Orchestrator workflow and any error details.
Schema	Describes the structure of the event's payload. You can use the schema to create workflows that can use the payload information.

Assign Workflow Conditions to a Subscription

The conditions you configure for the subscription determine whether the workflow is triggered to run based on the event data.

You can define workflow conditions to control how a workflow is initiated. If you select **Run based on conditions**, the available types can include:

- Data

  Tis includes information in the event message that is specific to the selected event topic. For example, if you create a condition for the virtual machine lifecycle event topic, the data fields are related to blueprints and virtual machines. If you select a pre-approval event topic, the data fields are related to approval policies.

  You can also add conditions for fields that are not included in the schema by entering the path in the text box above the tree. Use the format `${PATH}`. PATH is the path in the schema. Separate the nodes using ~. For example, `${data~machine~properties~SomeCustomProperty}`.

- Core event message values

  This includes general information about the event message. For example, the event type, time stamp, or user name.

Prerequisites

Procedure

1 Select **Administration > Events > Subscriptions**.

2 Click **New**and select an **Event Topic**.

3 Click **Next** and define your **Workflow Conditions**.

Table 7-8. Condition Types

Condition	Description
Run for all events	The selected workflow runs when the message for this event topic is received.
Run based on conditions	The selected workflow runs when the event message is detected and the event meets the configured conditions. If you select this option, you must define conditions based on the event data to trigger the selected workflow for this subscription. - **Single condition**. The workflow is triggered when the configured clause is true. - **All of the following**. The workflow is triggered when all the clauses are true and you provided at least two conditions. - **Any of the following**. The workflow is triggered when at least one of the clauses is true and you provided at least two conditions. - **Not the following**. The workflow is triggered when none of the clauses are true. If you create a condition based on a constant value, the value is processed as case insensitive. For example, if your condition is Blueprint name contains UNIX, but your blueprints use Unix in the name, the condition still processes correctly. To change the condition name to match the blueprint name, you must first change the value to something that does not contain the same string. For example, to edit the condition UNIX, change the value to xxxx, save it, then change xxxx to Unix and save it.

Assign a Workflow to a Subscription

The vRealize Orchestrator workflow that you select runs when the subscription conditions are evaluated as true.

Workflows combine ABX actions, decisions, and results that, when performed in a particular order, complete a specific task or a specific process in a virtual environment. Workflows perform tasks such as provisioning virtual machines, backing up, performing regular maintenance, sending emails, performing SSH operations, managing the physical infrastructure, and other general utility operations. Workflows accept inputs according to their function. Workflows can also call upon other workflows. For example, you can reuse in several different workflows a workflow that starts a virtual machine.

You can link workflows in a subscription to automate a procedure in response to a triggering event. This allows the workflow to perform and generate results without user intervention. More specifically it adds the ability to run workflows on virtual machine provisioning lifecycle events. You can also reuse subscription outputs to share data between workflows in the same state. Workflows registered to the same life cycle state can merge output payloads.

Prerequisites

The workflow must exist in vRealize Orchestrator as listed in **Administration > vRO Configuration > Server Configuration**.

Procedure

1 Select **Administration** > **Events** > **Subscriptions**.

2 Click **New** and select an **Event Topic**.

3 Click **Next** and define **Workflow Conditions**.

4 Click **Next** and select to the **Workflow** to apply to the subscription.

Table 7-9. Workflow Tab

Workflow Tab	Description
Select a Workflow	Navigate to the workflow.
Selected Workflow	Displays information about the workflow, including input and output parameters, so that you can verify it is the one that you want to run.

Define Workflow Subscription Details

The subscription details determine how the subscription is processed.

You can configure and customize the subscription further by defining additional subscription details.

Procedure

1 Select **Administration** > **Events** > **Subscriptions**

2 Click **New** and select an **Event Topic**.

3 Click **Next** and assign **Workflow Conditions**.

4 Click **Next** and select a **Workflow** to assign to the subscription.

5 Click **Next** and define **Workflow Subscription Details**.

Table 7-10. Workflow Details

Detail	Description
Name	By default, the displayed name is the name of the selected workflow. This name is displayed in the subscription list. The name must be unique in the tenant.
Priority	The order in which the blocking subscriptions run. Zero is the highest priority. If an event topic has multiple blocking workflow subscriptions with the same priority, the subscriptions are then processed in alphabetical order based on the subscription name. This option is only available for blocking workflow subscriptions.
Timeout (min)	Enter the number of minutes the workflow has to finish before it is considered failed. If the workflow fails to finish in the allowed amount of time, the workflow is canceled and the message is sent to the next subscription in the priority list. If you do not provide a value, the timeout is unlimited. Services that expect a reply, to blocking or replyable events, might have their own default timeout values. For example, IaaS provisioning and life cycle event topics time out at 30 minutes. This value is configured on the IaaS server. Approval topics have a 24-hour default value. This value is configured on the system.
Description	By default, the displayed description is the workflow description.

Table 7-10. Workflow Details (continued)

Detail	Description
Blocking	Determines if the workflow can block subsequent workflows for the same event topic from receiving an event message when waiting for a reply. Subscriptions with blocking enabled receive messages before subscriptions that are not configured as blocking for the same event topic, based on priority order. When the workflow is finished, a message is sent to the next prioritized blocking subscription. After all blocking subscriptions have processed, the message is sent to all non-blocking subscriptions simultaneously. The blocking option is available only if the event topic is blockable. This information is provided on the Event Topic tab. The blocking eligibility is indicated on the Event Topic tab. ■ If you do not select the check box, the event broker does not block subsequent workflows. ■ If you select check box, the event broker calculates which workflow subscriptions are eligible for this event based on the configured conditions and runs the workflows in priority order. The event broker waits for a response from each workflow before running the next one. All the changed parameters from the running of the current workflow are passed to the next one in the queue. When waiting for a response, no other workflows are notified of the event until the consuming system responds. You cannot modify this option after the workflow subscription is created.
Stop processing if workflow fails	If the blocking workflow subscription fails, no subsequent workflows run until the failure error is resolved. A failure message is added to the event log and an email is sent to the requesting user.

6 Click **Finish**.

Working with Provisioning and Life Cycle Workflow Subscriptions

You create provisioning and life cycle workflow subscriptions so that you can use vRealize Orchestrator to extend your management of IaaS machines. The provisioning subscriptions extend what you can do during the provisioning process. The life cycle subscriptions extend what you can do when the user is managing the provisioned items.

IaaS Service Integration

You create a workflow subscription based on a provisioning or life cycle event topic that runs a custom vRealize Orchestrator workflow based on a message generated by the IaaS service. vRealize Automation includes two event topics that you can use for IaaS integration.

- Machine provisioning. Create workflow subscriptions that run workflows during the provisioning and disposal of IaaS machines.
- Machine lifecycle. Create workflow subscriptions that run workflows related to management actions that an owning user runs on the provisioned machine.

Configuring vRealize Orchestrator Workflows for Provisioning and Life Cycle Workflows

You must configure your vRealize Orchestrator workflows to support the IaaS service message.

Provisioning and Life Cycle Event Topic Schema

The machine provisioning and machine life cycle event topics use the same life cycle schema. The differences are in the triggering states. Machine provisioning receives messages based on provisioning states and events, and machine life cycle receives messages based on active states and events. Some provisioning states include BuildingMachine and Disposing. Some life cycle states include InstallTools and Off.

The event message is the event data payload. The following is the structure of the event data payload.

```
{
  machine : {
      id                : STRING,      /* IaaS machine ID */
      name              : STRING,      /* machine name */
      externalReference : STRING,      /* machine ID on the hypervisor */
      owner             : STRING,      /* machine owner */
      type              : INTEGER,     /* machine type: 0 – virtual machine; 1 – physical machine; 2
– cloud machine */
      properties        : Properties   /* machine properties, see notes below how to expose virtual
machine properties */
  },
  blueprintName   : STRING,      /* blueprint name */
  componentId     : STRING,      /* component id */
  componentTypeId : STRING,      /* component type id */
  endpointId      : STRING,      /* endpoint id */
  requestId       : STRING,      /* request id */
  lifecycleState  : {                                                /* see Life Cycle State
Definitions*/
      state : STRING,
      phase : STRING,
      event : STRING
  },
  virtualMachineEvent                  : STRING,     /* fire an event on that machine – only processed
by Manager Service as consumer */
  workflowNextState                    : STRING,     /* force the workflow to a specific state – only
processed by Manager Service as consumer */
```

```
  virtualMachineAddOrUpdateProperties : Properties, /* properties on the machine to add/update - only
processed by Manager Service as consumer */
  virtualMachineDeleteProperties      : Properties  /* properties to remove from the machine - only
processed by Manager Service as consumer */
}
```

The vRealize Orchestrator parameters are mapped to the event's payload by name and type.

When you use `virtualMachineEvent` and `workflowNextState` as output parameters, the values that you provide must represent a state or event from the workflow that triggered the event and started the current vRealize Orchestrator workflow. To review the possible life cycle states and events, see VMPS Main Workflow Life Cycle States and Provisioning Life Cycle States by Machine Type.

Working with Extensibility Custom Properties

The virtual machine custom properties are not included in the event payload unless they are specified as an extensibility custom property for the life cycle state. You can add these properties to IaaS endpoints, reservations, blueprints, requests, and other objects that support custom properties.

The format of the custom property that you add to an object is `Extensibility.Lifecycle.Properties.{workflowName}.{stateName}`.

For example, if you want to include hidden properties and all properties starting with "Virtual" when the virtual machine state is BuildingMachine, you add the custom properties to the machine in the blueprint. The custom property name for this example is `Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.BuildingMachine`, and the values are `__*` and `Virtual*`, separated by a comma.

The double underscore (`__*`) includes the hidden properties. The `Virtual*` value includes all properties that begin with virtual. The asterisk (*) is a wildcard and can be used as the only value, but using the wildcard this way results in the transfer of large amounts of data.

If you have multiple, subsequently triggered workflow subscriptions that include custom properties, you must include the appropriate entries in the workflows to ensure that the payload check retains the custom properties.

Table 7-11. Task Entries to Preserve Custom Properties

State	Task Entries
Added or updated custom properties	`virtualMachineAddOrUpdateProperties = payload.virtualMachineAddOrUpdateProperties || new Properties();`
Deleted custom properties	`virtualMachineDeleteProperties = payload.virtualMachineDeleteProperties || new Properties();`

Creating a vRealize Orchestrator Workflow Based on the Life Cycle or Provisioning Schema

The custom workflow that you create must have an input parameter that is `payload` with the type `Properties`. The provisioning or life cycle event data payload is put in this parameter when the workflow runs in vRealize Orchestrator. You can also include separate input parameters that match the name and the type of the fields in the event's payload.

Workflow Subscription Life Cycle State Definitions

If you configure workflow subscription conditions based on life cycle states, the following definitions might help you identify the values.

Each message contains a lifecycleState element that is based on the IaaS machine state changes.

The element has the following structure in the message.

```
lifecycleState : {
    state : STRING,
    phase : STRING,
    event : STRING
}
```

Table 7-12. LifecycleState Elements

Property	Description	Format and Values	Examples
state	Contains workflow name and state name.	{workflowName}.{stateName}	■ VMPSMasterWorkflow32.Requested ■ VMPSMasterWorkflow32.MachineActivated ■ BasicVmWorkflow.BuildComplete
phase	Contains the phase that triggered a message.	PRE, POST, EVENT	■ PRE. An event is published when entering this state. ■ POST. An event is published when exiting this state. ■ EVENT. An event is published when an IaaS event is received in this state..
event	Contains the event. This property is optional and exists only when the phase is EVENT.	{workflowName}.{stateName}.EVENT.{eventName}	■ VMPSMasterWorkflow32.Requested.EVENT.OnProvisionMachine ■ VMPSMasterWorkflow32.VMPSMasterWorkflow32.EVENT.OnBuildSuccess ■ BasicVmWorkflow.CreatingMachine.EVENT.OnCreatingMachineComplete

VMPS Main Workflow Life Cycle States

The VMPS main workflow life cycle states represent an IaaS virtual machine life cycle, from request to destruction .You can use the VMPS main workflow states and events when you create triggering conditions based on life cycle state events and life cycle state names.

Each virtual machine goes through four basic stages.

- Request. Includes approvals.

- Provision. Includes different provisioning types, such as create, clone, kickstart, or WIM.
- Manage. Includes actions, such as power on, power off, or snapshot.
- Destroy. Includes deactivating, unprovisioning, and disposing of the machine.

These basic stages are included in the main workflow. You can use the *VMPSMasterWorkflow32* states when you create conditions for the following event topics:

- Machine life cycle
- Machine provisioning

The global event states are messages sent to the event broker by the VMPS Main Workflow. Global events can be triggered at any time.

You can subscribe the client to listen for events, but the events should not be thrown unless the table entry has a trigger string value. For example, Events [Triggering String] (Topic).

Table 7-13. Global Events

State(Topic)	Events [Triggering String] (Topic)
Global	■ onBuildFailure (Provision) ■ OnBuildSuccess (Provision) ■ OnFinalizeMachine [Destroy] (Provision) ■ OnForceUnregisterEvent [ForceUnregister] (Provision) ■ ReconfigureVM.Pending [ReconfigureVM.Pending] (Active) ■ ReconfigureVM.ExecutionUpdated (Active) ■ ReconfigureVM.RetryRequestMade (Active) ■ ReconfigureVM.Failed (Active) ■ ReconfigureVM.Successful (Active) ■ ReconfigureVM.Complete (Active) ■ ReconfigureVM.Canceled (Active)

The active global states are actions that you can run on provisioned machines.

Table 7-14. Active Events

State	Events [Triggering String] (Topic)
Active	■ OnExpireLease [Expire] (Active) ■ OnForceExpire [ForceExpire] (Active) ■ onReprovision [Reprovision] (Active) ■ onResetBuildSuccess [ResetBuildSuccess] (Active)

In the main workflow, provision events occur during the machine provisioning life cycle. Active events are actions you can run on provisioned machines. For an illustration of the main workflow, see Example of VMPS Main Workflow.

Each machine type has its own provisioning workflow. For information about individual machine types, see Provisioning Life Cycle States by Machine Type.

Table 7-15. VMPSMasterWorkflow32 States and Events

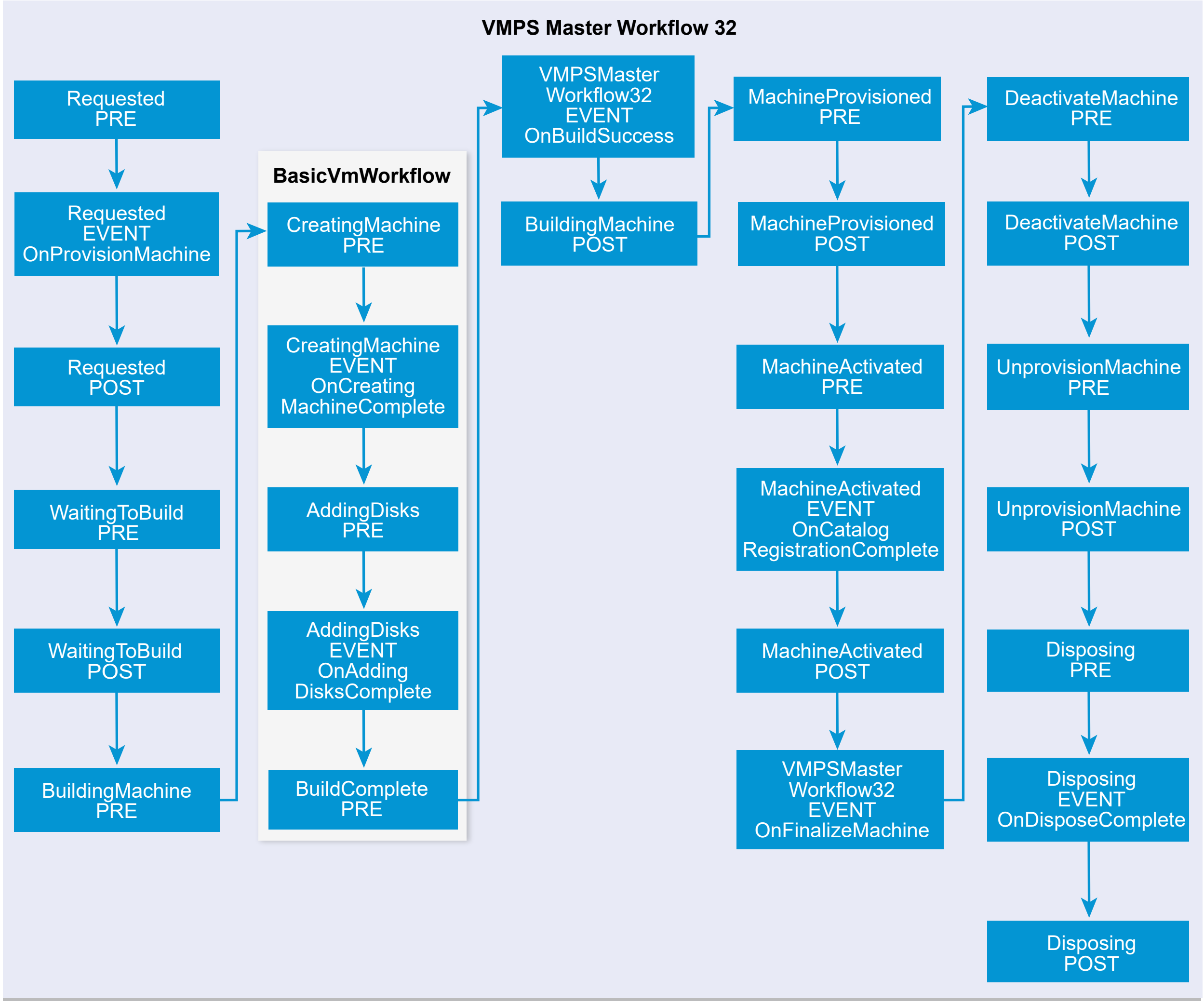
State(Topic)	Events [Triggering String] (Topic)
BuildingMachine ■ Pre(Provision) ■ Post(Provision)	
DeactivateMachine ■ Pre(Provision) ■ Post(Provision)	
Disposing ■ Pre(Provision) ■ Post(Provision)	■ OnDisposeComplete(Provision) ■ OnDisposeTimeout(Provision) ■ OnUnregisterMachine [Unregister] (Provision)
Expired ■ Pre(Active) ■ Post(Active)	■ OnActiveExpiredMachine [ActivateExpiredMachine] (Active) ■ TurnOffFromExpired [TurnOffExpiredMachine] (Active)
InstallTools ■ Pre(Active) ■ Post(Active)	■ InstallToolsComplete(Active) ■ TimeoutInstallTools(Active)
Leased	■ OnChangeLease (Active) ■ OnUpdateDescription (Active) ■ OnUpdateOwner (Active)
MachineActivated ■ Pre(Provision) ■ Post(Provision)	■ OnCatalogRegistrationComplete (Provision)
MachineProvisioned ■ Pre(Provision) ■ Post(Provision)	
Off ■ Pre(Active) ■ Post(Active)	■ OnForceOn [ForceOn] (Active) ■ OnResetOff [Turn Off] (Active) ■ OnTurnOn [Turn On] (Active)
On ■ Pre(Active) ■ Post(Active)	■ OnForceOff [ForceOff] (Active) ■ onInstallTools [InstallTools] (Active) ■ OnReboot [Reboot] (Active) ■ OnReset [Reset] (Active) ■ OnResetOn [Turn On] (Active) ■ OnShutdown [Shutdown] (Active) ■ OnSuspend [Suspend] (Active) ■ OnTurnOff [Turn Off] (Active)
Rebooting ■ Pre(Active) ■ Post(Active)	■ OnRebootComplete(Active) ■ TimoutFromReboot(Active)

Table 7-15. VMPSMasterWorkflow32 States and Events (continued)

State(Topic)	Events [Triggering String] (Topic)
RegisterMachine - Pre(Provision) - Post(Provision)	- onRegisterComplete(Provision) - RegisterTimeout(Provision)
Requested - Pre(Provision) - Post(Provision)	- OnProvisionMachine [Provision] (Provision)
Resetting - Pre(Active) - Post(Active)	- OnResetComplete(Active) - TimoutFromReset(Active)
ShuttingDown - Pre(Active) - Post(Active)	- OnShutdownComplete(Active) - TimoutFromShutdown(Active)
Suspending - Pre(Active) - Post(Active)	- OnSuspendComplete(Active) - TimoutFromSuspend(Active)
TurningOff - Pre(Active) - Post(Active)	- OnTurningOffComplete(Active) - TimoutFromPowerOff(Active)
TurningOn - Pre(Active) - Post(Active)	- OnTurningOnComplete(Active) - TimeoutPowerOn(Active)
UnprovisionMachine - Pre(Provision) - Post(Provision)	
WaitingToBuild - Pre(Provision) - Post(Provision)	

Example of VMPS Main Workflow

The VMPS workflow is the main workflow in which the other provisioning workflows are embedded. This example includes the Basic VM Workflow to illustrate the life cycle of a virtual machine. It does not represent a specific workflow in your environment.



Provisioning Life Cycle States by Machine Type

The life cycle states by machine type are specific to certain virtual machine types. In addition to the master workflow, you can use the provisioning workflow states and events when you create triggering conditions for workflow subscriptions.

You can subscribe the client to listen for events, but the events should not be thrown unless the table entry has a trigger string value. For example, Events [Triggering String] (Topic).

Blade Logic Bare Metal

State (Topic)	Events (Topic)
BuildFinished ■ Pre(Provision)	
CreatingMachine ■ Pre(Provision)	

Opsware Bare Metal

State (Topic)	Events (Topic)
BuildFinished ■ Pre(Provision)	
OpswareRegister ■ Pre(Provision)	■ OnOpswareRegister(Provision)

Cloud Provisioning Workflow

State (Topic)	Events (Topic)
BuildComplete ■ Pre(Provision)	
CloudProvisioning ■ Pre(Provision)	■ OnCloudProvisioningTimeout(Provision)
FailedProvisioning ■ Pre(Provision)	

App Service Provisioning Workflow

State (Topic)	Events (Topic)
AppServiceProvisioning ■ Pre(Provision)	■ OnAppServiceProvisioningTimeout(Provision)
BuildComplete ■ Pre(Provision)	
FailedProvisioning ■ Pre(Provision)	

Basic VM Workflow

State (Topic)	Events (Topic)
AddingDisks ■ Pre(Provision)	■ OnAddingDisksComplete(Provision) ■ OnAddingDisksTimeout(Provision)
BuildComplete ■ Pre(Provision)	
CreatingMachine ■ Pre(Provision)	■ OnCreatingMachineComplete(Provision) ■ OnCreatingMachineTimeout(Provision)
FailedProvisioning ■ Pre(Provision)	

Opsware Virtual

State (Topic)	Events (Topic)
AddingDisks ■ Pre(Provision)	■ OnAddingDisksComplete(Provision) ■ OnAddingDisksTimeout(Provision)
BuildFinished ■ Pre(Provision)	
CreatingVM ■ Pre(Provision)	■ OnCreateVMComplete(Provision) ■ OnCreateVMTimeout(Provision)
InitialPowerOn ■ Pre(Provision)	■ OnInitialPowerOnComplete(Provision) ■ OnInitialPowerOnTimeout(Provision)
OpswareRegister ■ Pre(Provision)	■ OnOpswareRegister(Provision)

Cloud Linux Kickstart Workflow

State (Topic)	Events (Topic)
BuildComplete ■ Pre(Provision)	
CreatingMachine ■ Pre(Provision)	■ OnCreatingMachineComplete(Provision) ■ OnCreatingMachineTimeout(Provision)
CustomizeOS ■ Pre(Provision)	■ OnCustomizeOSComplete(Provision) ■ OnCustomizeOSTimeout(Provision)
FailedProvisioning ■ Pre(Provision)	
InitialPowerOn ■ Pre(Provision)	■ OnInitialPowerOnComplete(Provision) ■ OnInitialPowerOnTimeout(Provision)
InstallingOS ■ Pre(Provision)	■ OnInstallingOSComplete(Provision) ■ OnInstallingOSTimeout(Provision)

Clone Workflow

State (Topic)	Events (Topic)
BuildComplete ■ Pre(Provision)	
CloneMachine ■ Pre(Provision)	■ OnCloneMachineComplete(Provision) ■ OnCloneMachineTimeout(Provision)
CustomizeMachine ■ Pre(Provision)	■ OnCustomizeMachineComplete(Provision) ■ OnCustomizeMachineTimeout(Provision)
CustomizeOS	■ OnCustomizeOS(Provision) ■ OnCustomizeOSComplete(Provision) ■ OnCustomizeOSTimeout(Provision)

State (Topic)	Events (Topic)
EjectCD ■ Pre(Provision)	■ OnEjectCDComplete(Provision) ■ OnEjectCDTimeout(Provision)
FailedProvisioning ■ Pre(Provision)	
FinalizeProvisioning ■ Pre(Provision)	■ OnFinalizeComplete(Provision) ■ OnFinalizeTimeout(Provision)
InitialPowerOn ■ Pre(Provision)	■ OnInitialPowerOnComplete(Provision) ■ OnInitialPowerOnTimeout(Provision)
InstallSoftware ■ Pre(Provision)	■ OnInstallSoftwareComplete(Provision) ■ OnInstallSoftwareTimeout(Provision)
MountCD ■ Pre(Provision)	■ OnMountCDComplete(Provision) ■ OnMountCDTimeout(Provision)
PostInstallSoftwareChecks ■ Pre(Provision)	
PrepareInstallSoftware ■ Pre(Provision)	

Cloud WIM Image Workflow

State (Topic)	Events (Topic)
BuildComplete ■ Pre(Provision)	
CreatingMachine ■ Pre(Provision)	■ OnCreatingMachineComplete(Provision) ■ OnCreatingMachineTimeout(Provision)
FailedProvisioning ■ Pre(Provision)	
InitialPowerOn ■ Pre(Provision)	■ OnInitialPowerOnComplete(Provision) ■ OnInitialPowerOnTimeout(Provision)
InstallOS ■ Pre(Provision)	■ onInstallOSComplete(Provision) ■ OnInstallOSTimeout(Provision)
Reboot ■ Pre(Provision)	■ OnRebootComplete(Provision) ■ OnRebootTimeout(Provision)
SetupOS ■ Pre(Provision)	■ OnSetupOSComplete(Provision) ■ OnSetupOSTimeout(Provision)

External Provisioning Workflow

State (Topic)	Events (Topic)
AddingDisks - Pre(Provision)	- OnAddingDisksComplete(Provision) - OnAddingDisksTimeout(Provision)
BuildComplete - Pre(Provision)	
CreatingMachine - Pre(Provision)	- OnCreatingMachineComplete(Provision) - OnCreatingMachineTimeout(Provision)
EpiRegister - Pre(Provision)	- OnEpiRegisterComplete(Provision)
FailedProvisioning - Pre(Provision)	
InitialPowerOn - Pre(Provision)	- OnInitialPowerOnComplete(Provision) - OnInitialPowerOnTimeout(Provision)

Linux Kickstart Workflow

State (Topic)	Events (Topic)
AddingDisks - Pre(Provision)	- OnAddingDisksComplete(Provision) - OnAddingDisksTimeout(Provision)
BuildComplete - Pre(Provision)	
CreatingMachine - Pre(Provision)	- OnCreatingMachineComplete(Provision) - OnCreatingMachineTimeout(Provision)
CustomizeOS - Pre(Provision)	- OnCustomizeOSComplete(Provision) - OnCustomizeOSTimeout(Provision)
EjectingCD - Pre(Provision)	- OnEjectingCDComplete(Provision) - OnEjectingCDTimeout(Provision)
FailedProvisioning - Pre(Provision)	
InitialPowerOn - Pre(Provision)	- OnInitialPowerOnComplete(Provision) - OnInitialPowerOnTimeout(Provision)
InstallingOS - Pre(Provision)	- OnInstallingOSComplete(Provision) - OnInstallingOSTimeout(Provision)

Physical Provisioning Workflow

State (Topic)	Events (Topic)
FailedProvisioning - Pre(Provision)	
FinalizeProvisioning - Pre(Provision)	- OnFinalizeProvisioningTimeout(Provision)

State (Topic)	Events (Topic)
InitializeProvisioning ■ Pre(Provision)	■ OnInitializeProvisioningTimeout(Provision)
InitialPowerOn ■ Pre(Provision)	■ OnInitialPowerOnTimeout(Provision)
InstallOS ■ Pre(Provision)	■ OnInstallOSComplete(Provision) ■ OnInstallOSTimeout(Provision)
Reboot ■ Pre(Provision)	■ OnRebootComplete(Provision) ■ OnRebootTimeout(Provision)
SetupOS ■ Pre(Provision)	■ OnSetupOSComplete(Provision) ■ OnSetupOSTimeout(Provision)

Physical PXE Provisioning Workflow

State (Topic)	Events (Topic)
CheckHardwareType ■ Pre(Provision)	
CleanPxe ■ Pre(Provision)	■ OnCleanPxeTimeout(Provision)
FailedProvisioning ■ Pre(Provision)	
FinalizeProvisioning ■ Pre(Provision)	■ OnFinalizeProvisioningTimeout(Provision)
InitializeProvisioning ■ Pre(Provision)	■ OnInitializeProvisioningTimeout(Provision)
InitialPowerOn ■ Pre(Provision)	■ OnInitialPowerOnTimeout(Provision)
InstallOS ■ Pre(Provision)	■ OnInstallOSComplete(Provision) ■ OnInstallOSTimeout(Provision)
Reboot ■ Pre(Provision)	■ OnRebootComplete(Provision) ■ OnRebootTimeout(Provision)
SetupOS ■ Pre(Provision)	■ OnSetupOSComplete(Provision) ■ OnSetupOSTimeout(Provision)
SetupPxe ■ Pre(Provision)	■ OnSetupPxeTimeout(Provision)

Physical SCCM Provisioning Workflow

State (Topic)	Events (Topic)
CheckHardwareType ■ Pre(Provision)	
Complete ■ Pre(Provision)	■ OnCompleteProvisioningComplete(Provision) ■ OnCompleteProvisioningTimeout(Provision)
FailedProvisioning ■ Pre(Provision)	■ OnFailedProvisioningTimeout(Provision)
FinalizeProvisioning ■ Pre(Provision)	■ OnFinalizeProvisioningTimeout(Provision)
InitializeProvisioning ■ Pre(Provision)	■ OnInitializeProvisioningTimeout(Provision)
InitialPowerOn ■ Pre(Provision)	■ OnInitialPowerOnTimeout(Provision)
SccmRegistration ■ Pre(Provision)	■ OnSccmRegistrationTimeout(Provision)

Physical SCCM PXE Provisioning Workflow

State (Topic)	Events (Topic)
CheckHardwareType ■ Pre(Provision)	
CleanPxe ■ Pre(Provision)	■ OnCleanPxeTimeout(Provision)
Complete ■ Pre(Provision)	■ OnCompleteProvisioningComplete(Provision) ■ OnCompleteProvisioningTimeout(Provision)
Disposing ■ Pre(Provision)	
FailedProvisioning ■ Pre(Provision)	■ OnFailedProvisioningTimeout(Provision)
FinalizeProvisioning ■ Pre(Provision)	■ OnFinalizeProvisioningTimeout(Provision)
InitializeProvisioning ■ Pre(Provision)	■ OnInitializeProvisioningTimeout(Provision)
InitialPowerOn ■ Pre(Provision)	■ OnInitialPowerOnTimeout(Provision)
SccmRegistration ■ Pre(Provision)	■ OnSccmRegistrationTimeout(Provision)
SetupPxe ■ Pre(Provision)	■ OnSetupPxeTimeout(Provision)

vApp Clone Workflow

State (Topic)	Events [Triggering String] (Topic)
Global	■ OnFailProvisioning(Provision) ■ OnMasterProvisioned(Provision)
BuildComplete ■ Pre(Provision)	
CloneMachine ■ Pre(Provision)	■ OnCloneMachineComplete(Provision) ■ OnCloneMachineTimeout(Provision)
CustomizeMachine ■ Pre(Provision)	■ OnCustomizeMachineComplete(Provision) ■ OnCustomizeMachineTimeout(Provision)
CustomizeOS	■ OnCustomizeOS(Provision) ■ OnCustomizeOSComplete(Provision) ■ OnCustomizeOSTimeout(Provision)
FailedProvisioning ■ Pre(Provision)	
FinalizeProvisioning ■ Pre(Provision)	■ OnFinalizeComplete(Provision) ■ OnFinalizeTimeout(Provision)
InitialPowerOn ■ Pre(Provision)	■ OnInitialPowerOnComplete(Provision) ■ OnInitialPowerOnTimeout(Provision)
WaitingForMaster ■ Pre(Provision)	■ OnWaitingForMasterTimeout(Provision)

Virtual SCCM Provisioning Workflow

State (Topic)	Events (Topic)
AddingDisks ■ Pre(Provision)	■ OnAddingDisksComplete(Provision) ■ OnAddingDisksTimeout(Provision)
BuildComplete ■ Pre(Provision)	
CreatingMachine ■ Pre(Provision)	■ CreatingMachineComplete(Provision) ■ OnCreatingMachineTimeout(Provision)
Disposing ■ Pre(Provision)	
EjectingCD ■ Pre(Provision)	■ OnEjectingCDComplete(Provision) ■ OnEjectingCDTimeout(Provision)
FailedProvisioning ■ Pre(Provision)	
InitialPowerOn ■ Pre(Provision)	■ OnInitialPowerOnComplete(Provision) ■ OnPowerOnTimeout(Provision)

State (Topic)	Events (Topic)
InstallingOS - Pre(Provision)	- OnInstallingOSComplete(Provision) - OnInstallingOSTimeout(Provision)
SccmRegistration - Pre(Provision)	- OnSccmRegistrationTimeout(Provision)

WIM Image Workflow

State (Topic)	Events (Topic)
AddingDisks - Pre(Provision)	- OnAddingDisksComplete(Provision) - OnAddingDisksTimeout(Provision)
BuildComplete - Pre(Provision)	
CreatingMachine - Pre(Provision)	- OnCreatingMachineComplete(Provision) - OnCreatingMachineTimeout(Provision)
EjectingCD - Pre(Provision)	- OnEjectingCDComplete(Provision) - OnEjectingCDTimeout(Provision)
FailedProvisioning - Pre(Provision)	
InitialPowerOn - Pre(Provision)	- OnInitialPowerOnComplete(Provision) - OnInitialPowerOnTimeout(Provision)
InstallOS - Pre(Provision)	- onInstallOSComplete(Provision) - OnInstallOSTimeout(Provision)
Reboot - Pre(Provision)	- OnRebootComplete(Provision) - OnRebootTimeout(Provision)
SetupOS - Pre(Provision)	- OnSetupOSComplete(Provision) - OnSetupOSTimeout(Provision)

Configuring the Timeout Values for States and Events

The default timeout value for all states and events is 30 minutes and is configured in the vRealize Automation global settings. Some workflows might take more time to run successfully. To accommodate different workflows in your environment, you can add timeout override values for individual workflows or states.

To modify the default timeout value, select **Infrastructure > Administration > Global Settings** and edit the value for **Extensibility lifecycle message timeout**. If you make changes to the global setting, you must restart the manager service.

To configure individual timeout values, add the workflow or event property to the `appSetting` section of the `ManagerService.exe.config file`, located on the IaaS server. The file is typically located in the `%System-Drive%\Program Files x86\VMware\vCAC\Server` directory. You should always make a copy of the file before editing it. If you make changes to the individual settings, you must restart the manager service.

The basic format for the keys is similar to the following examples.

- For a workflow. `Extensibility.{workflow}.Timeout`
- For events. `Extensibility.{workflow}.{state}.EVENT.{event}.Timeout`
- For states. `Extensibility.{workflow}.{state}.(PRE/POST).Timeout`

Use the following as examples when adding keys to the `appSetting` section. The timeout value format is D.HH:mm:ss.ms. D is day and ms is milliseconds. Day and milliseconds are optional. Hours, minutes, and seconds are required.

- To set the timeout for the entire BasicVmWorkflow workflow to 30 minutes, add `<add key="Extensibility.BasicVmWorkflow.Timeout" value="00:30:00"/>`.
- To set the timeout for the OnFinalizeMachine global event in the VMPSMasterWorkflow32 to two hours, add `<add key="Extensibility.VMPSMasterWorkflow32.VMPSMasterWorkflow32.EVENT.OnFinalizeMachine.Timeout" value="02:00:00"/>`.
- To set the timeout for the pre-request state of the VMPSMasterWorkflow32 to 2 days, add `<add key="Extensibility.VMPSMasterWorkflow32.Requested.PRE.Timeout" value="2.00:00:00"/>`.

Configuring the Error Behavior for States and Events

The workflow subscription timeout and error handling has default behavior. You can custom the behavior for machines in your environment.

IaaS handles event timeout and error proccessing from the Event Broker Service.

At each state transition, SendEBSMessage sends an event to the Event Broker Service and waits for a reply. By default, if a timeout or an error is reported by the Event Broker Service, might occur, it is logged and the workflow resumes.

If a timeout or an error occurs during the following states in the master workflow, the workflow is forced into the error state rather than resuming the workflow.

Table 7-16. Exceptions Where Workflows Do Not Resume

State Where Error Occurs	Error State
PRE MachineProvisioned	UnprovisionMachine
PRE BuildingMachine	Disposing
PRE RegisterMachine	Finalized

To customize the timeout or error behavior, you can add custom properties to the machine for any events or states where you want to trigger an event or force a state change. Use the following examples to configure the custom properties.

- `Extensibility.Lifecycle.Error.Event.{Workflow}.{State}`. The value of the property is the name of the event to be triggered on in the workflow in case of timeout or error.

- `Extensibility.Lifecycle.Error.State.{Workflow}.{State}`. The value of the property is the name of the state in which the workflow will forcibly transition to in case of timeout or error.

Scenario: Take a Post-Provisioning Snapshot of a Virtual Machine

As a tenant administrator, you want your service catalog users to have a post-provisioning snapshot of their virtual machines so that they can revert to the fresh machine rather than requesting a new one.

Procedure

1 Scenario: Create a vRealize Orchestrator Workflow for a Post-Provisioning Snapshot Action

You create a vRealize Orchestrator workflow that accepts the required input parameter. You design the workflow to accomplish your post-provisioning goal.

2 Scenario: Create a Post-Provisioning Snapshot Workflow Subscription

As a tenant administrator, you want to create a snapshot of each virtual machine after it is created. You configure a workflow subscription based on the machine provisioning event topic, and publish it to make it active.

Scenario: Create a vRealize Orchestrator Workflow for a Post-Provisioning Snapshot Action

You create a vRealize Orchestrator workflow that accepts the required input parameter. You design the workflow to accomplish your post-provisioning goal.

For information about creating vRealize Orchestrator folders and workflows, see *Developing with VMware vRealize Orchestrator*.

Prerequisites

Log in to the vRealize Orchestrator that is the instance configured for vRealize Automation with privileges that allow you to create a workflow.

Procedure

1 Create a folder for your workflow subscription workflows in the workflow library.

2 Create a new workflow.

For this scenario, name the workflow **`Automation Post-Provisioning Snapshot`**.

3 Add the following input parameter.

Name	Type
payload	Properties

4 Add a scriptable task that accepts the input parameter and creates a virtual machine snapshot.

5 Save the workflow.

What to do next

You create a workflow subscription that runs your Automation Post-Provisioning Snapshot workflow. Scenario: Create a Post-Provisioning Snapshot Workflow Subscription.

Scenario: Create a Post-Provisioning Snapshot Workflow Subscription

As a tenant administrator, you want to create a snapshot of each virtual machine after it is created. You configure a workflow subscription based on the machine provisioning event topic, and publish it to make it active.

You configure the workflow subscription to run a create snapshot workflow when a virtual machine is provisioned and the detected event message is in the activated state.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.
- Configure a vCenter Server plug-in as a vRealize Orchestrator endpoint. See Configure the vCenter Server Plug-In as an Endpoint.
- Verify that you have a vSphere virtual machine blueprint.
- Verify that you have a vRealize Orchestrator workflow that creates a snapshot of a virtual machine. You cannot use the Create a snapshot workflow provided by the vRealize Automation plug-in. The provided snapshot workflow is specific to XaaS integration. See Configuring vRealize Orchestrator Workflows for Provisioning and Life Cycle Workflows.

Procedure

1. Select **Administration > Events > Subscriptions**
2. Click the **New** icon (✚).
3. Select **Machine provisioning**.
4. Click **Next**.
5. On the Conditions tab, configure the triggering conditions.

   a. Select **Run based on conditions**.

   b. From the **Clause** drop-down menu, select **All of the following**.

   c. Configure the following conditions:

Property	Operator	Value
Data > Machine > Machine type	Equals	Constant > Virtual Machine
Data > Lifecycle state > Lifecycle state name	Equals	Constant > VMPSMasterWorkflow32.MachineActivated
Data > Lifecycle state > State phase	Equals	Constant > POST

   d. Click **Next**.

6 On the Workflow tab, browse the Orchestrator tree and select your **Automation Post-Provisioning Snapshot** workflow.

7 Click **Next**.

8 On the Details tab, enter the **Name** and **Description**.

   In this scenario, enter `Post-Provisioning Virtual Machine Snapshot` as the name and `Create a snapshot when new virtual machine is provisioned and activated` as the description.

9 Click **Finish**.

10 Select the Post-Provisioning Virtual Machine Snapshot row and click **Publish**.

Results

The workflow subscription is active and will trigger your snapshot workflow when an event message indicates that a requested virtual machine is provisioned and activated.

What to do next

To test the workflow subscription, request a virtual machine in the service catalog. After the request indicates successful provisioning, verify that the snapshot was created.

Working with Approval Workflow Subscriptions

You create pre-approval and post-approval workflow subscriptions so that you can send an approval request to an external system for processing. The response, approved or rejected, is then processed by vRealize Automation.

Approval Service Integration

You create a pre-approval or post-approval workflow subscription that runs a custom vRealize Orchestrator workflow that processes the approval request in a system outside of vRealize Automation.

In an approval policy approval level, you can select **Use event subscription** as the approver. This level can be one of several in an approval policy. When a service catalog user requests an item to which an approval policy is applied that includes the **Use event subscription** approver, the approval service sends a message to the event broker service with the following results.

- If you have a published workflow subscription with matching criteria, vRealize Orchestrator runs your approve or reject workflow.
- If you have a published workflow subscription, but the criteria do not match, you unpublished the workflow subscription, or you do not have a published subscription, the approval level is approved and the approval process goes to the next approval level.

The approval workflow subscription receives messages from the approval service and compares the messages to the configured criteria for approval subscriptions. When it finds a message that matches the criteria, the selected vRealize Orchestrator workflow starts to run. The published event data is passed to the workflow as input and processed in the method specified in the workflow. The results of the workflow are returned to vRealize Automation and the request is processed. If approved, the next approval level is evaluated. If rejected, the request is rejected. If the approval service does not receive a reply within 24 hours, the default timeout for the approval service, the request is processed as rejected.

Configuring vRealize Orchestrator Workflows for Approval Event Topics

You must configure your custom vRealize Orchestrator workflow to support the approval message and to reply with correctly formatted information that vRealize Automation can process.

Approval Event Topic Schema

The pre-approval and post-approval event message schema includes the field names and values, the information included in the request, and information about the source of the request.

The following is the structure of the event data payload.

```
{
    fieldNames : Properties,            // Property names

    fieldValues : Properties,           // Property values

    // Information about the request
    requestInfo : {
        requestRef : STRING,            // Identifier for the source request
        itemName : STRING,              // Name of the requested item
        itemDescription : STRING,       // Description of the requested item
        reason : STRING,                // Justification provided by the user specifying why the 
request is required
        description : STRING,           // Description entered by the user specifying the purpose of 
the request
        approvalLevel:ExternalReference,// Approval level ID. This is a searchable field
        approvalLevelName : STRING,     // Approval level name
        createDate : DATE_TIME,         // Time the approval request is created
        requestedFor : STRING,          // Principal id of the user for whom the source request is 
initiated
        subtenantId : STRING,           // Business group id
        requestedBy : STRING            // Principal id of the user who actually submits the request
    },

    // Information about the source of the request
    sourceInfo : {
        externalInstanceId : STRING,    // Identifier of the source object, as defined by the 
intiator service
        serviceId : STRING,             // Identifier of the service which initiated the approval
        externalClassId : STRING        // Identifier of the class to which the source object belongs
    }
}
```

Property names and property values can be the custom properties or system properties that you configure in the approval policy level. These properties are configured in the approval policy to allow the approver to change the values during an approval process. For example, if CPU is included, the approver can decrease the number of CPUs in the approval request form.

The reply event data payload is the information that it returned to vRealize Automation by the workflow. The contents of the reply payload determines whether the request is approved or rejected.

```
{
    approved : BOOLEAN,

    // Property values
    fieldValues : Properties
}
```

The approved parameter in the reply event payload is either true, for approved, or false, for rejected requests. The property values are the custom or system properties that were modified by the vRealize Orchestrator workflow and returned to vRealize Automation and included in the approval process.

As a best practice, you should configure the vRealize Orchestrator workflow with an output parameter for `businessJustification`. You can use this parameter to pass comments provided by the approver in the outside system to the vRealize Automation approval process. These comments can be for approvals or rejections.

Creating a vRealize Orchestrator Workflow Based on the Approval Schema

The custom approval workflow that you create must have an input parameter, with any useful name, that is configured with the type `Properties`. The approval event data payload is put into this parameter when the workflow subscription is triggered to run.

The output parameters of the workflow that are sent back to vRealize Automation are `approved : Boolean` and `fieldValues : Properties`. The returned `approved : Boolean` parameter determines if the approval level is approved or rejected. The `fieldValues : Properties` parameter contains the values that were modified in the external system.

Scenario: Send Software Requests to an External System for Approval

As a tenant administrator, you want users outside vRealize Automation to approve a software component when a service catalog user requests a machine that includes software. You configure an approval policy that requires approval for all software provisioning and a workflow subscription that is configured to run when it receives pre-approval messages that match your defined conditions.

Procedure

1 Scenario: Create a vRealize Orchestrator Workflow for Approval Workflow Subscriptions

   You create a vRealize Orchestrator workflow that accepts the required approval input parameters from vRealize Automation and returns the necessary output parameters to complete the approval process.

2 Scenario: Create an Approval Policy for External Approval

   As a tenant administrator, you create an approval policy that generates an event message that is published by the approval service. If you configured a workflow subscription with criteria that match the event message, vRealize Orchestrator runs the selected workflow.

3 Scenario: Create a Pre-Approval Workflow Subscription

   As a tenant administrator, you create a pre-approval workflow subscription that runs a vRealize Orchestrator workflow when a service catalog request generates an approval request that matches the configured conditions.

Scenario: Create a vRealize Orchestrator Workflow for Approval Workflow Subscriptions

You create a vRealize Orchestrator workflow that accepts the required approval input parameters from vRealize Automation and returns the necessary output parameters to complete the approval process.

You must design the workflow to accomplish your approval goal. For information about creating vRealize Orchestrator folders and workflows, see *Developing with VMware vRealize Orchestrator*.

Prerequisites

Log in to the vRealize Orchestrator that is the instance configured for vRealize Automation with privileges that allow you to create a workflow.

Procedure

1 Create a folder for your workflow subscription workflows in the workflow library.

2 Create a new workflow.

For this scenario, name the workflow **`Automation Approval Request`**.

a Add the following input parameter.

Name	Type
input	Properties

b Add the following output parameters.

Name	Type
approved	boolean
fieldValues	Properties

3 Create a scriptable task that processes the input and output parameters.

4 Save the workflow.

What to do next

You create an approval policy that uses the workflow subscription as an approver. Scenario: Create an Approval Policy for External Approval

Scenario: Create an Approval Policy for External Approval

As a tenant administrator, you create an approval policy that generates an event message that is published by the approval service. If you configured a workflow subscription with criteria that match the event message, vRealize Orchestrator runs the selected workflow.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **approval administrator**.

Procedure

1 Select **Administration > Approval Policies**.

2 Create an approval policy for your software components.

a Click the **New** icon (✚).

b Select **Select an approval policy type**.

c In the list, select **Service Catalog - Catalog Item Request - Software Component**.

   d Click **OK**.

   e Configure the following options:

Option	Configuration
Name	Enter `Software external approval`.
Description	Enter `Approval request sent to external approval system`.
Status	Select **Active**.

3 On the **Pre Approval** tab, click the **Add** icon (✚).

4 Configure the **Level Information** tab with the triggering criteria and the approval actions.

   a In the **Name** text box, enter `External level for software`.

   b In the **Description** text box, enter `Software approval request sent to external approval system`.

   c Select **Always required**.

   d Select **Use event subscription**.

5 Click **OK**.

What to do next

- Create a pre-approval workflow subscription that receives event messages based on the configured approval level. See Scenario: Create a Pre-Approval Workflow Subscription.
- Apply the approval policy to a software component in an entitlement. See Entitle Users to Services, Catalog Items, and Actions.

Scenario: Create a Pre-Approval Workflow Subscription

As a tenant administrator, you create a pre-approval workflow subscription that runs a vRealize Orchestrator workflow when a service catalog request generates an approval request that matches the configured conditions.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator**.
- Configure an approval policy level named External level for software. See Scenario: Create an Approval Policy for External Approval.
- Create a custom vRealize Orchestrator workflow that sends the request to your external system. In this scenario, use the Automation Approval Request workflow.

Procedure

1 Select **Administration** > **Events** > **Subscriptions**

2 Click the **New** icon (✚).

3 Click **Pre Approval**.

4 Click **Next**.

5 On the **Conditions** tab, configure the triggering conditions.

   a Select **Run based on conditions**.

   b From the **Clause** drop-down menu, configure the following condition:

Property	Operator	Value
Data > Information about the request > Approval level name	Equals	External level for software

   c Click **Next**.

6 On the Workflow tab, browse the Orchestrator tree and select your **Automation Approval Request** workflow.

7 Click **Next**.

8 On the Details tab, enter the name and description.

   In this scenario, enter `Software External` as the name and **Sends approval request to external system** as the description.

9 In the **Timeout (min)** text box, enter 120.

   The amount of time you specify until the subscription workflow times out depends on the target system. If vRealize Automation does not process a reply from the target system in the specified number of minutes, the request is automatically rejected.

   If you do not provide a value, the default timeout is 24 hours.

10 Click **Finish**.

11 Select the Software External row and click **Publish**.

Results

The Software External pre-approval event subscription now receives pre-approval event messages.

What to do next

- If you applied the approval policy to a software component in an active entitlement, request the item in the service catalog and verify that your approval policy and workflow subscription work as designed.

Troubleshooting Workflow Subscriptions

Troubleshooting workflow subscriptions includes some common problems. You might also need to examine various logs.

- Troubleshooting vRealize Orchestrator Workflows That Do Not Start

  You configure a workflow subscription to run a custom workflow when the event message is received, but the workflow does not run.

- Troubleshooting Provisioning Requests That Take Too Much Time

  An IaaS machine take ten or more hours to provision.

- Troubleshooting a vRealize Orchestrator Workflow That Does Not Run for an Approval Request

  You configured a pre-approval or post-approval workflow subscription to run a vRealize Orchestrator workflow. The workflow does not run when a machine that matches the defined criteria is requested in the service catalog.

- Troubleshooting a Rejected Approval Request That Should Be Approved

  You configure a pre-approval or post-approval workflow subscription that runs the specified vRealize Orchestrator workflow, but the request is rejected when you know that it was approved.

- Troubleshooting a Rejected Approval Request

  You configure a pre-approval or post-approval workflow subscription that runs the specified vRealize Orchestrator workflow, but the request is unexpectedly rejected.

Troubleshooting vRealize Orchestrator Workflows That Do Not Start

You configure a workflow subscription to run a custom workflow when the event message is received, but the workflow does not run.

Solution

1. Verify that you published the workflow subscription.
2. Verify that the workflow subscription conditions are configured correctly.
3. Verify that the vRealize Orchestrator server has the specified workflow.

Troubleshooting Provisioning Requests That Take Too Much Time

An IaaS machine take ten or more hours to provision.

Cause

If you configured a workflow subscription to trigger on a provisioning state, you might have two instances of the IaaS manager service running in your environment.

Solution

- Verify that only one instance of the IaaS manager service is active. If you have more than one instance active, you will also see similar errors in the logs:

  ```
  [EventBrokerService] Failed resuming workflow b6e9276a-f20f-40f1-99ad-6d9524560cc2 on queue
  3679fa71-ac2a-42d5-8626-f98ea096f0d3.
  ```

```
System.Workflow.Runtime.QueueException: Event Queue operation failed with MessageQueueErrorCode
QueueNotFound for queue '3679fa71-ac2a-42d5-8626-f98ea096f0d3'.
   at System.Workflow.Runtime.WorkflowQueuingService.EnqueueEvent(IComparable queueName, Object
item)
   at System.Workflow.Runtime.WorkflowExecutor.EnqueueItem(IComparable queueName, Object item,
IPendingWork pendingWork, Object workItem)
   at System.Workflow.Runtime.WorkflowInstance.EnqueueItem(IComparable queueName, Object item,
IPendingWork pendingWork, Object workItem)
   at DynamicOps.VMPS.Service.Workflow.Services.EventBrokerService.OnMessage(EventObject obj)
[UTC:2015-11-14 07:14:25 Local:2015-11-13 23:14:25] [Error]: Thread-Id="15" - context="HKBsp6Tt"
token="JeuTG7ru" [EventBrokerClient] Invoking subscription callback failed: Event Queue operation
failed with MessageQueueErrorCode QueueNotFound for queue '3679fa71-ac2a-42d5-8626-f98ea096f0d3'.
```

Troubleshooting a vRealize Orchestrator Workflow That Does Not Run for an Approval Request

You configured a pre-approval or post-approval workflow subscription to run a vRealize Orchestrator workflow. The workflow does not run when a machine that matches the defined criteria is requested in the service catalog.

Cause

To successfully run a workflow subscription for an approval, you must ensure that all the components are configured correctly.

Solution

1. Verify that the approval policy is active and that you selected **Use event subscription** for an approval level in the policy.
2. Verify that the approval policy is correctly applied in your entitlement.
3. Verify that your workflow subscription is correctly configured and published.
4. Review the event logs for messages related to approvals.

Troubleshooting a Rejected Approval Request That Should Be Approved

You configure a pre-approval or post-approval workflow subscription that runs the specified vRealize Orchestrator workflow, but the request is rejected when you know that it was approved.

Solution

1. Review the workflow in vRealize Orchestrator.
   a. Log in to vRealize Orchestrator as with administrator privileges.
   b. Verify that the workflow ran without errors.
   c. Verify that the expected values were returned for the `approval` and `fieldValues` parameters.

2 Review the request in vRealize Automation.

   a Log in to vRealize Automation as the user who requested the rejected item.

   b Click the **Deployment** tab.

   c Click the rejected request deployment name and click **History**.

   d Click the status and review the signpost for more information.

   If an error occurred, information about the error is displayed as Justification data.

Troubleshooting a Rejected Approval Request

You configure a pre-approval or post-approval workflow subscription that runs the specified vRealize Orchestrator workflow, but the request is unexpectedly rejected.

Problem

All the approval levels prior to this external approval level were approved, and this level should have been approved, but was processed as rejected.

Cause

One possible cause is an internal error when vRealize Orchestrator tried to run the workflow. For example, the workflow is missing or the vRealize Orchestrator server is not running.

Solution

1 Select **Administration** > **Events** > **Event Logs**.

2 Review the logs for messages related to approvals.

Extending Machine Life Cycles By Using vRealize Automation Designer

You can inject custom logic into predetermined stages of the IaaS machine life cycle by using vRealize Automation Designer to directly edit the state change workflow stubs and, optionally, call out to custom vRealize Orchestrator workflows.

Note The workflow stubs are replaced by the event broker workflow subscriptions. They are still available, supported, and they can be used, but expect them to be removed in a future version of vRealize Automation. To ensure future product compatibility, you should use the workflow subscriptions to run custom workflows based on state changes. See Configuring Workflow Subscriptions to Extend vRealize Automation.

Extending Machine Life Cycles By Using vRealize Automation Designer Checklist

The Extending Machine Life Cycles By Using vRealize Automation Designer Checklist provides a high-level overview of the steps required to install and configure vRealize Automation Designer to customize IaaS machine life cycles.

Table 7-17. Extending Machine Lifecycles By Using vRealize Automation Designer Checklist

Task	Details
❑ Download and install vRealize Automation Designer.	Installing vRealize Automation Designer
❑ Create a vRealize Automation endpoint for your vRealize Orchestrator instance.	Create a vRealize Orchestrator Endpoint
❑ Associate your vRealize Orchestrator endpoint with a machine blueprint.	Associate a vRealize Orchestrator Endpoint with a Blueprint
❑ Using vRealize Automation Designer activities, customize an IaaS Workflow stub. **Note** The workflow stubs are replaced by the event broker workflow subscriptions. They are still available, supported, and they can be used, but expect them to be removed in a future version of vRealize Automation. To ensure future product compatibility, you should use the workflow subscriptions to run custom workflows based on state changes. See Configuring Workflow Subscriptions to Extend vRealize Automation. Optionally, you can use vRealize Orchestrator workflow activities to call out to custom vRealize Orchestrator workflows.	Customize an IaaS Workflow
❑ After you create a custom state change workflow, a tenant administrator or business group manager must enable it for specific blueprints by adding a custom property.	Configure a Blueprint to Call a State Change Workflow

Installing and Configuring vRealize Automation Designer

You can install vRealize Automation Designer on a Windows machine and configure it to communicate with a remote Model Manager instance. If you are using IaaS workflows to call vRealize Orchestrator workflows, you must also configure the vRealize Orchestrator instance in IaaS.

Installing vRealize Automation Designer

You can install vRealize Automation Designer on a Windows machine and configure it to communicate with a remote Model Manager instance.

vRealize Automation Designer Prerequisites

vRealize Automation Designer is typically installed on a development machine rather than a server.

Supported Operating Systems

Supported operating systems for vRealize Automation Designer are listed in the *vRealize Automation Support Matrix* on the VMware vRealize Automation Documentation page.

System Configuration Requirements

See the *vRealize Automation Support Matrix* for your vRealize Automation version for potential updates to this information.

- .NET Framework 4.5 must be installed.
- The vRealize Automation Designer host must have network access to the IaaS Website components (specifically, the Model Manager Web component).
- If the Model Manager is installed remotely, the certificate used for the Model Manager Web component must be trusted on the vRealize Automation Designer host.

Download the vRealize Automation Designer Installer

You can download the vRealize Automation Designer installer from the vRealize Automation appliance.

Prerequisites

- Log in to the Windows machine as a local administrator.
- If you are using Internet Explorer, verify that Enhanced Security Configuration is not enabled. See `res://iesetup.dll/SoftAdmin.htm`.

Procedure

1. Open a browser.
2. Navigate to the Windows installer download page by using the host name of the (https://*vra-va-hostname.domain.name*:5480/installer/).
3. Click **vRealize Automation Designer**.
4. When prompted, save the installer.

What to do next

Install vRealize Automation Designer.

Install vRealize Automation Designer

The vRealize Automation Designer installer is packaged as Windows installation wizard.

Prerequisites

Download the vRealize Automation Designer Installer.

Procedure

1. Navigate to the directory where you downloaded the installer.
2. Right-click `DesignCenter-Setup.exe` and select **Run as administrator**.
3. On the **Welcome** page, click **Next**.

4. Read the License Agreement, select **I accept the terms in the License Agreement**, and click **Next**.
5. On the **Custom Setup** page, click **Next**.
6. Specify the fully qualified domain name and port of the Model Manager Web instance in *hostname:port* format.

   The default port is 443.
7. Specify the Model Manager service user credentials.
8. Click **Next**.

   The installer validates the combination of Model Manager host and credentials by attempting to access to the Model Manager. If an error is returned, you must provide the correct combination of Model Manager host and credentials before proceeding.
9. Click **Install**.
10. Click **Finish**.

What to do next

You can launch the vRealize Automation Designer from the Windows Start menu by navigating to the installation directory.

Configuring vRealize Orchestrator Endpoints

If you are using vRealize Automation workflows to call vRealize Orchestrator workflows, you must configure the vRealize Orchestrator instance or server as an endpoint.

You can associate a vRealize Orchestrator endpoint with a machine blueprint to make sure that all of the vRealize Orchestrator workflows for machines provisioned from that blueprint are run using that endpoint.

vRealize Automation by default includes an embedded vRealize Orchestrator instance. It is recommended that you use the embedded instance as your vRealize Orchestrator endpoint for running vRealize Automation workflows in a production or test environment or when creating a proof of concept.

You can also install a plug-in on an external vRealize Orchestrator server, although this method is not recommended for production.

vRealize Orchestrator Integration Prerequisites

If you are using vRealize Automation workflows to run vRealize Orchestrator workflows that have input or output parameters of type `VC:VirtualMachine`, verify that you have the vRealize Orchestrator workflows for converting virtual machine types between vRealize Orchestrator and IaaS.

The required workflows are included by default in vRealize Orchestrator 5.5 and later as part of the vCenter plug-in.

If you are using vRealize Orchestrator 5.1, install the vRealize Automation integration package for vRealize Orchestrator. Download the package `com.vmware.library.vcenter.vcac-integration.package` from the vRealize Orchestrator community site at https://communities.vmware.com/t5/vRealize-Orchestrator-Documents/vCloud-Automation-Center-integration-package/ta-p/2777982. Import the package on each vRealize Orchestrator server that you set up as an endpoint in IaaS.

For information about importing packages to vRealize Orchestrator, refer to the vRealize Orchestrator documentation.

Create a vRealize Orchestrator Endpoint

You can create a vRealize Orchestrator endpoint to connect to a vRealize Orchestrator server.

You can configure multiple endpoints to connect to different vRealize Orchestrator servers, but you must configure a priority for each endpoint.

When executing vRealize Orchestrator workflows, vRealize Automation tries the highest priority vRealize Orchestrator endpoint first. If that endpoint is not reachable, then it proceeds to try the next highest priority endpoint until a vRealize Orchestrator server is available to run the workflow.

Prerequisites

- Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1 Select **Infrastructure > Endpoints > Endpoints**.

2 Select **New > Orchestration > vRealize Orchestrator**.

3 Enter a name and, optionally, a description.

4 Enter a URL with the fully qualified name or IP address of the vRealize Orchestrator server and the vRealize Orchestrator port number.

   The transport protocol must be HTTPS. If no port is specified, the default port 443 is used.

   To use the default vRealize Orchestrator instance embedded in the vRealize Automation appliance, type **`https://`*`vrealize-automation-appliance-hostname`*`:443/vco`**.

5 Provide your vRealize Orchestrator credentials in the **User name** and **Password** text boxes to connect to the vRealize Orchestrator endpoint.

   The credentials you use should have Execute permissions for any vRealize Orchestrator workflows to call from IaaS.

   To use the default vRealize Orchestrator instance embedded in the vRealize Automation appliance, the user name is **`administrator@vsphere.local`** and the password is the administrator password that was specified when configuring SSO.

6 Enter an integer greater than or equal to 1 in **Priority** text box.

   A lower value specifies a higher priority.

7. (Optional) Click **Properties** and add supplied custom properties, property groups, or your own property definitions for the endpoint.
8. Click **OK**.

Associate a vRealize Orchestrator Endpoint with a Blueprint

You can specify a particular vRealize Orchestrator endpoint to use with a blueprint.

When IaaS runs a vRealize Orchestrator workflow for any machine provisioned from this blueprint, it always uses the associated endpoint. If the endpoint is not reachable, the workflow fails.

Prerequisites

Log in to vRealize Automation as an **infrastructure architect**.

Procedure

1. Select **Design > Blueprints**.
2. Create a blueprint or edit an existing blueprint.

   If you are editing an existing blueprint, the vRealize Orchestrator endpoint you specify only applies to new machines provisioned from the updated blueprint. Existing machines provisioned from the blueprint continue to use the highest priority endpoint unless you manually add this property to the machine.
3. Click the **Blueprint Properties** icon (⚙).
4. Click **Properties** tab.
   1. Click **Custom Property > New**.
   2. Type `VMware.VCenterOrchestrator.EndpointName` in the **Name** text box.

      The property name is case-sensitive.
   3. Click **OK** to save your property.
5. Click **OK**.

Customizing IaaS Workflows By Using vRealize Automation Designer

VMware provides a number of workflows that you can customize using the vRealize Automation Designer. These include state change workflows and menu operation workflows.

IaaS workflows are created using Microsoft Windows Workflow Foundation 4, part of .NET Framework 4. For information on Windows Workflow Foundation and workflow creation, refer to the Microsoft documentation. vRealize Automation also provides several vRealize Automation Designer activities for running and monitoring vRealize Orchestrator workflows.

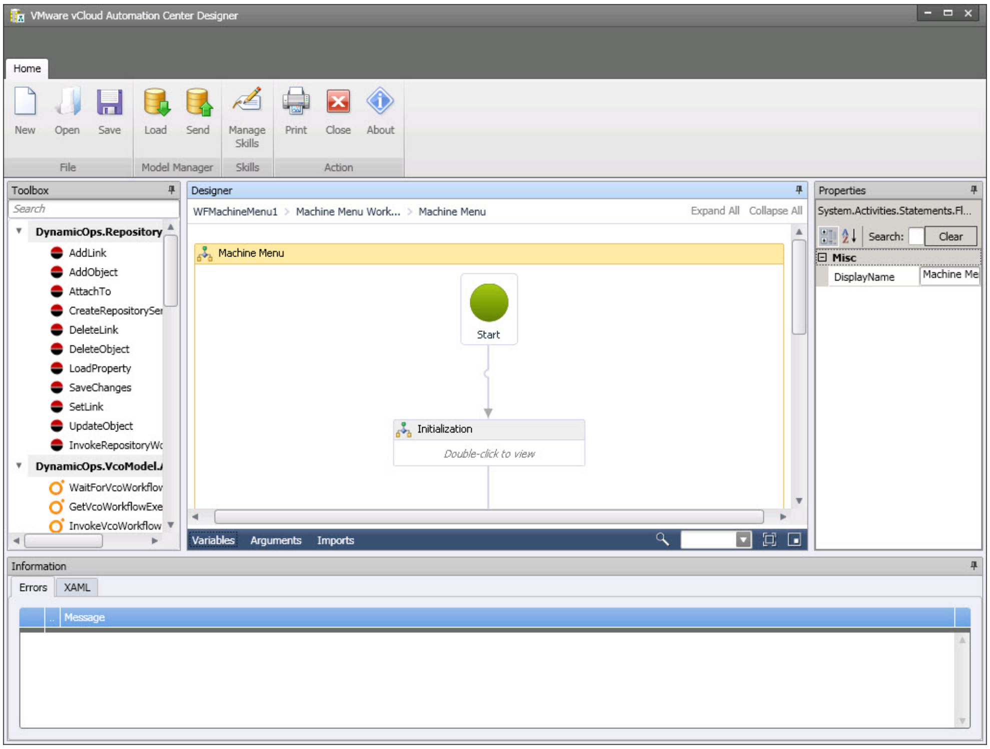
The customizable workflow templates provided by VMware demonstrate best practices for structuring workflows with separate sequences for initialization, custom logic, and finalization. The entire workflow is wrapped in a TryCatch block for error handling. Any uncaught or rethrown exceptions are logged by the Distributed Execution Manager that executes the workflow.

After you create a custom IaaS workflow, a blueprint author must enable the workflow on specific blueprints.

The vRealize Automation Designer Console

The vRealize Automation Designer console provides a visual workflow editor for customizing IaaS workflows.

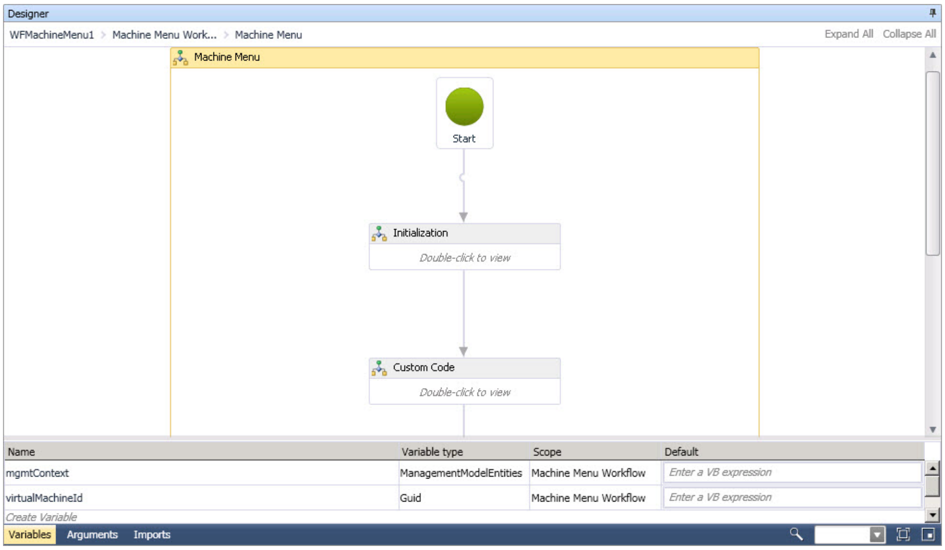
You must have local administrator rights on the vRealize Automation Designer host (typically a development machine) in order to launch the vRealize Automation Designer console.



The Toolbox pane on the left provides access to the vRealize Automation workflow activity library. You can drag activities from the toolbox onto the Designer pane to add them to a workflow. The Properties pane displays the configurable properties of the currently selected activity on the Designer pane. This interface is very similar to the workflow designer in Visual Studio.

The detail tabs at the bottom of the Designer pane enable you to display and edit variables within the scope of the selected activity or arguments to the selected activity.

Note Variables and arguments are both specified as Visual Basic expressions. However, variable names are not case sensitive while argument names are case sensitive. For information about valid arguments for the IaaS workflow activities, see vRealize Automation Workflow Activity Reference.



Name	Variable type	Scope	Default
mgmtContext	ManagementModelEntities	Machine Menu Workflow	*Enter a VB expression*
virtualMachineId	Guid	Machine Menu Workflow	*Enter a VB expression*
Create Variable			

The Imports tab displays imported namespaces from which you can select entity types to add to the workflow.

The collapsible Information pane at the bottom of the console displays any errors in configuring activities and provides access to the XAML representation of the workflow.

IaaS Workflow Types

You can customize two types of workflows by using vRealize Automation Designer: state change workflows and menu operation workflows.

- A state change workflow is run when the main workflow transitions between states, for example at a particular stage during the process of provisioning a new machine.
- A menu operation workflow is run when a user selects an option from the Action menu in the service catalog or from the machine menu in the Infrastructure tab.

State Change Workflows

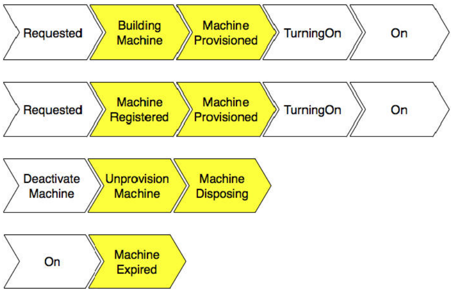
Creating a state change workflow enables you to run a workflow before the IaaS main workflow enters a specific state.

For example, you can create custom workflows to integrate with an external database and record information at different stages of the machine life cycle:

- Create a custom workflow that runs before the main workflow enters the MachineProvisioned state to record such information as machine owner, approvers and so on.

- Create a custom workflow that runs before a machine enters the MachineDisposing state to record the time at which the machine was destroyed and data such as its resource utilization at last data collection, last login, and so on.

The following illustrations show the primary states of the main workflow.



vRealize Automation Designer provides a customizable workflow for each of these states.

Table 7-18. Customizable State Change Workflows

Main Workflow State	Customizable Workflow Name
BuildingMachine	WFStubBuildingMachine
Disposing	WFStubMachineDisposing
Expired	WFStubMachineExpired
MachineProvisioned	WFStubMachineProvisioned
RegisterMachine	WFStubMachineRegistered
UnprovisionMachine	WFStubUnprovisionMachine

Configuring a State Change Workflow Overview

You can customize a state change workflow by using vRealize Automation Designer. A blueprint author can then enable it for specific blueprints.

The following is a high-level overview of the steps required to enable state change workflows:

1 A workflow developer customizes one of the state change workflow templates by using vRealize Automation Designer. See Customize an IaaS Workflow.

  Any IaaS workflow can call a vRealize Orchestrator workflow. For more information, see Using vRealize Orchestrator Workflow Activities.

2 A tenant administrator or business group manager configures a blueprint to call the customized workflow for machines provisioned from that blueprint. See Configure a Blueprint to Call a State Change Workflow.

Menu Operation Workflows

A menu operation workflow is executed when a user selects an option from the Actions menu in the service catalog or the machine menu in the Infrastructure tab.

For example, you can create a custom workflow that enables a user to create a support ticket related to a machine by selecting Raise Support Issue from the machine menu.

vRealize Automation Designer provides templates for customizing menu operation workflows.

In addition to the workflow definition, a menu operation workflow depends on an operation configuration file, which defines the aspects of the custom menu option such as the display text, which roles have access to it, and the machine states for which the operation is available.

Note An XaaS architect can define custom actions for any catalog item by using the XaaS. Creating custom actions for IaaS machines other than those provisioned by using vSphere or vCloud Director requires vRealize Automation 6.1 or later.

Configuring a Menu Operation Workflow Overview

You can customize a menu operation workflow by using vRealize Automation Designer and the CloudUtil command-line utility. A blueprint author can then enable it for specific blueprints.

The following is a high-level overview of the steps required to enable menu operation workflows:

1 A workflow developer customizes one of the menu operation workflow templates by using vRealize Automation Designer. See Customize an IaaS Workflow.

  Any IaaS workflow can call a vRealize Orchestrator workflow. For more information, see Using vRealize Orchestrator Workflow Activities.

2 A workflow developer configures the menu operation in the Model Manager. See Configure a Menu Operation.

3 A workflow developer registers the new menu operation with the service catalog. See Register New Menu Operations with the Service Catalog.

4 A tenant administrator or business group manager configures a blueprint to enable the menu operation for machines provisioned from that blueprint. See Configure a Blueprint to Enable a Menu Operation Workflow.

If the menu operation is intended to be used in the service catalog, it must also be entitled to users. For more information, see *Tenant Administration*.

Customize an IaaS Workflow

vRealize Automation Designer enables you to edit the customizable workflows and update workflows in the Model Manager.

Prerequisites

Launch the vRealize Automation Designer.

Procedure

1 Click **Load**.

2 Select the workflow that you want to customize.

Option	Description
WFMachineMenu*N*	Customizable menu operation workflow
WFStubBuildingMachine	Customizable state change workflow that executes before a machine enters the BuildingMachine state
WFStubMachineDisposing	Customizable state change workflow that executes before a machine enters the Disposing state
WFStubMachineExpired	Customizable state change workflow that executes before a machine enters the Expired state
WFStubMachineProvisioned	Customizable state change workflow that executes before a machine enters the MachineProvisioned state
WFStubMachineRegistered	Customizable state change workflow that executes before a machine enters the RegisterMachine state
WFStubUnprovisionMachine	Customizable state change workflow that executes before a machine enters the UnprovisionMachine state

3 Click **OK**.

The workflow displays in the Designer pane.

4 Customize the workflow by dragging activities from the Toolbox to the Designer pane and configuring their arguments.

5 When you are finished editing the workflow, update the workflow in the Model Manager by clicking **Send**.

The workflow is saved and appears as a new revision in the list the next time you load a workflow. You can access an earlier version of a workflow at any time. See Revert to a Previous Revision of a Workflow.

Using vRealize Orchestrator Workflow Activities

You can use vRealize Automation Designer activities to call vRealize Orchestrator workflows either synchronously or asynchronously.

A vRealize Orchestrator endpoint is specified in one of the following ways:

- `VirtualMachineId` is the name of the variable representing the virtual machine ID. A virtual machine with this ID is selected and the value that is retrieved from the `VMware.VCenterOrchestrator.EndpointName` custom property for a virtual machine is used as the vRealize Orchestrator endpoint name.
- `GetVcoEndpointByManagementEndpoint` returns the value of a custom property on a specified `ManagementEndpoint` object. If the `CustomPropertyName` is not specified, the value of the `VMware.VCenterOrchestrator.EndpointName` property is used.
- `GetVcoEndpointByHost` returns the value of a custom property on a specified host. If the `CustomPropertyName` is not specified, the value of the `VMware.VCenterOrchestrator.EndpointName` property is used.

Synchronous Execution

The `InvokeVcoWorkflow` activity calls a vRealize Orchestrator workflow and blocks further running of its parent IaaS workflow until the vRealize Orchestrator workflow completes. The activity returns the output parameters for the vRealize Orchestrator workflow.

In addition, the synchronous running supports the following property:

- `WorkflowTimeout` is a timeout value in seconds. If the vRealize Orchestrator workflow does not finish in the specified time, an exception is generated rather than blocking the workflow until a response is returned. If no value is defined or a value of zero is supplied, the timeout is not activated. The workflow status is checked every 10 seconds during that period unless the polling time is modified for the endpoint by specifying a value in the `VMware.VCenterOrchestrator.PollingInterval` custom property.

Asynchronous Running of Workflows

The `InvokeVcoWorkflowAsync` activity is an activity that calls a vRealize Orchestrator workflow and continues to run activities in the IaaS workflow without waiting for the vRealize Orchestrator workflow to complete.

The activity returns either a unique workflow token that can be used to monitor the workflow or an error if the REST API call to the vRealize Orchestrator server failed (for example, if the server could not be reached).

Two additional activities are available for use with this activity:

- `GetVcoWorkflowExecutionStatus` enables you to poll the vRealize Orchestrator workflow for its status.
- `WaitForVcoWorkflowCompletion` enables you to block further running of the IaaS workflow until the vRealize Orchestrator workflow has completed or timed out. You can use this activity to retrieve the results of a vRealize Orchestrator workflow that you run asynchronously.

Call a vRealize Orchestrator Workflow

You can use either the `InvokeVcoWorkflow` or the `InvokeVcoWorkflowAsync` activity to call a vRealize Orchestrator workflow from an IaaS workflow.

Some vRealize Orchestrator workflows require user interaction. For these workflows, the user prompt appears in the vRealize Orchestrator client rather than in the vRealize Automation console, so it is not apparent to the end user in vRealize Automation that a workflow is waiting for input.

To avoid workflows that block user input, do not call vRealize Orchestrator workflows that require user interaction from IaaS workflows.

Procedure

1. In vRealize Automation Designer, open a workflow and navigate to the context where you want to call a vRealize Orchestrator workflow.
2. Drag the `InvokeVcoWorkflow` or the `InvokeVcoWorkflowAsync` activity into the Designer pane.

3. Select the vCenter Orchestrator workflow to run.

   a. Under General, click the ellipsis next to Workflow.

   b. In the Browse for vCO workflow dialog box, select a workflow.

   c. Click **OK**.

   The Inputs and Outputs sections display the input and output parameters of the selected workflow.

4. In the Properties pane, specify one of the following target parameters.

   - `VirtualMachineId` is the name of the variable representing the virtual machine ID. A virtual machine with this ID is selected and the value that is retrieved from the `VMware.VCenterOrchestrator.EndpointName` custom property for a virtual machine is used as the vRealize Orchestrator endpoint name.

   - `VcoEndpointName` is the endpoint name that is used to run the workflow. If specified, this value overrides the `VirtualMachineId` value when selecting the vRealize Orchestrator endpoint.

   - `WorkflowTimeout` is a timeout value in seconds. If the vRealize Orchestrator workflow does not finish in the specified time, an exception is generated rather than blocking the workflow until a response is returned. If no value is defined or a value of zero is supplied, the timeout is not activated. The workflow status is checked every 10 seconds during that period unless the polling time is modified for the endpoint by specifying a value in the `VMware.VCenterOrchestrator.PollingInterval` custom property.

5. Specify the parameters for the vRealize Orchestrator workflow.

   - Enter the values in the activity in the Designer pane.

   - In the Properties pane, click the ellipsis next to **InputParameters** or **OutputParameters** to open the Parameters dialog box. This dialog box displays the IaaS type of each parameter. If the parameter type appears in bold, the parameter is required.

   Point to the text box for any parameter to view a tooltip indicating the vRealize Orchestrator type.

   If you are using the `InvokeVcoWorkflowAsync` activity, the output parameters of the vRealize Orchestrator workflow are displayed with their corresponding types for informational purposes, but you cannot specify an expression for the parameter in this activity.

What to do next

To retrieve the results of a workflow that you run asynchronously, use the `WaitForVcoWorkflowCompletion` activity.

Get the Status of a vRealize Orchestrator Workflow

You can check the status of a vRealize Orchestrator workflow that was called with the `InvokeVcoWorkflowAsync` activity using the `GetVcoWorkflowExecutionStatus` activity.

Prerequisites

Call a vRealize Orchestrator Workflow using the `InvokeVcoWorkflowAsync` activity.

Procedure

1. In vRealize Automation Designer, open a workflow where you have used the `InvokeVcoWorkflowAsync` activity.
2. Navigate to the context where you want to check the status of the vRealize Orchestrator workflow.
3. Drag the `GetVcoWorkflowExecutionStatus` activity into the Designer pane.
4. In the Properties pane, specify the name of the variable representing the virtual machine ID in `VirtualMachineId.`

   The customizable workflows contain a variable by default named `virtualMachineId` that is set during initialization.
5. Create a variable of type `DynamicOps.VcoModel.Common.VcoWorkflowExecutionToken`.
6. Specify the name of the token variable as the `executionToken` output parameter on the `InvokeVcoWorkflowAsync` activity.
7. Specify the same variable name as the `WorkflowExecutionToken` property of the `GetVcoWorkflowExecutionStatus` activity.
8. Create a variable of type string.
9. Specify the name of the string variable as the `VcoWorkflowExecutionStatus` property of the `GetVcoWorkflowExecutionStatus` activity.

Results

When the workflow runs, the value of the `VcoWorkflowExecutionStatus` variable is set to the status of the vRealize Orchestrator workflow.

Get the Results of a vRealize Orchestrator Workflow

If you want to call a vRealize Orchestrator workflow asynchronously and then retrieve the results of the completed workflow at a later point, you can use the `WaitForVcoWorkflowCompletion` activity.

The `WaitForVcoWorkflowCompletion` activity blocks the IaaS workflow until the vRealize Orchestrator workflow has completed or a timeout is reached. The activity returns the results of the vRealize Orchestrator workflow if it completes successfully, an error if the workflow fails, or null if the workflow times out.

Prerequisites

Call a vRealize Orchestrator Workflow using the `InvokeVcoWorkflowAsync` activity.

Procedure

1 In vRealize Automation Designer, open a workflow where you have used the `InvokeVcoWorkflowAsync` activity.

2 Navigate to the context where you want to retrieve the results of the vRealize Orchestrator workflow.

3 Drag the `WaitForVcoWorkflowCompletion` activity into the Designer pane.

4 In the Properties pane, specify the name of the variable representing the virtual machine ID in `VirtualMachineId.`

   The customizable workflows contain a variable by default named `virtualMachineId` that is set during initialization.

5 Create a variable of type DynamicOps.VcoModel.Common.VcoWorkflowExecutionToken.

6 Create a variable of type `DynamicOps.VcoModel.Common.VcoWorkflowExecutionToken`.

7 Specify the name of the token variable as the `executionToken` output parameter on the `InvokeVcoWorkflowAsync` activity.

8 Specify the same variable name as the `WorkflowExecutionToken` property of the `WaitForVcoWorkflowCompletion` activity.

9 Retrieve the output of the vRealize Orchestrator workflow.

   a Create a variable of type `DynamicOps.VcoModel.Common.VcoWorkflowExecutionResult`.

   b Specify the name of the results variable as the `WorkflowOutput` property of the `WaitForVcoWorkflowCompletion` activity.

      When the workflow runs, the value of the variable is set to the results of the vRealize Orchestrator workflow, if any.

vRealize Orchestrator and IaaS Object Types

When you use either the `InvokeVcoWorkflow` or the `InvokeVcoWorkflowAsync` activity in vRealize Automation Designer, input and output properties for the activity are automatically populated based on the parameters of the vRealize Orchestrator workflow that you select.

Basic vRealize Orchestrator object types are converted into the following IaaS object types:

Table 7-19. vRealize Orchestrator and IaaS Object Types

vRealize Orchestrator Type	IaaS Type
string	string
boolean	bool
number	decimal
SecureString	string
Text	string

Table 7-19. vRealize Orchestrator and IaaS Object Types (continued)

vRealize Orchestrator Type	IaaS Type
Array/T	Array<T>
Properties	Dictionary<string,object>
Date	DateTime
VC:VirtualMachine	VirtualMachine

Note If you are using vRealize Orchestrator 5.1, you must have installed the vRealize Automation integration package to enable the conversion of `VC:VirtualMachine` object types to `VirtualMachine`.

All other vRealize Orchestrator types are converted to the IaaS type `VcoSdkObject`.

Configure a Blueprint to Call a State Change Workflow

After you create a custom state change workflow, a tenant administrator or business group manager must enable it for specific blueprints by adding a custom property.

Each state change workflow is associated with a specific custom property. When a machine is entering a state with a corresponding state change workflow, IaaS checks to see if the machine has the corresponding custom property; if so, the associated workflow is executed. For example, if a machine has the custom property `ExternalWFStubs.MachineProvisioned`, the `WFStubMachineProvisioned` workflow is executed before the master workflow enters the MachineProvisioned state.

While custom properties can be applied to a machine from a number of sources, typically the property for a state change workflow is specified in a blueprint, enabling the workflow for all machines provisioned from that blueprint.

Prerequisites

Log in to vRealize Automation as a **tenant administrator** or **business group manager**.

Procedure

1 Select **Design > Blueprints**.

2 Point to the name of a blueprint and click **Edit**.

3 Select the **Blueprint Properties** icon (⚙).

4 Click the **Properties** tab.

5 Click **Custom Properties > New**.

6 Type the name of the custom property associated with the workflow you want to enable in the **Name** text box.

Customizable Workflow Name	Associated Property Name
WFStubMachineProvisioned	ExternalWFStubs.MachineProvisioned
WFStubBuildingMachine	ExternalWFStubs.BuildingMachine
WFStubMachineDisposing	ExternalWFStubs.MachineDisposing
WFStubUnprovisionMachine	ExternalWFStubs.UnprovisionMachine
WFStubMachineRegistered	ExternalWFStubs.MachineRegistered
WFStubMachineExpired	ExternalWFStubs.MachineExpired

7 Leave the **Value** text box blank.

The workflow depends on the presence of the property, not on any particular value.

8 Click **OK** to save your property.

9 Click **OK**.

Results

The workflow is now enabled for new machines that are provisioned from this blueprint.

Configuring a Menu Operation Workflow

After you customize a menu operation workflow, additional configuration is required before it is available to users in the vRealize Automation console.

Configure a Menu Operation

To configure a menu operation, you create an operation configuration file and install it in the Model Manager.

Procedure

1 Create an Operation Configuration File

The operation configuration file is required for menu operation workflows. It specifies the aspects of the custom menu option in the vRealize Automation console such as the display text, which roles have access to the option, and the machine states for which the option is available.

2 Install an Operation in the Model Manager

You install an operation in the Model Manager by using the CloudUtil command-line utility.

What to do next

If the menu operation is intended to be used in the service catalog, it must be registered with the service catalog so that it can be entitled to users. Register New Menu Operations with the Service Catalog.

Create an Operation Configuration File

The operation configuration file is required for menu operation workflows. It specifies the aspects of the custom menu option in the vRealize Automation console such as the display text, which roles have access to the option, and the machine states for which the option is available.

Procedure

1 Create a new XML file.

```
<?xml version="1.0" encoding="utf-8"?>
```

2 Create the root element `customOperations`.

```
<customOperations xmlns="http://www.dynamicops.com/schemas/2009/OperationConfig/">
</customOperations>
```

The element must specify the XML namespace `http://www.dynamicops.com/schemas/2009/OperationConfig/`.

3 For each operation you want to define, add an `operation` element within `customOperations`.

```
<operation name="WFMachineMenu1" displayName="Execute Machine Menu task">
</operation>
```

The `operation` element takes the following attributes:

Attribute	Description
name	The name of the workflow that this operation executes.
displayName	A descriptive label for the option in the machine menu.

4 Specify the roles to grant access to the menu operation.

   a Add the `authorizedTasks` element.

   ```
   <operation name="WFMachineMenu1" displayName="Execute Machine Menu task">
     <authorizedTasks>
     </authorizedTasks>
   </operation>
   ```

   b For each role that you want to grant access to the operation, add a `task` element, for example:

   ```
   <authorizedTasks>
     <task>VRM User Custom Event</task>
     <task>VRM Support Custom Event</task>
     <task>Group Administrator Custom Event</task>
     <task>Enterprise Administrator Custom Event</task>
     <task>VRM Administrator Custom Event</task>
   </authorizedTasks>
   ```

   The valid contents of the `task` element are as follows:

Element content	Description
VRM User Custom Event	Grants access to the operation for all users.
VRM Support Custom Event	Grants access to the operation for support users.
Group Administrator Custom Event	Grants access to the operation for business group managers.
Enterprise Administrator Custom Event	Grants access to the operation for fabric administrators.
VRM Administrator Custom Event	Grants access to the operation for IaaS administrators only.

5 (Optional) Specify the machine states for which the operation is available.

   a Add the `machineStates` element.

   ```
   <operation name="WFMachineMenu1" displayName="Execute Machine Menu task">
     <machineStates>
     </machineStates>
   </operation>
   ```

   b For each state in which the operation should be available, add a `state` element.

   ```
   <machineStates>
     <state>On</state>
     <state>Off</state>
   </machineStates>
   ```

   The value may be any of the possible machine states. For a full list of machine states, see *IaaS Configuration for Virtual Platforms*, *IaaS Configuration for Physical Machines*, or *IaaS Configuration for Cloud Platforms*.

   If the element is omitted, the operation is available for all machine states.

Example

The following is an example of a complete operation configuration file:

```
<?xml version="1.0" encoding="utf-8" ?>
<customOperations xmlns="http://www.dynamicops.com/schemas/2009/OperationConfig/">
  <operation name="WFMachineMenu1" displayName="Execute Machine Menu task">
    <authorizedTasks>
      <task>VRM User Custom Event</task>
      <task>VRM Support Custom Event</task>
      <task>Group Administrator Custom Event</task>
      <task>Enterprise Administrator Custom Event</task>
      <task>VRM Administrator Custom Event</task>
    </authorizedTasks>
    <machineStates>
      <state>On</state>
      <state>Off</state>
    </machineStates>
  </operation>
</customOperations>
```

Install an Operation in the Model Manager

You install an operation in the Model Manager by using the CloudUtil command-line utility.

Prerequisites

Create an Operation Configuration File.

Procedure

1 Open an elevated command prompt.

2 Run the CloudUtil.exe command with the following arguments.

-
  ```
  CloudUtil.exe Operation-Create -c <path to operation definition file>
  ```

- Optionally, you can specify a Model Manager host and request a stack trace in case of error.

  ```
  CloudUtil.exe Operation-Create -c <path to operation definition file>
  --repository <Model Manager Root URI> -v
  ```

What to do next

If the menu operation is intended to be used in the service catalog, it must be registered with the service catalog so that it can be entitled to users. Register New Menu Operations with the Service Catalog.

Register New Menu Operations with the Service Catalog

After installing new menu operations, the workflow developer must register them with the service catalog so they can be entitled to users.

Prerequisites

- Configure a Menu Operation.
- On the IaaS Model Manager host, log in to Windows as a local user with **administrator** privileges.

Procedure

1. Open an elevated command prompt.
2. Navigate to the IaaS root installation directory.

   In a typical installation, this is `C:\Program Files (x86)\VMware\vCAC`.
3. Navigate to `Server\Model Manager Data\Cafe`.
4. Execute the following command:

   ```
   Vcac-Config.exe RegisterCatalogTypes -v
   ```

What to do next

A tenant administrator or business group manager must entitle the new action before it is available to users in the service catalog. For more information, see *Tenant Administration*.

Configure a Blueprint to Enable a Menu Operation Workflow

You enable a menu operation workflow for machines provisioned from a specific blueprint by updating the security configuration for the blueprint.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **business group manager**.
- A configured menu operation must exist and be registered with the Service Catalog.

Procedure

1. Select **Design > Blueprints**.
2. Point to the name of a blueprint and click **Edit**.
3. Click the **Actions** tab.
4. Select the checkbox that corresponds to the operation that you want to enable.
5. Click **OK**.

Results

The menu operation is now enabled for machines provisioned from this blueprint and available to all user roles specified in the operation configuration file.

What to do next

If the menu operation is intended to be used in the service catalog, it must also be entitled to users. For more information, see *Tenant Administration*.

Revert to a Previous Revision of a Workflow

The **Load Workflow** dialog displays all revisions of a workflow in the Model Manager so that you have access to the full version history of the workflows.

Each time you send a workflow to the Model Manager, the Revision and Time Stamp are updated.

Prerequisites

Launch the vRealize Automation Designer console.

Procedure

1 Click **Load**.

2 Select the revision of the workflow that you want to revert to.

   The original workflows provided by VMware are revision 0 (zero).

3 Click **OK**.

4 Update the workflow in the Model Manager by clicking **Send**.

Results

The earlier revision becomes the latest revision in the Model Manager. For example, if you have created revisions 1 and 2 of a workflow, then load and save revision 0, revisions 0 and 3 are now identical and you have returned the workflow to the version provided by VMware.

Workflows and Distributed Management

You can use skills to restrict running of workflows to specific Distributed Execution Managers.

A skill is similar to a tag that you can apply to both workflows and DEM Worker instances. If a workflow is not associated with any skills, any DEM Worker can run it. If a workflow is associated with one or more skills, then only DEM Workers that are associated with all of the same skills can run it.

Skills are useful when a particular workflow requires a DEM installed on a host with specific prerequisites. For example, you may want to restrict cloud provisioning workflows to a specific DEM running on a host with the required network access to Amazon URLs.

Skills can also be used to associate workflows with a particular data center location. For example, you might install one DEM in your Boston data center and another in your London data center, and use skills to direct certain operations to one data center or the other.

Associate Workflows and DEM Workers by Using Skills

You associate workflows with a specific DEM Worker or set of Worker instances by adding a skill to the Model Manager and then associating the skill with one or more workflows and DEM Workers.

Prerequisites

Launch the vRealize Automation Designer console.

Procedure

1. On the ribbon, click **Manage Skills**.
2. In the text field at the upper left of the **Manage Skills** dialog, type the name of a new skill and click the Add button.

   The skill name must be unique. If the name of the new skill matches the name of an existing skill, the Add button is unavailable.
3. Select the name of the skill in the list on the left.
4. Associate the skill with one or more DEM Workers.

   a. Click the **Add** icon (➕) next to Distributed Execution Managers.

   b. In the **Select DEMs** dialog, select one or more DEM Worker instances.

   c. Click **OK**.
5. Associate the skill with one or more Workflows.

   a. Click the **Add** icon (➕) next to Workflows.

   b. In the **Select Workflows** dialog, select one or more workflows.

   c. Click **OK**.

   The workflows associated with this skill can only be executed by the DEM Workers that are associated with this skill.
6. When you are done adding skills and associating them with DEM workers and workflows, click **OK** to close the **Manage Skills** dialog and save your changes to the Model Manager.

Remove Associations between Skills and DEM Workers

When you remove the association between a skill and a DEM Worker, that Worker instance can no longer execute the workflows associated with the skill.

Prerequisites

Launch the vRealize Automation Designer console.

Procedure

1. On the ribbon, click **Manage Skills**.
2. In the **Manage Skills** dialog, select the name of the skill in the list on the left.
3. Select the name of one or more DEM Worker instances from the Distributed Execution Managers list and click the **Remove** icon (➖).
4. Click **OK** to close the **Manage Skills** dialog and save your changes to the Model Manager.

Remove Associations between Skills and Workflows

When you remove the association between a skill and a workflow, that workflow is no longer restricted to the DEM Workers that are associated with the same skill.

Prerequisites

Launch the vRealize Automation Designer console.

Procedure

1. On the ribbon, click **Manage Skills**.
2. In the **Manage Skills** dialog, select the name of the skill in the list on the left.
3. Select the name of one or more workflows from the Workflows list and click the **Remove** icon (−).
4. Click **OK** to close the **Manage Skills** dialog and save your changes to the Model Manager.

Remove a Skill

Removing a skill also removes its associations to any DEM Workers and workflows.

Prerequisites

Launch the vRealize Automation Designer console.

Procedure

1. On the ribbon, click **Manage Skills**.
2. In the **Manage Skills** dialog, select the name of the skill in the list on the left.
3. Click the **Remove** icon (−) at the top of the list of skills.

   After you confirm that you want to delete the skill, its name appears dimmed to indicate that it is marked for deletion.
4. Click **OK** to close the **Manage Skills** dialog and save your changes to the Model Manager or **Cancel** if you do not want to delete the skill and its associations with DEMs and workflows.

CloudUtil Command Reference

This section provides a reference to the commands in the CloudUtil command line interface.

CloudUtil is the command line interface for the vRealize Automation Designer. You run the commands on the Windows machine on which you are running the designer. The default installation location on the Windows machine is `C:\Program Files (x86)\VMware\vCAC\Design Center`.

Note In the CloudUtil commands, the Model Manager is referred to as the `repository` and a Distributed Execution Manager (DEM) is referred to as an `agent`.

DEM Commands

The DEM commands enable you to view a list of Distributed Execution Managers registered with the Model Manager and add or remove associations between skills and DEMs.

DEM-Add-Skills

Associates skills with a registered Distributed Execution Manager.

Synopsis

```
CloudUtil.exe DEM-Add-Skills -n|--name <Name> -s|--skills <Skills> [--repository <Model Manager Root
URI>] [-v|--verbose]
```

DEM-Add-Skills Arguments

Argument	Description
-n | - -name	Name of a registered Distributed Execution Manager.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-s | - -skills	Comma-delimited list of skills to associate with this Distributed Execution Manager.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Note The skills must already exist in the Model Manager. See Skill-Install.

DEM-List

Lists all Distributed Execution Managers registered with the Model Manager and their associated skills.

Synopsis

```
CloudUtil.exe DEM-List [--repository <Model Manager Root URI>] [-v|--verbose]
```

DEM-List Arguments

Argument	Description
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

DEM-Remove-Skills

Removes association between skills and a Distributed Execution Manager.

Synopsis

```
CloudUtil.exe DEM-Remove-Skills -n|--name <Name> -s|--skills <Skills> [--repository <Model Manager
Root URI>] [-v|--verbose]
```

DEM-Remove-Skills Arguments

Argument	Description
-n | - -name	Name of a registered Distributed Execution Manager.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-s | - -skills	Comma-delimited list of skills to remove from this Distributed Execution Manager.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

File Commands

The File commands enable you to store and manage files (typically scripts) in the Model Manager.

File-Export

Exports a file from the Model Manager.

Synopsis

```
CloudUtil.exe File-Export -n|--name <Name> -o|--output <Output File> [-i|--iteration <Iteration>] [--
repository <Model Manager Root URI>] [-v|--verbose]
```

File-Export Arguments

Argument	Description
-i | - -iteration	(Optional) Version string of the file in the Model Manager. Default is `0.0`.
-n | - -name	Friendly name of the file in the Model Manager.
-o | - -output	Path for file output.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

File-Import

Imports a file into the Model Manager.

Synopsis

```
CloudUtil.exe File-Import -n|--name <Name> -f|--filename <File Name> [-d|--description <Description>]
[-i|--iteration <Iteration>] [--repository <Model Manager Root URI>] [-v|--verbose]
```

File-Import Arguments

Argument	Description
-d | - -description	(Optional) Description of the file.
-f | - -filename	Path to a file to import into the Model Manager.
-i | - -iteration	(Optional) Version string of the file in the Model Manager. Default is `0.0`.
-n | - -name	Friendly name to assign to the file in the Model Manager.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

File-List

Lists all files imported into the Model Manager.

Synopsis

```
CloudUtil.exe File-List [--repository <Model Manager Root URI>] [-v|--verbose]
```

File-List Arguments

Argument	Description
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

File-Remove-All

Removes all revisions for a given version of a file from the Model Manager.

Synopsis

```
CloudUtil.exe File-Remove-All -n|--name <Name> [-i|--iteration <Iteration>]
[--repository <Model Manager Root URI>] [-v|--verbose]
```

File-Remove-All Arguments

Table 7-20.

Argument	Description
-i | - -iteration	(Optional) Version string of the file in the Model Manager. Default is `0.0`.
-n | - -name	Friendly name of the file in the Model Manager.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

File-Remove-Rev

Removes a specific revision of a file from the Model Manager.

Synopsis

```
CloudUtil.exe File-Remove-Rev -n|--name <Name> -r|--revision <Revision> [-i|--iteration <Iteration>]
[--repository <Model Manager Root URI>] [-v|--verbose]
```

File-Export Arguments

Argument	Description
-i | - -iteration	(Optional) Version string of the file in the Model Manager. Default is `0.0`.
-n | - -name	Friendly name of the file in the Model Manager.
-r | - -revision	Revision of the file to remove.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

File-Rollback

Reverts a file in the Model Manager to a specified revision.

Synopsis

```
CloudUtil.exe File-Rollback -n|--name <Name> -r|--revision <Revision> [-i|--iteration <Iteration>] [--
repository <Model Manager Root URI>] [-v|--verbose]
```

File-Rollback Arguments

Table 7-21.

Argument	Description
-i | - -iteration	(Optional) Version string of the file in the Model Manager. Default is `0.0`.
-n | - -name	Friendly name of the file in the Model Manager.
-r | - -revision	Revision of the file to revert to.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

File-Update

Updates a previously imported file in the Model Manager with a new revision.

Synopsis

```
CloudUtil.exe File-Update -n|--name <Name> -f|--filename <File Name> [-i|--iteration <Iteration>] [--
repository <Model Manager Root URI>] [-v|--verbose]
```

File-Update Arguments

Argument	Description
-f | - -filename	Path to the updated file.
-i | - -iteration	(Optional) Version string of the file in the Model Manager. Default is `0.0`.
-n | - -name	Friendly name of the file in the Model Manager.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Operation Commands

The Operation commands enable you to manage custom operations in the Model Manager.

Operation-Create

Creates a custom operation or set of operations that can be performed on a machine based on an operation definition file.

Synopsis

```
CloudUtil.exe Operation-Create -c|--operationConfig <Operation Definition File> [--repository <Model
Manager Root URI>] [-v|--verbose]
```

Operation-Create Arguments

Argument	Description
-c | - -operationConfig	Path to an operation definition file (XML).
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Operation-Delete

Deletes a custom operation from the Model Manager.

Synopsis

```
CloudUtil.exe Operation-Delete -n|--name <Name> [--force] [--repository <Model Manager
Root URI>] [-v|--verbose]
```

Operation-Delete Arguments

Argument	Description
- -force	(Optional) Force deletion of the operation.
-n | - -name	Name of the custom operation in the Model Manager.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Operation-List

Lists all custom operations in the Model Manager.

Synopsis

```
CloudUtil.exe Operation-List [--repository <Model Manager Root URI>] [-v|--verbose]
```

Operation-List Arguments

Argument	Description
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Skill Commands

The Skill commands enable you to manage the skills associated with Distributed Execution Managers and workflows.

Skill-Install

Installs a skill in the Model Manager.

Synopsis

```
CloudUtil.exe Skill-Install -n|--name <Name> [--repository <Model Manager Root URI>] [-v|--verbose]
```

Skill-Install Arguments

Argument	Description
-n | - -name	Name for the skill in the Model Manager.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Skill-List

List all skills installed in the Model Manager.

Synopsis

```
CloudUtil.exe Skill-List [--repository <Model Manager Root URI>] [-v|--verbose]
```

Skill-List Arguments

Argument	Description
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Skill-Uninstall

Uninstall a skill from the Model Manager.

Synopsis

```
CloudUtil.exe Skill-Uninstall -n|--name <Name> [--repository <Model Manager Root URI>]
[-v|--verbose]
```

Skill-Uninstall Arguments

Argument	Description
-n | - -name	Name of the skill to uninstall from the Model Manager.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Note A skill cannot be uninstalled if it is associated with either a Distributed Execution Manager or a workflow. See DEM-Remove-Skills or Workflow-Remove-Skills.

Workflow Commands

The Workflow commands enable you to manage customizable IaaS workflows in the Model Manager, as well as the skills associated with any workflows.

Workflow-Add-Skills

Associate skills with a workflow in the Model Manager.

```
CloudUtil.exe Workflow-Add-Skills -n|--name <Name> -s|--skills <Skills> [--repository <Model Manager
Root URI>] [-v|--verbose]
```

Table 7-22. Workflow-Add-Skills Arguments

Argument	Description
Name	Name of a workflow in the Model Manager.
Skills	Comma-delimited list of skills to associate with this workflow.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Note The skills must already exist in the Model Manager. See Skill-Install.

Workflow-List

List all workflows installed in the Model Manager and their associated skills.

```
CloudUtil.exe Workflow-List [--repository <Model Manager Root URI>] [-v|--verbose]
```

Table 7-23. Workflow-List Arguments

Argument	Description
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Workflow-Remove-Skills

Removes association between skills and a workflow in the Model Manager.

Synopsis

```
CloudUtil.exe Workflow-Remove-Skills -n|--name <Name> -s|--skills <Skills> [--repository
<Model Manager Root URI>] [-v|--verbose]
```

Workflow-Remove-Skills Arguments

Argument	Description
-n | - -name	Name of a workflow in the Model Manager.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.

Argument	Description
-s | - -skills	Comma-delimited list of skills to remove from this workflow.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Workflow-Rollback

Reverts a workflow to a given revision.

Synopsis

```
CloudUtil.exe Workflow-Rollback -n|--name <Name> -r|--revision <Revision> [--repository <Model
Manager Root URI>] [-v|--verbose]
```

Workflow-Rollback Arguments

Argument	Description
-n | - -name	Name of the workflow in the Model Manager.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-r | - -revision	Revision of the workflow to revert to.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Workflow-Update

Update a customizable workflow with a new revision.

```
CloudUtil.exe Workflow-Update -f|--filename <File Name> -n|--name <Name> [-d|--description
<Description>] [--repository <Model Manager Root URI>] [-v|--verbose]
```

Table 7-24. Workflow-Update Arguments

Argument	Description
File Name	Path to a file (XAML) containing the updated workflow.
Name	Name of the workflow to update.
Description	(Optional) Description of workflow.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Import Commands

The import commands enable you to import one or more virtual machines into a vRealize Automation deployment.

Machine-BulkRegisterExport

Creates a CSV data file that is used to import virtual machines to a vRealize Automation deployment.

Synopsis

```
CloudUtil.exe Machine-BulkRegisterExport [-b|--blueprint] [-m|--managed] [-e|--exportNames] [-p|--
properties] -f|--filename <Value> [-g|--group <Value>] [-i|--ignore] [-o|--owner <Value>] [-t|--
machinetype <Value>] [-n|--resourceName <Value>] [-r|--resourceType <Value>] [--repository <Value>] [-
sn|--sourcename <Value>] [-st|--sourcetype <Value>] -u|--user <value> [-v|--verbose]
```

Machine-BulkRegisterExport Arguments

Table 7-25.

Argument	Description
-b | - -blueprint	(Optional) Include blueprint name.
-e | - -exportNames	(Optional) Export names instead of GUIDs.
-f | - -filename	Specify the name of the CSV data file containing a list of machine names, for example, `filename.csv`. File is saved in the current path by default. You can also specify the complete path to a preferred directory.
-g | - -group	(Optional) Specify business group name, for example, Engineering.
-i | - -ignore	(Optional) Ignore invalid arguments.
-m | - -managed	(Optional) Export managed virtual machines. The default is export unmanaged virtual machines.
-n | - -resourceName	(Optional) To filter by resource name, specify the name of the Compute Resource or Endpoint.
-o | - -owner	(Optional) Specify owner of imported virtual machine, for example, jsmith.
-p | - -properties	(Optional) Export properties for managed virtual machines.
-r | - -resourceType	(Optional) To filter by resource type, specify 1 for Compute Resource or 2 for Endpoint .
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-sn | - -sourcename	(Optional) Specify the name of the cluster or endpoint.

Table 7-25. (continued)

Argument	Description
-st | - -sourcetype	(Optional) Specify the source type as Cluster or Endpoint.
-t | - -machinetype	(Optional) Specify machine type to be exported, for example, Virtual, Physical, Cloud, AppService, vApp.
-u | - -user	Specify the Fabric Administrator who performs the bulk registration.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.

Machine-BulkRegisterImport

Imports one or more virtual machines to a target vRealize Automation deployment.

Synopsis

```
CloudUtil.exe Machine-BulkRegisterImport [-b|--batch][-d|--delay <value>] -f|--filename <value> [-i|--
ignore] [-h|--humanreadable] -n|--name <value> [--repository <value>] [-s|--skipUser] -t|--time
<value> -u|--user <value> [-v|--verbose] [-w|--whatIf]
```

Machine-BulkRegisterImport Arguments

Table 7-26.

Argument	Description
-b | - -batch	(Optional) Batch size.
-d | - -delay	(Optional) Specify processing delay time in this format: hh:mm:ss, for example, 02:20:10.
-f | - -filename	Specify the CSV data file name containing the list of machine names. For example, `filename.csv`.
-h | --humanreadable	(Optional) Input file contains the virtual machine names and not the GUIDs.
-i | - -ignore	(Optional) Ignore registered or managed virtual machines.
-n | - -name	Specify the name of the work queue to perform the import to the target vRealize Automation.
- -repository	(Optional) The root URI of the Model Manager, for example, http://*hostname*/repository. Default is specified in the CloudUtil config file in the repositoryAddress key under the <appSettings> section.
-s | - -skipUser	(Optional) Sets a machine's owner to the value listed in the Owner column of the CSV data file without verifying that the user exists. Selecting this option can decrease the time required for import.

Table 7-26. (continued)

Argument	Description
-t | - -time	Specify the workflow start time in the format MM/DD/YYYY hh:mm GMT, for example, 04/18/2014 10:01 GMT. The specified start time is assumed to be the server's local time and not the local time of the user's workstation.
-u | - -user	Specify the Fabric Administrator who performs the bulk registration.
-v | - -verbose	(Optional) If an error occurs, outputs a stack trace instead of just the exception message.
whatif	(Optional) Set to validate the CSV file but do not import any virtual machines.

vRealize Automation Workflow Activity Reference

VMware provides a library of workflow activities with vRealize Automation Designer for use in customizing workflows.

Note The CDK is deprecated beginning with vRealize Automation 7.0. You can use the vRealize Orchestrator workflows to address use cases that you previously addressed with CDK.

Five categories of Windows Workflow Foundation activities are also included in vRealize Automation Designer, including Control Flow, Flowchart, Primitives, Collection and Error Handling.

This section provides a reference to the IaaS workflow activities included with vRealize Automation Designer in the `DynamicOps.Repository.Activities` and `DynamicOps.Cdk.Activities` namespaces. Activities related to calling vRealize Orchestrator workflows are described in Using vRealize Orchestrator Workflow Activities.

Note In the IaaS activity library, the Model Manager is referred to as the `repository`.

DynamicOps.Repository.Activities

The `DynamicOps.Repository.Activities` namespace contains basic workflow activities for IaaS workflows.

Note The CDK is deprecated beginning with vRealize Automation 7.0. You can use the vRealize Orchestrator workflows to address use cases that you previously addressed with CDK.

AddLink

Adds the specified link to the set of objects the `DataServiceContext` is tracking.

Table 7-27. AddLink Activity Input Parameters

Argument	Type	Description
DataServiceContext	RepositoryServiceContext	The `DataServiceContext` to which to add the link.
Source	Object	The source object for the new link.
SourceProperty	String	The name of the navigation property on the source object that returns the related object.
Target	Object	The object related to the source object by the new link.

AddObject

Adds the specified object to the set of objects the `DataServiceContext` is tracking.

Table 7-28. AddObject Activity Input Parameters

Argument	Type	Description
DataServiceContext	RepositoryServiceContext	The `DataServiceContext` to which to add the object.
Instance	Object	The object to be tracked by the `DataServiceContext`.

AttachTo

Notifies the `DataServiceContext` to start tracking the specified resource.

Table 7-29. AttachTo Activity Input Parameters

Argument	Type	Description
DataServiceContext	RepositoryServiceContext	The `DataServiceContext` that should track the resource.
Instance	Object	The resource to be tracked by the `DataServiceContext`. The resource is attached in the Unchanged state.

CreateRepositoryServiceContext<T>

Creates a context for a model loaded into the Model Manager.

When you add this activity to a workflow in vRealize Automation Designer, you must select a class that inherits from the `RepositoryServiceContext` class.

Table 7-30. CreateRepositoryServiceContext<T> Activity Input Parameters

Argument	Type	Description
Uri	URI	(Optional) Root URI to use in connecting to the model.
Username	String	(Optional) Username to use in connecting to the context.

Table 7-31. CreateRepositoryServiceContext<T> Activity Output Parameters

Argument	Type	Description
Result	RepositoryServiceContext	The specific type returned is an instance of the class selected when the activity was added to the workflow.

DeleteLink

Changes the state of the link to deleted in the list of links being tracked by the `DataServiceContext`.

Table 7-32. DeleteLink Activity Input Parameters

Argument	Type	Description
DataServiceContext	RepositoryServiceContext	The `DataServiceContext` from which to delete the link.
Source	Object	The source object in the link to be marked for deletion.
SourceProperty	String	The name of the navigation property on the source object that is used to access the target object.
Target	Object	The target object involved in the link that is bound to the source object. The target object must be of the type identified by the source property or a subtype.

DeleteObject

Changes the state of the specified object to be deleted in the `DataServiceContext`.

Table 7-33. DeleteObject Activity Input Parameters

Argument	Type	Description
DataServiceContext	RepositoryServiceContext	The `DataServiceContext` from which to delete the resource.
Instance	Object	The tracked entity to be changed to the deleted state.

InvokeRepositoryWorkflow

Executes a workflow installed in the Model Manager.

Table 7-34. InvokeRepositoryWorkflow Activity Input Parameters

Argument	Type	Description
WorkflowType	WorkflowDefinition entity	The workflow to execute.
WorkflowInputs	Dictionary<string, object>	(Optional) Inputs to the workflow.
CallingInstance	WorkflowInstance entity	(Optional) The workflow that calls the executed workflow and to which it will return.

LoadProperty

Loads deferred content for a specified property from the data service.

Table 7-35. LoadProperty Activity Input Parameters

Argument	Type	Description
DataServiceContext	RepositoryServiceContext	The `DataServiceContext` from which to load the property.
Instance	Object	The entity that contains the property to load.
InstanceProperty	String	The name of the property of the specified entity to load.

SaveChanges

Saves the changes that the `DataServiceContext` is tracking to storage.

Table 7-36. SaveChanges Activity Input Parameters

Argument	Type	Description
DataServiceContext	RepositoryServiceContext	The `DataServiceContext` that is tracking the changes to save.

SetLink

Notifies the `DataServiceContext` that a new link exists between the objects specified and that the link is represented by the property specified in the `SourceProperty` argument.

Table 7-37. SetLink Activity Input Parameters

Argument	Type	Description
DataServiceContext	RepositoryServiceContext	The `DataServiceContext` to notify of the link.
Source	Object	The source object for the new link.
SourceProperty	String	The property on the source object that identifies the target object of the new link.
Target	Object	The child object involved in the new link to initialize by calling this method. The target object must be a subtype of the type identified by SourceProperty. If Target is set to null, the call represents a delete link operation.

UpdateObject

Changes the state of the specified object in the `DataServiceContext` to Modified.

Table 7-38. UpdateObject Activity Input Parameters

Argument	Type	Description
DataServiceContext	RepositoryServiceContext	The `DataServiceContext` tracking the entity to update.
Instance	Object	The tracked entity to be assigned to the Modified state.

DynamicOps.Cdk.Activities

The `DynamicOps.Cdk.Activities` namespace contains advanced activities for IaaS workflows.

Note The CDK is deprecated beginning with vRealize Automation 7.0. You can use the vRealize Orchestrator workflows to address use cases that you previously addressed with CDK.

ExecutePowerShellScript

Executes a PowerShell script stored in the Model Manager under the specified name.

Before you use the `ExecutePowerShellScript` activity, you must first load the script that you want to execute into the Model Manager using the `CloudUtil File-Import` command.

Table 7-39. ExecutePowerShellScript Activity Input Parameters

Argument	Type	Description
ScriptName	String	Name in the Model Manager of the script to execute.
ScriptVersion	Object	(Optional) Version in the Model Manager of the script to execute. Default is 0.0.
MachineId	Guid	(Optional) If specified, the machine is loaded and all its properties are passed to the script.
Arguments	Dictionary<string,string>	Additional arguments to pass to the script. If MachineId is specified and there is a machine property with the same name as an argument (case-insensitive), the value of the machine property overrides the value of the argument.
PSModules	IEnumerable<string>	(Optional) Modules loaded into the PowerShell runtime during command execution. This option is only available in the Properties pane and not in the Designer pane.

Table 7-40. ExecutePowerShellScript Activity Output Parameters

Argument	Type	Description
Output	Collection<PSObject>	Output of script if any. Throws exception on error.

If you receive the error message `Type PSObject is not defined` in the vRealize Automation Designer console when you are dealing with the output of `ExecutePowerShellScript`, perform the following steps:

1 Click **Imports** in the lower left corner of the Designer pane.

2 Select the **System.Management.Automation** assembly.

ExecuteSshScript

Executes an SSH script stored in the model manager under the specified name.

Before you use the `ExecuteSshScript` activity, you must first load the script that you want to execute into the Model Manager using the `CloudUtil File-Import` command.

Table 7-41. ExecuteSshScript Activity Input Parameters

Argument	Type	Description
ScriptName	String	Name in the Model Manager of the script to execute.
Host	String	Server name against which to execute the script.
Username	String	Username to use in connecting to the host.
Password	String	Password to use in connecting to the host.
ScriptVersion	Object	(Optional) Version in the Model Manager of the script to execute. Default is 0.0.
Timeout	TimeSpan	(Optional) Period of time after which execution of the script times out. Default is 30 minutes.

Table 7-42. ExecuteSshScript Activity Output Parameters

Argument	Type	Description
EnvironmentVariables	Dictionary<string, string>	Script execution result if any.

GetMachineName

Gets a machine’s name.

Table 7-43. GetMachineName Activity Input Parameters

Argument	Type	Description
MachineId	Guid	The machine whose name to retrieve.

Table 7-44. GetMachineName Activity Output Parameters

Argument	Type	Description
MachineName	String	Name of the machine identifed by `MachineId`.

GetMachineOwner

Gets the username of a machine’s owner.

Table 7-45. GetMachineOwner Activity Input Parameters

Argument	Type	Description
MachineId	Guid	The machine whose owner to retrieve.

Table 7-46. GetMachineOwner Activity Output Parameters

Argument	Type	Description
Owner	String	Owner of the machine identified by `MachineId`, or null if there is no owner.

GetMachineProperties

Gets the list of custom properties associated with a machine.

Table 7-47. GetMachineProperties Activity Input Parameters

Argument	Type	Description
MachineId	Guid	The machine whose properties to retrieve.

Table 7-48. GetMachineProperties Activity Output Parameters

Argument	Type	Description
Properties	Dictionary<string, string>	List of the machine's properties. Values are returned decrypted if they were stored encrypted.

GetMachineProperty

Gets the value of the specified property for a machine.

Table 7-49. GetMachineProperty Activity Input Parameters

Argument	Type	Description
MachineId	Guid	The machine from which to retrieve a property.
PropertyName	String	Name of the property whose value to return.
IsRequired	bool	If the property is required and is not found the activity throws an exception, otherwise returns null.

Table 7-50. GetMachineProperty Activity Output Parameters

Argument	Type	Description
PropertyValue	String	Value of the property specified by `PropertyName`. The value is returned decrypted if it was stored encrypted.

GetScriptFromName

Gets the contents of the script stored in the Model Manager under the specified name.

Table 7-51. GetScriptFromName Activity Input Parameters

Argument	Type	Description
ScriptName	String	Name in the Model Manager of the script to retrieve.
ScriptVersion	Object	(Optional) Version in the Model Manager of the script to retrieve. Default is 0.0.

Table 7-52. GetScriptFromName Activity Output Parameters

Argument	Type	Description
ScriptContent	String	Contents of the script identified by `ScriptName`.

InvokePowerShell

Executes a PowerShell command.

Table 7-53. InvokePowerShell Activity Input Parameters

Argument	Type	Description
CommandText	String	Command to execute.
Arguments	IEnumerable<string>	(Optional) Arguments to the command.
Input	IEnumerable	(Optional) The input pipeline.
IsScript	bool	(Optional) Indicates whether `CommandText` is a script. Default is False. This option is only available in the Properties pane and not in the Designer pane.
Parameters	Collection	(Optional) Collection of name-value pairs passed as parameters to the PowerShell script. This option is only available in the Properties pane and not in the Designer pane.
PowerShellVariables	Collection	(Optional) Variables copied into the PowerShell runtime. This option is only available in the Properties pane and not in the Designer pane.
PSModules	IEnumerable<string>	(Optional) Modules loaded into the PowerShell runtime during command execution. This option is only available in the Properties pane and not in the Designer pane.
Runspace	Runspace	(Optional) Creating a PowerShell runspace and supplying it to this argument enables you to reuse the same runspace in multiple PowerShell invocations, which may result in performance improvements. This option is only available in the Properties pane and not in the Designer pane.

Table 7-54. InvokePowerShell Activity Output Parameters

Argument	Type	Description
Output	Collection<PSObject>	Output of command if any. Throws exception on error.
Errors	Collection<ErrorRecord>	Errors resulting from execution if any.

If you receive the error message `Type PSObject is not defined` in the vRealize Automation Designer console when you are dealing with the output of `ExecutePowerShellScript`, perform the following steps:

1 Click **Imports** in the lower left corner of the Designer pane.

2 Select the **System.Management.Automation** assembly.

InvokeSshCommand

Executes an SSH command.

Table 7-55. InvokeSshCommand Activity Input Parameters

Argument	Type	Description
CommandText	String	Command to execute.
Host	String	Server name against which to execute the command.
Username	String	Username to use in connecting to the host.
Password	String	Password to use in connecting to the host.
Timeout	TimeSpan	(Optional) Period of time after which execution of the command times out. Default is 30 minutes.

Table 7-56. InvokeSshCommand Activity Output Parameters

Argument	Type	Description
EnvironmentVariables	Dictionary<string, string>	Output of command if any. Throws exception on error.

LogMachineEvent

Logs a machine event to the user log that is visible to the machine owner.

Table 7-57. LogMachineEvent Activity Input Parameters

Argument	Type	Description
MachineId	Guid	Machine generating the event to log.
Message	String	Message to write to the user log.
Type	String	Select a message type from the drop-down list (Info, Warn, Error)

LogMessage

Logs to the Distributed Execution Manager log.

Table 7-58. LogMessage Activity Input Parameters

Argument	Type	Description
Message	String	Message to write to the DEM log.
MessageCategory	String	Select a category from the drop-down menu (**Debug**, **Error**, **Info**, **Trace**) or enter a custom category.
MessageSeverity	String	Select a severity from the drop-down menu; bound to the list of Severities supplied in `System.Diagnostics.TraceEventType`.

RunProcess

Executes a process on the same machine as the DEM that executes this activity.

Note vRealize Automation cannot present the UI from processes launched by the `RunProcess` activity to the user, therefore these process must be non-interactive. In order to avoid leaving orphaned processes on the DEM machine, the processes must also be self-terminating.

Table 7-59. RunProcess Activity Input Parameters

Argument	Type	Description
Command	String	Path to the executable to run on the DEM machine.
WorkingDirectory	String	(Optional) The working directory under which the process should run.
Arguments	String	(Optional) The list of command-line arguments to pass to the command.
WaitForExit	bool	(Optional) If true, the workflow waits for the process to complete before continuing with the workflow. Default is false. This option is only available in the Properties pane and not in the Designer pane.

SendEmail

Sends an email to the given set of addresses.

Table 7-60. SendEmail Activity Input Parameters

Argument	Type	Description
To	IEnumerable<string>	The list of addresses to which to send the email.
From	String	The address with which to populate the “From” field of the email.
Subject	String	The subject line for the email.
Body	String	The body text of the email.
Host	String	The host name or IP address of the outgoing SMTP server.
Port	Integer	The SMTP port on the server specified in Host. This option is only available in the Properties pane and not in the Designer pane.
CC	IEnumerable<string>	(Optional) The address or list of addresses to copy on the email. This option is only available in the Properties pane and not in the Designer pane.

Table 7-60. SendEmail Activity Input Parameters (continued)

Argument	Type	Description
Bcc	IEnumerable<string>	(Optional) The address or list of addresses to blind copy on the email. This option is only available in the Properties pane and not in the Designer pane.
EnableSsl	bool	(Optional) Indicates whether to use SSL. This option is only available in the Properties pane and not in the Designer pane.
UserName	String	The user name with which to authenticate with the SMTP server specified in `Host` . This option is only available in the Properties pane and not in the Designer pane.
Password	String	The password of the user specified in `UserName`. This option is only available in the Properties pane and not in the Designer pane.

SetMachineProperty

Creates or updates a custom property on a machine.

Table 7-61. SetMachineProperty Activity Input Parameters

Argument	Type	Description
MachineId	Guid	Machine on which to create or update the custom property.
PropertyName	String	Name of property to create or update.
PropertyValue	String	Value with which to create or update the property.
IsEncrypted	bool	(Optional) Indicates whether the value of the property is encrypted.
IsHidden	bool	(Optional) Indicates whether the property is a hidden property.
IsRuntime	bool	(Optional) Indicates whether the requesting user provides the property value at request time (equivalent to being marked Prompt User in the vRealize Automation console).

SetWorkflowResult

Sets an external workflow’s state to either Complete or Failed to be honored by `ExternalWF.xml` settings.

Table 7-62. SetWorkflowResult Activity Input Parameters

Argument	Type	Description
WorkflowId	Guid	The workflow for which to set the state.
Next State	WorkflowState	Select **Complete** or **Failed** from the drop-down menu.

Custom Properties and the Property Dictionary

8

You can use supplied vRealize Automation custom properties to control various aspects of machine provisioning. You can also use the property dictionary to create new property definitions and property groups that are tailored to your specific needs.

You can use properties to add values or override existing or default values for configuring network, platform, and guest agent settings and many other deployment-related parameters.

This chapter includes the following topics:

Using Custom Properties

You can use vRealize Automation custom properties to add values or override existing or default values for configuring settings for network, platform, guest agent, and many other deployment parameters.

Some properties are determined by standard settings that you must specify for all machines. For example, memory and disk size values are required for all blueprints. You can specify additional properties individually or in property groups in blueprints and in reservations.

When you add a property to a blueprint or a property group, you can mark it as a required property. When a property is specified as required, the user must provide a value for that property when they request a machine, such as in the following examples.

- Require information about multiple disks sharing the machine’s allocated storage.
- Require information about users or groups to be added to a local group on the machine.
- Require the host name of the machine.

The Windows guest agent records property values on the provisioned machine in the `%SystemDrive%\VRMGuestAgent\site\workitem.xml` file.

The Linux guest agent records property values on the provisioned machine in the `/usr/share/gugent/site/workitem.xml` file.

Creating and Adding Custom Properties and Property Groups

You can use custom properties to control machine provisioning. You can add supplied custom properties and also create and add your own properties and property groups.

You can add properties and property groups to overall blueprints, components in a blueprint, reservations and other vRealize Automation items, including some endpoint types. You can also create new custom properties and property groups.

You can add properties and property groups when you create a blueprint, or later when the blueprint is in the draft or published state. Alternatively you can add custom properties and property groups to individual components in the blueprint.

Blueprint-level custom properties take precedence over custom properties that are configured at the component level. For information about custom property precedence, see Understanding Custom Properties Precedence.

You can edit blueprint-level properties by using the blueprint properties page.

A custom property can optionally require that the user specify a property value when they create a machine request.

- Custom property names and values are typically case-sensitive. For example, a custom property expressed as `hostname` and another custom property expressed as `HOSTNAME` are considered different custom properties.
- Custom property names cannot contain spaces. When creating and using custom properties, do not include a space in the property name.
- Some custom property names are reserved and cannot be used as names when you create new custom properties. For example the property name `Encrypted` and `encrypted` is reserved.

For more information about creating new custom properties and property groups, see Using the Property Dictionary.

Using Properties in Machine Provisioning

Custom properties are vRealize Automation-supplied properties. You can also define your own properties. Properties are name-value pairs used to specify attributes of a machine or to override default specifications.

You can use custom properties to control different provisioning methods, types of machines, and machine options as in the following examples:

- Specify a particular type of guest OS.

- Enable WIM-based provisioning, in which a Windows Imaging File Format (WIM) image of a reference machine is used to provision new machines.
- Customize the behavior of Remote Desktop Protocol when connecting to a machine.
- Register a virtual machine with a XenDesktop Desktop Delivery Controller (DDC) server.
- Customize a virtual machine's system specifications, such as adding multiple disk drives.
- Customize the guest OS for a machine, for instance, by including specified users in selected local groups.
- Specify network and security settings.
- Add additional control options such as drop-down menus to make input and selection options available to the consumer at request time.

When you add a property to a blueprint, reservation, or other form you can specify if the property is to be encrypted and also if the user must be prompted to specify a value when provisioning. These options cannot be overridden when provisioning.

For an example of how to additional control options to dynamically set a custom property based on a consumer's selection from a list of predefined options, see the Adding a Network Selection Drop-Down in vRA 7 blog post.

A property specified in a blueprint overrides the same property specified in a property group. This enables a blueprint to use most of the properties in a property group while differing from the property group in some limited way. For example, a blueprint that incorporates a standard developer workstation property group might override the US English settings in the group with UK English settings.

You can apply properties in reservations and business groups to many machines. Their use is typically limited to purposes related to their sources, such as resource management. Specifying the characteristics of the machine to be provisioned is generally done by adding properties to blueprints and property groups.

Understanding Custom Properties Precedence

Properly authorized users can specify custom properties for blueprints, endpoints, business groups, and reservations. When the same property exists in more than one source, vRealize Automation follows a specific order of precedence when applying properties to the machine.

You can add custom properties that apply to provisioned machines to the following elements:

- A reservation, to apply the custom properties to all machines provisioned from that reservation.
- A business group, to apply the custom properties to the deployment and to all machines provisioned by business group members.
- A blueprint, to apply the custom properties to all machines provisioned from the blueprint.
- Property groups, which can be included in a blueprint, to apply all the custom properties in the group to all machines provisioned from the blueprint.

A blueprint can contain one or more property groups.

- A machine request to apply the custom properties to the machine being provisioned.
- An approval policy, if advanced approval support is enabled, to require approvers to provide values for the machine being approved.

The following list shows the order of precedence for custom properties. Property values specified in a source that appears later in the list override values for the same property specified in sources that appear earlier in the list.

If a conflict exists between a vRealize Automation-supplied custom property name and a user-defined property name, the vRealize Automation-supplied custom property name takes precedence.

1. Property group
2. Blueprint
3. Business group
4. Compute resource
5. Reservations
6. Endpoint
7. Runtime

Property group, blueprint, and business group custom properties are assigned at request time, while other compute resource, reservation, and endpoint properties are assigned during provisioning.

This order is further clarified as follows:

1. Custom properties and groups at the overall blueprint level
2. Custom properties and groups at the component level
3. Custom properties for the business group
4. Custom properties for the compute resource
5. Custom properties for the reservation
6. Custom properties for the endpoint
7. Custom properties at the nested blueprint request level
8. Custom properties at the component request level

In most situations, a runtime property takes precedence over other properties. A runtime property meets the following conditions:

- The custom property option to prompt the user is selected, which specifies that the user must supply a value for the property when they request machine provisioning.

- A business group manager is requesting machine provisioning and the property appears in the custom properties list on the machine request confirmation page.

There are exceptions to the precedence rules. For example, you add the `VMware.VirtualCenter.Folder` custom property to a business group, provide a property value, and do not select the option to show the property in the request. You add the same custom property to a blueprint and specify that the property be shown in the request. When your designated users request provisioning from the catalog, the property does not appear in the catalog request form because the property applies to reservation information that is only available after provisioning begins, and not when you request provisioning.

Custom Property Types

You can use vRealize Automation external and updated property types for cloned machines. You cannot use Internal and read-only property types for cloned machines.

The following vRealize Automation custom property types are available.

- Internal

  The specified value is maintained in the database only. For example, the email address of the manager who approved a machine request is recorded in the `VirtualMachine.Admin.Approver` property but the property has no effect on the machine.

- Read-only

  The specified value is implemented on the machine and cannot be changed. For example, `VirtualMachine.Admin.UUID` specifies the UUID of the machine, which cannot be changed.

- External

  A machine’s external properties are determined when the virtualization platform creates the machine or during the WinPE phase of the build process. To set these properties, their values must be provided to the proxy agent, which passes them on to the virtualization platform, or to the guest agent , which implements them in the WinPE phase.

  The specified value is implemented on the machine but is never updated. For example, if the property `VirtualMachine.Admin.AddOwnerToAdmins` is set to true, the owner of the machine is added to its local administrators group. If the owner is later removed from this group, the property is not updated to false.

- Updated

  The specified value is implemented on the machine and is updated through data collection. For example, if the compute resource of a machine is changed, a proxy agent updates the value of the machine’s `VirtualMachine.Admin.Hostname` property.

Internal and read-only property types set attributes that the template determines.

You can use the vRealize Automation machine menu to change all reserved custom properties except the read-only properties `VirtualMachine.Admin.AgentID`, `VirtualMachine.Admin.UUID`, and `VirtualMachine.Admin.Name`.

Custom Properties Grouped by Function

You can use custom properties to provide additional vRealize Automation controls.

Custom properties have been grouped here by function. To explore custom properties grouped by name, see Custom Properties Grouped by Name.

- Custom Properties for Deployments

  vRealize Automation provides several custom properties that are applicable to most deployments.

- Custom Properties for Naming and Analyzing Deployments

  If provisioning fails, vRealize Automation rolls back all resources included in the catalog item. For deployments that contain multiple components, you can use a custom property to override that default and receive information to debug the failure. These properties are best used when applied to the overall blueprint.

- Custom Properties for OpenStack Endpoints

  vRealize Automation includes custom properties you might want to use when you configure your OpenStack endpoints in vRealize Automation.

- Custom Properties for Clone Blueprints

  vRealize Automation includes custom properties that you can use to provide additional controls for clone blueprints.

- Custom Properties for Linked Clone Blueprints

  vRealize Automation includes custom properties that you can use to provide additional controls for linked clone blueprints.

- Custom Properties for FlexClone Blueprints

  vRealize Automation includes custom properties that you can use to provide additional controls for FlexClone blueprints.

- Custom Properties for Basic Workflow Blueprints

  vRealize Automation includes custom properties that you can use to provide additional controls for basic workflow blueprints.

- Custom Properties for Linux Kickstart Blueprints

  vRealize Automation includes custom properties that you can use to provide additional controls for Linux Kickstart blueprints.

- Custom Properties for SCCM Blueprints

  vRealize Automation includes custom properties that you can use to provide additional controls for SCCM blueprints.

- Custom Properties for WIM Blueprints

  vRealize Automation includes custom properties that provide additional controls for WIM blueprints.

- Custom Properties for vCloud Air and vCloud Director Blueprints

  You can add certain custom properties to a vCloud Air or vCloud Director machine component definition in a blueprint.

- Custom Properties for Networking and Security

  The vRealize Automation custom properties for networking specify configuration for a specific network device on a machine.

- Custom Properties and Property Groups for Containers

  You can add predefined property groups to a containers component in a vRealize Automation blueprint. When machines are provisioned by using a blueprint that contain these properties, the provisioned machine is registered as a Docker Container host machine.

- Custom Properties for PXE Provisioning

  PXE is the only provisioning method supported for Cisco UCS Manager. You can use the network bootstrap program with vRealize Automation custom properties to initiate WIM, SCCM, or Linux Kickstart provisioning. You can also use custom properties to call your own PowerShell scripts. Linux Kickstart provisioning does not require custom properties.

- Custom Properties for OVF Import

  When you import an OVF to a blueprint, you can import and configure several settings as custom properties.

- Custom Properties for vRealize Automation Guest Agent

  If you have installed the vRealize Automation guest agent in your templates for cloning or in your WinPE, you can use custom properties to run custom scripts within the guest operating system of a provisioned machine after the machine is fully deployed.

- Custom Properties for BMC BladeLogic Configuration Manager Integration

  vRealize Automation includes custom properties that you can use to provide additional controls for BMC BladeLogic Configuration Manager integration.

- Custom Properties for HP Server Automation Integration

  vRealize Automation includes custom properties that you can use to provide additional controls for HP Server Automation integration. Some custom properties are required for HP Server Automation integration. Other custom properties are optional.

Custom Properties for Deployments

vRealize Automation provides several custom properties that are applicable to most deployments.

Table 8-1. Custom Properties for Blueprints and Deployments

Custom Property	Description
`_debug_deployment`	Except for scale operations which allow partially successful deployments, the default behavior is to destroy the entire deployment if any of the individual resources fail to provision. You can override the default behavior by setting the `_debug_deployment` custom property value to true. If provisioning fails, the debugging custom property stops the resources from being rolled back so you can identify which of the components failed to provision successfully. In other words, by setting `_debug_deployment` to true, you can more easily debug customization and first-boot (for example, agent) issues because the setting ensures that machines are not destroyed after a provisioning failure. Otherwise, the setting doesn't directly change anything about the provisioning process, or affect guest agent or customization (for example, settings our outcomes relative to a vCenter customization spec). Note: A failed catalog item is normally inaccessible because it is immediately rolled back on failure. But when `_debug_deployment` is set to true, vRealize Automation treats the otherwise failed deployment as partially successful, which enables its accessibility. To apply the custom property to a blueprint, add `_debug_deployment` to the **Blueprint Properties** page using the **Properties** tab when you create or edit a blueprint. The `_debug_deployment` property is consumed at the software provisioning level, not the guest agent or machine provisioning level. You can also configure vRealize Automation to not delete virtual machines after deployment failure by using settings in the `VRMAgent.exe.config` file.
`_deploymentName`	When added to a blueprint, this property allows you to specify a custom name for the deployment by setting the value of `_deploymentName` to your custom string. If more than one instance of this deployment is provisioned in a single request, your custom name becomes a prefix. If you want users to specify their own deployment names, set this custom property to allow override. The following two caveats are required for usage: ■ You must add this property at the blueprint level, not at the component level. For example, when creating or editing a blueprint, click the **Properties** tab and then select **Custom Properties > New** to add the `_deploymentName` property to the blueprint. Do not add the property to a machine or other component in the blueprint. ■ You must add this property as a separate property and not as a member of a property group.

Custom Properties for Naming and Analyzing Deployments

If provisioning fails, vRealize Automation rolls back all resources included in the catalog item. For deployments that contain multiple components, you can use a custom property to override that default and receive information to debug the failure. These properties are best used when applied to the overall blueprint.

Table 8-2. Custom Properties for Analyzing Deployments

Custom Property	Description
`_debug_deployment`	Except for scale operations which allow partially successful deployments, the default behavior is to destroy the entire deployment if any of the individual resources fail to provision. You can override the default behavior by setting the `_debug_deployment` custom property value to true. If provisioning fails, the debugging custom property stops the resources from being rolled back so you can identify which of the components failed to provision successfully. Note: A failed catalog item is normally inaccessible because it is immediately rolled back on failure. But when `_debug_deployment` is set to true, vRealize Automation treats the otherwise failed deployment as partially successful, which enables its accessibility. In other words, by setting `_debug_deployment` to true, you can more easily debug customization and first-boot (for example, agent) issues because the setting ensures that machines are not destroyed after a provisioning failure. Otherwise, the setting doesn't directly change anything about the provisioning process, or affect guest agent or customization (for example, settings our outcomes relative to a vCenter customization spec). To apply the custom property to a blueprint, add `_debug_deployment` to the **Blueprint Properties** page using the **Properties** tab when you create or edit a blueprint. The `_debug_deployment` property is consumed at the software provisioning level, not the guest agent or machine provisioning level. You can also configure vRealize Automation to not delete virtual machines after deployment failure by using settings in the `VRMAgent.exe.config` file.
`_deploymentName`	When added to a blueprint, this property allows you to specify a custom name for the deployment by setting the value of `_deploymentName` to your custom string. If more than one instance of this deployment is provisioned in a single request, your custom name becomes a prefix. If you want users to specify their own deployment names, set this custom property to allow override. The following two caveats are required for usage: ■ You must add this property at the blueprint level, not at the component level. For example, when creating or editing a blueprint, click the **Properties** tab and then select **Custom Properties > New** to add the `_deploymentName` property to the blueprint. Do not add the property to a machine or other component in the blueprint. ■ You must add this property as a separate property and not as a member of a property group.

Custom Properties for OpenStack Endpoints

vRealize Automation includes custom properties you might want to use when you configure your OpenStack endpoints in vRealize Automation.

Table 8-3. Custom Properties for Openstack Endpoints

Custom Property	Description
`VirtualMachine.Admin.ConnectAddress.Regex`	Used by a vRealize Automation administrator to define a regular expression to match an IP address for terminal connections, such as an RDP connection. If matched, the IP address is saved under the `VirtualMachine.Admin.ConnectAddress` custom property. Otherwise, the first available IP address is designated. For example, setting the property value to `10.10.0.` allows selection of an IP address from a 10.10.0.* subnet that is assigned to the virtual machine. If the subnet has not been assigned, the property is ignored. This property is available for use with OpenStack.
`VirtualMachine.NetworkN.AdditionAddress`*M*	Defines additional *M* IP address allocated for an OpenStack instance for network *N*, excluding the IP address set specified by the `VirtualMachine.Network`*N*`.Address`. property. More addresses are displayed on the Network tab in the Additional Addresses column. This property is used by OpenStack machine state data collection. While this property is only data-collected by the OpenStack endpoint, it is not specific to OpenStack and can be used for lifecycle extensibility by other endpoint types. This property is not supported for on-demand NAT or on-demand routed networks.
`VMware.Endpoint.Openstack.IdentityProvider.Domain.Name`	Allows vRealize Automation to support required Keystone V3 domain-name authentication. If Keystone V3 is in effect, you can use the property to designate a specific domain for the OpenStack endpoint to authenticate with a Keystone V3 OpenStack identity provider. ■ For new endpoints, add the custom property to designate a specific domain. ■ For upgraded or migrated endpoints, add the custom property only if data collection fails after upgrade or migration.
`VMware.Endpoint.Openstack.IdentityProvider.Version`	Specifies the version of OpenStack identity provider (Keystone) to use when authenticating an OpenStack endpoint. Configure a value of **3** to authenticate with Keystone V3 OpenStack identity provider. If you use any other value, or do not use this custom property, authentication defaults to Keystone V2.

Custom Properties for Clone Blueprints

vRealize Automation includes custom properties that you can use to provide additional controls for clone blueprints.

Table 8-4. Custom Properties for Clone Blueprints

Custom Property	Description
`VirtualMachine.DiskN.StorageReservationPolicy`	Specifies the storage reservation policy to use to find storage for disk *N*. Also assigns the named storage reservation policy to a volume. To use this property, substitute the volume number for *N* in the property name and specify a storage reservation policy name as the value. This property is equivalent to the storage reservation policy name specified on the blueprint. Disk numbering must be sequential. This property is valid for all Virtual and vCloud reservations. This property is not valid for Physical, Amazon, or OpenStack reservations.
`VirtualMachine.NetworkN.NetworkProfileName`	Specifies the name of a network profile from which to assign a static IP address to network device *N* or from which to obtain the range of static IP addresses that can be assigned to network device *N* of a cloned machine, where *N*=0 for the first device, 1 for the second, and so on. The network profile that the property points to is used to allocate an IP address. The property determines the network that the machine attaches to, based on the reservation. Changing this property value after the network is assigned has no effect on the expected IP address values for the designated machines. With WIM-based provisioning for virtual machines, you can use this property to specify a network profile and network interface or you can use the Network section of the Virtual Reservation page. The following attributes of the network profile are available to enable static IP assignment in a cloning blueprint: ■ `VirtualMachine.NetworkN.SubnetMask` ■ `VirtualMachine.NetworkN.Gateway` ■ `VirtualMachine.NetworkN.PrimaryDns` ■ `VirtualMachine.NetworkN.SecondaryDns` ■ `VirtualMachine.NetworkN.PrimaryWins` ■ `VirtualMachine.NetworkN.SecondaryWins` ■ `VirtualMachine.NetworkN.DnsSuffix` ■ `VirtualMachine.NetworkN.DnsSearchSuffixes` `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation.

Table 8-4. Custom Properties for Clone Blueprints (continued)

Custom Property	Description
`Linux.ExternalScript.Name`	Specifies the name of an optional customization script, for example `config.sh`, that the Linux guest agent runs after the operating system is installed. This property is available for Linux machines cloned from templates on which the Linux agent is installed. If you specify an external script, you must also define its location by using the `Linux.ExternalScript.LocationType` and `Linux.ExternalScript.Path` properties.
`Linux.ExternalScript.LocationType`	Specifies the location type of the customization script named in the `Linux.ExternalScript.Name` property. This can be either local or nfs. You must also specify the script location using the `Linux.ExternalScript.Path` property. If the location type is nfs, also use the `Linux.ExternalScript.Server` property.
`Linux.ExternalScript.Server`	Specifies the name of the NFS server, for example lab-ad.lab.local, on which the Linux external customization script named in `Linux.ExternalScript.Name` is located.
`Linux.ExternalScript.Path`	Specifies the local path to the Linux customization script or the export path to the Linux customization on the NFS server. The value must begin with a forward slash and not include the file name, for example `/scripts/linux/config.sh`.

If your administrators installed the guest agent to run scripts that accept custom properties and customize provisioned machines, you can use custom properties to further customize cloned machines that use the guest agent.

Table 8-5. Custom Properties for Customizing Cloned Machines with a Guest Agent

Custom Property	Description
`VirtualMachine.Admin.AllowLogin`	Set to True (default) to add the machine owner to the local remote desktop users group, as specified by the `VirtualMachine.Admin.Owner` property.
`VirtualMachine.Admin.UseGuestAgent`	If the guest agent is installed as a service on a template for cloning, set to True on the machine blueprint to enable the guest agent service on machines cloned from that template. When the machine is started, the guest agent service is started. Set to False to deactivate the guest agent. If set to False, the enhanced clone workfow will not use the guest agent for guest operating system tasks, reducing its functionality to `VMwareCloneWorkflow`. If not specified or set to anything other than False, the enhanced clone workflow sends work items to the guest agent.

Table 8-5. Custom Properties for Customizing Cloned Machines with a Guest Agent (continued)

Custom Property	Description
`VirtualMachine.DiskN.Active`	Set to True (default) to specify that the machine's disk *N* is active. Set to False to specify that the machine's disk *N* is not active.
`VirtualMachine.DiskN.Label`	Specifies the label for a machine's disk *N*. The disk label maximum is 32 characters. Disk numbering must be sequential. When used with a guest agent, specifies the label of a machine's disk *N* inside the guest operating system.
`VirtualMachine.DiskN.Letter`	Specifies the drive letter or mount point of a machine's disk *N*. The default is C. For example, to specify the letter D for Disk 1, define the custom property as `VirtualMachine.Disk1.Letter` and enter the value D. Disk numbering must be sequential. When used in conjunction with a guest agent, this value specifies the drive letter or mount point under which an additional disk *N* is mounted by the guest agent in the guest operating system.
`VirtualMachine.Admin.CustomizeGuestOSDelay`	Specifies the time to wait after customization is complete and before starting the guest operating system customization. The value must be in HH:MM:SS format. If the value is not set, the default value is one minute (00:01:00). If you choose not to include this custom property, provisioning can fail if the virtual machine reboots before guest agent work items are completed, causing provisioning to fail.
`VirtualMachine.Customize.WaitComplete`	Set to True to prevent the provisioning workflow from sending work items to the guest agent until all customizations arecomplete. Set to False to allow work items to be created before customization is complete.
`VirtualMachine.SoftwareN.Name`	Specifies the descriptive name of a software application *N* or script to install or run during provisioning. This is an optional and information-only property. It serves no real function for the enhanced clone workflow or the guest agent but it is useful for a custom software selection in a user interface or for software use reporting.
`VirtualMachine.SoftwareN.ScriptPath`	Specifies the full path to an application's install script. The path must be a valid absolute path as seen by the guest operating system and must include the name of the script filename. You can pass custom property values as parameters to the script by inserting {*CustomPropertyName*} in the path string. For example, if you have a custom property named `ActivationKey` whose value is 1234, the script path is `D:\InstallApp.bat –key {ActivationKey}`. The guest agent runs the command `D:\InstallApp.bat –key 1234`. Your script file can then be programmed to accept and use this value.

Table 8-5. Custom Properties for Customizing Cloned Machines with a Guest Agent (continued)

Custom Property	Description
`VirtualMachine.SoftwareN.ISOName`	Specifies the path and filename of the ISO file relative to the datastore root. The format is `/folder_name/subfolder_name/file_name.iso`. If a value is not specified, the ISO is not mounted.
`VirtualMachine.SoftwareN.ISOLocation`	Specifies the storage path that contains the ISO image file to be used by the application or script. Format the path as it appears on the host reservation, for example `netapp-1:it_nfs_1`. If a value is not specified, the ISO is not mounted.

Custom Properties for Linked Clone Blueprints

vRealize Automation includes custom properties that you can use to provide additional controls for linked clone blueprints.

Certain vRealize Automation custom properties are required to use with linked clone blueprints.

Table 8-6. Custom Properties for Linked Clone Blueprints

Custom Property	Description
`VirtualMachine.DiskN.Storage`	Specifies the datastore on which to place the machine disk *N*, for example DATASTORE01. This property is also used to add a single datastore to a linked clone blueprint. *N* is the index (starting at 0) of the volume to assign. Enter the name of the datastore to assign to the volume. This is the datastore name as it appears in the Storage Path on the Edit Compute Resource page. Disk numbering must be sequential.
`VirtualMachine.DiskN.StorageReservationPolicy`	Specifies the storage reservation policy to use to find storage for disk *N*. Also assigns the named storage reservation policy to a volume. To use this property, substitute the volume number for *N* in the property name and specify a storage reservation policy name as the value. This property is equivalent to the storage reservation policy name specified on the blueprint. Disk numbering must be sequential. This property is valid for all Virtual and vCloud reservations. This property is not valid for Physical, Amazon, or OpenStack reservations.
`VirtualMachine.DiskN.Label`	Specifies the label for a machine's disk *N*. The disk label maximum is 32 characters. Disk numbering must be sequential. When used with a guest agent, specifies the label of a machine's disk *N* inside the guest operating system.

Table 8-6. Custom Properties for Linked Clone Blueprints (continued)

Custom Property	Description
`VirtualMachine.DiskN.Letter`	Specifies the drive letter or mount point of a machine's disk *N*. The default is C. For example, to specify the letter D for Disk 1, define the custom property as `VirtualMachine.Disk1.Letter` and enter the value D. Disk numbering must be sequential. When used in conjunction with a guest agent, this value specifies the drive letter or mount point under which an additional disk *N* is mounted by the guest agent in the guest operating system.
`MaximumProvisionedMachines`	Specifies the maximum number of linked clones for one machine snapshot. The default is unlimited.
`Linux.ExternalScript.Name`	Specifies the name of an optional customization script, for example `config.sh`, that the Linux guest agent runs after the operating system is installed. This property is available for Linux machines cloned from templates on which the Linux agent is installed. If you specify an external script, you must also define its location by using the `Linux.ExternalScript.LocationType` and `Linux.ExternalScript.Path` properties.
`Linux.ExternalScript.LocationType`	Specifies the location type of the customization script named in the `Linux.ExternalScript.Name` property. This can be either local or nfs. You must also specify the script location using the `Linux.ExternalScript.Path` property. If the location type is nfs, also use the `Linux.ExternalScript.Server` property.
`Linux.ExternalScript.Server`	Specifies the name of the NFS server, for example lab-ad.lab.local, on which the Linux external customization script named in `Linux.ExternalScript.Name` is located.
`Linux.ExternalScript.Path`	Specifies the local path to the Linux customization script or the export path to the Linux customization on the NFS server. The value must begin with a forward slash and not include the file name, for example `/scripts/linux/config.sh`.

If you installed the guest agent to customize cloned machines, you use some custom properties more often than others.

Table 8-7. Custom Properties for Customizing Cloned Machines with a Guest Agent

Custom Property	Description
`VirtualMachine.Admin.UseGuestAgent`	If the guest agent is installed as a service on a template for cloning, set to True on the machine blueprint to enable the guest agent service on machines cloned from that template. When the machine is started, the guest agent service is started. Set to False to deactivate the guest agent. If set to False, the enhanced clone workfow will not use the guest agent for guest operating system tasks, reducing its functionality to `VMwareCloneWorkflow`. If not specified or set to anything other than False, the enhanced clone workflow sends work items to the guest agent.
`VirtualMachine.Admin.CustomizeGuestOSDelay`	Specifies the time to wait after customization is complete and before starting the guest operating system customization. The value must be in HH:MM:SS format. If the value is not set, the default value is one minute (00:01:00). If you choose not to include this custom property, provisioning can fail if the virtual machine reboots before guest agent work items are completed, causing provisioning to fail.
`VirtualMachine.Customize.WaitComplete`	Set to True to prevent the provisioning workflow from sending work items to the guest agent until all customizations arecomplete. Set to False to allow work items to be created before customization is complete.
`VirtualMachine.SoftwareN.ScriptPath`	Specifies the full path to an application's install script. The path must be a valid absolute path as seen by the guest operating system and must include the name of the script filename. You can pass custom property values as parameters to the script by inserting {*CustomPropertyName*} in the path string. For example, if you have a custom property named `ActivationKey` whose value is 1234, the script path is `D:\InstallApp.bat -key {ActivationKey}`. The guest agent runs the command `D:\InstallApp.bat -key 1234`. Your script file can then be programmed to accept and use this value.

Custom Properties for FlexClone Blueprints

vRealize Automation includes custom properties that you can use to provide additional controls for FlexClone blueprints.

Table 8-8. Custom Properties for FlexClone Blueprints

Custom Property	Description
`VirtualMachine.NetworkN.NetworkProfileName`	Specifies the name of a network profile from which to assign a static IP address to network device *N* or from which to obtain the range of static IP addresses that can be assigned to network device *N* of a cloned machine, where *N*=0 for the first device, 1 for the second, and so on. The network profile that the property points to is used to allocate an IP address. The property determines the network that the machine attaches to, based on the reservation. Changing this property value after the network is assigned has no effect on the expected IP address values for the designated machines. With WIM-based provisioning for virtual machines, you can use this property to specify a network profile and network interface or you can use the Network section of the Virtual Reservation page. The following attributes of the network profile are available to enable static IP assignment in a cloning blueprint: ■ `VirtualMachine.NetworkN.SubnetMask` ■ `VirtualMachine.NetworkN.Gateway` ■ `VirtualMachine.NetworkN.PrimaryDns` ■ `VirtualMachine.NetworkN.SecondaryDns` ■ `VirtualMachine.NetworkN.PrimaryWins` ■ `VirtualMachine.NetworkN.SecondaryWins` ■ `VirtualMachine.NetworkN.DnsSuffix` ■ `VirtualMachine.NetworkN.DnsSearchSuffixes` `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation.
`Linux.ExternalScript.Name`	Specifies the name of an optional customization script, for example `config.sh`, that the Linux guest agent runs after the operating system is installed. This property is available for Linux machines cloned from templates on which the Linux agent is installed. If you specify an external script, you must also define its location by using the `Linux.ExternalScript.LocationType` and `Linux.ExternalScript.Path` properties.

Table 8-8. Custom Properties for FlexClone Blueprints (continued)

Custom Property	Description
`Linux.ExternalScript.LocationType`	Specifies the location type of the customization script named in the `Linux.ExternalScript.Name` property. This can be either local or nfs. You must also specify the script location using the `Linux.ExternalScript.Path` property. If the location type is nfs, also use the `Linux.ExternalScript.Server` property.
`Linux.ExternalScript.Server`	Specifies the name of the NFS server, for example lab-ad.lab.local, on which the Linux external customization script named in `Linux.ExternalScript.Name` is located.
`Linux.ExternalScript.Path`	Specifies the local path to the Linux customization script or the export path to the Linux customization on the NFS server. The value must begin with a forward slash and not include the file name, for example `/scripts/linux/config.sh`.

If you installed the guest agent to customize cloned machines, the Custom Properties for Customizing FlexClone Machines with a Guest Agent table describes the most commonly used custom properties for your situation.

Table 8-9. Custom Properties for Customizing FlexClone Machines with a Guest Agent

Custom Property	Description
`VirtualMachine.Admin.UseGuestAgent`	If the guest agent is installed as a service on a template for cloning, set to True on the machine blueprint to enable the guest agent service on machines cloned from that template. When the machine is started, the guest agent service is started. Set to False to deactivate the guest agent. If set to False, the enhanced clone workfow will not use the guest agent for guest operating system tasks, reducing its functionality to `VMwareCloneWorkflow`. If not specified or set to anything other than False, the enhanced clone workflow sends work items to the guest agent.
`VirtualMachine.DiskN.Label`	Specifies the label for a machine's disk *N*. The disk label maximum is 32 characters. Disk numbering must be sequential. When used with a guest agent, specifies the label of a machine's disk *N* inside the guest operating system.
`VirtualMachine.DiskN.Letter`	Specifies the drive letter or mount point of a machine's disk *N*. The default is `C`. For example, to specify the letter D for Disk 1, define the custom property as `VirtualMachine.Disk1.Letter` and enter the value D. Disk numbering must be sequential. When used in conjunction with a guest agent, this value specifies the drive letter or mount point under which an additional disk *N* is mounted by the guest agent in the guest operating system.

Table 8-9. Custom Properties for Customizing FlexClone Machines with a Guest Agent (continued)

Custom Property	Description
`VirtualMachine.Admin.CustomizeGuestOSDelay`	Specifies the time to wait after customization is complete and before starting the guest operating system customization. The value must be in HH:MM:SS format. If the value is not set, the default value is one minute (00:01:00). If you choose not to include this custom property, provisioning can fail if the virtual machine reboots before guest agent work items are completed, causing provisioning to fail.
`VirtualMachine.Customize.WaitComplete`	Set to True to prevent the provisioning workflow from sending work items to the guest agent until all customizations arecomplete. Set to False to allow work items to be created before customization is complete.
`VirtualMachine.SoftwareN.ScriptPath`	Specifies the full path to an application's install script. The path must be a valid absolute path as seen by the guest operating system and must include the name of the script filename. You can pass custom property values as parameters to the script by inserting {*CustomPropertyName*} in the path string. For example, if you have a custom property named `ActivationKey` whose value is 1234, the script path is `D:\InstallApp.bat –key {ActivationKey}`. The guest agent runs the command `D:\InstallApp.bat –key 1234`. Your script file can then be programmed to accept and use this value.

Custom Properties for Basic Workflow Blueprints

vRealize Automation includes custom properties that you can use to provide additional controls for basic workflow blueprints.

Table 8-10. Custom Properties for Basic Workflow Blueprints

Custom Property	Description
`VirtualMachine.CDROM.Attach`	Set to False to provision the machine without a CD-ROM device. The default is True.
`VirtualMachine.Admin.ThinProvision`	Determines whether thin provisioning is used on ESX compute resources. Disk provisioning is abstracted from the underlying storage. Set to True to use thin provisioning. Set to False to use standard provisioning. This property is for virtual provisioning.

Table 8-10. Custom Properties for Basic Workflow Blueprints (continued)

Custom Property	Description
`VirtualMachine.DiskN.StorageReservationPolicy`	Specifies the storage reservation policy to use to find storage for disk *N*. Also assigns the named storage reservation policy to a volume. To use this property, substitute the volume number for *N* in the property name and specify a storage reservation policy name as the value. This property is equivalent to the storage reservation policy name specified on the blueprint. Disk numbering must be sequential. This property is valid for all Virtual and vCloud reservations. This property is not valid for Physical, Amazon, or OpenStack reservations.
`VirtualMachine.Storage.AllocationType`	Stores collected groups to a single datastore. A distributed environment stores disks round-robin style. Specify one of the following values: ■ Collected Keep all disks together. ■ Distributed Allow disks to be placed on any datastore or datastore cluster that is available in the reservation. For an example of how to use the `VirtualMachine.Storage.AllocationType` property to create datastore clusters, see the Keeping Multiple Disks Together blog post.
`VirtualMachine.Storage.Name`	Identifies the storage path on which the machine resides. The default is the value specified in the reservation that was used to provision the machine.
`VirtualMachine.Storage.ReserveMemory`	Set to True to manage vSwap storage allocation to ensure availability and set allocation in the reservation. vSwap allocation is considered when you create or reconfigure a virtual machine. vSwap allocation checking is only available for vSphere endpoints. **Note** If you do not specify the `VirtualMachine.Storage.ReserveMemory` custom property when you create or provision the machine from vRealize Automation, swap space availability is not ensured. If you add the property for an already provisioned machine, and the allocated reservation is full, the storage allocated in the reservation might exceed the actual allocated storage.
`VMware.Hardware.Version`	Specifies the VM hardware version to be used for vSphere settings. Supported values are currently vmx-04, vmx-07, vmx-08, vmx-09 and vmx-10. This property is applicable for VM Create and VM Update workflows and is available only for basic workflow blueprints.

Custom Properties for Linux Kickstart Blueprints

vRealize Automation includes custom properties that you can use to provide additional controls for Linux Kickstart blueprints.

Certain vRealize Automation custom properties are required to use with Linux Kickstart blueprints.

Table 8-11. Required Custom Properties for Linux Kickstart Blueprints

Custom Property	Description
`VMware.VirtualCenter.OperatingSystem`	Specifies the vCenter Server guest operating system version (`VirtualMachineGuestOsIdentifier`) with which vCenter Server creates the machine. This operating system version must match the operating system version to be installed on the provisioned machine. Administrators can create property groups using one of several property sets, for example, `VMware[OS_Version]Properties`, that are predefined to include the correct `VMware.VirtualCenter.OperatingSystem` values. This property is for virtual provisioning. For related information, see the enumeration type `VirtualMachineGuestOsIdentifier` in vSphere API/SDK Documentation. For a list of currently accepted values, see the vCenter Server documentation.
`Image.ISO.Location`	Values for this property are case sensitive. Specifies the location of the ISO image from which to boot. The format of this value depends on your platform. For details, see documentation provided for your platform. This property is required for WIM-based provisioning, Linux Kickstart and autoYaST provisioning, and SCCM-based provisioning.
`Image.ISO.Name`	Values for this property are case sensitive. Specifies the name of the ISO image from which to boot, for example /ISO/Microsoft/WinPE.iso. The format of this value depends on your platform. For details, see documentation provided for your platform. This property is required for WIM-based provisioning, Linux Kickstart and autoYaST provisioning, and SCCM-based provisioning.
`Image.ISO.UserName`	Specifies the user name to access the CIFS share in the format *username@domain*. For Dell iDRAC integrations where the image is located on a CIFS share that requires authentication to access.
`Image.ISO.Password`	Specifies the password associated with the `Image.ISO.UserName` property. For Dell iDRAC integrations where the image is located on a CIFS share that requires authentication to access.

Optional custom properties are available.

Table 8-12. Optional Custom Properties for Linux Kickstart Blueprints

Custom Property	Description
`VirtualMachine.Admin.ThinProvision`	Determines whether thin provisioning is used on ESX compute resources. Disk provisioning is abstracted from the underlying storage. Set to True to use thin provisioning. Set to False to use standard provisioning. This property is for virtual provisioning.
`Machine.SSH`	Set to True to enable the **Connect Using SSH** option on the vRealize Automation Items page for Linux machines provisioned from this blueprint. If set to True and the **Connect using RDP or SSH** machine operation is enabled in the blueprint, all Linux machines that are provisioned from the blueprint display the **Connect Using SSH** option to entitled users. The **Connect Using SSH** option requires that your browser has a plug-in that supports SSH, for example the FireSSH SSH terminal client for Mozilla Firefox and Google Chrome. When the plug-in is present, selecting **Connect Using SSH** displays an SSH console and prompts for your administrator credentials.

Custom Properties for SCCM Blueprints

vRealize Automation includes custom properties that you can use to provide additional controls for SCCM blueprints.

Certain custom properties are required to use with SCCM blueprints.

Table 8-13. Required Custom Properties for SCCM Blueprints

Custom Property	Description
`Image.ISO.Location`	Values for this property are case sensitive. Specifies the location of the ISO image from which to boot. The format of this value depends on your platform. For details, see documentation provided for your platform. This property is required for WIM-based provisioning, Linux Kickstart and autoYaST provisioning, and SCCM-based provisioning.
`Image.ISO.Name`	Values for this property are case sensitive. Specifies the name of the ISO image from which to boot, for example /ISO/Microsoft/WinPE.iso. The format of this value depends on your platform. For details, see documentation provided for your platform. This property is required for WIM-based provisioning, Linux Kickstart and autoYaST provisioning, and SCCM-based provisioning.
`Image.ISO.UserName`	Specifies the user name to access the CIFS share in the format *username@domain*. For Dell iDRAC integrations where the image is located on a CIFS share that requires authentication to access.

Table 8-13. Required Custom Properties for SCCM Blueprints (continued)

Custom Property	Description
`Image.ISO.Password`	Specifies the password associated with the `Image.ISO.UserName` property. For Dell iDRAC integrations where the image is located on a CIFS share that requires authentication to access.
`SCCM.Collection.Name`	Specifies the name of the SCCM collection that contains the operating system deployment task sequence.
`SCCM.Server.Name`	Specifies the fully qualified domain name of the SCCM server on which the collection resides, for example lab-sccm.lab.local.
`SCCM.Server.SiteCode`	Specifies the site code of the SCCM server.
`SCCM.Server.UserName`	Specifies a user name with administrator-level access to the SCCM server.
`SCCM.Server.Password`	Specifies the password associated with the `SCCM.Server.UserName` property.

Certain custom properties are used most often with SCCM blueprints.

Table 8-14. Common Custom Properties for SCCM Blueprints

Custom Property	Description
`SCCM.CustomVariable.`*Name*	Specifies the value of a custom variable, where *Name* is the name of any custom variable to be made available to the SCCM task sequence after the provisioned machine is registered with the SCCM collection. The value is determined by your choice of custom variable. If your integration requires it, you can use `SCCM.RemoveCustomVariablePrefix` to remove the `SCCM.CustomVariable.` prefix from your custom variable.
`SCCM.RemoveCustomVariablePrefix`	Set to *true* to remove the prefix `SCCM.CustomVariable.` from SCCM custom variables you created by using the custom property `SCCM.CustomVariable.`*Name*.

Custom Properties for WIM Blueprints

vRealize Automation includes custom properties that provide additional controls for WIM blueprints.

Certain vRealize Automation custom properties are required for WIM blueprints.

Table 8-15. Required Custom Properties for WIM Blueprints

Custom Property	Description
`Image.ISO.Location`	Values for this property are case sensitive. Specifies the location of the ISO image from which to boot. The format of this value depends on your platform. For details, see documentation provided for your platform. This property is required for WIM-based provisioning, Linux Kickstart and autoYaST provisioning, and SCCM-based provisioning.
`Image.ISO.Name`	Values for this property are case sensitive. Specifies the name of the ISO image from which to boot, for example /ISO/Microsoft/WinPE.iso. The format of this value depends on your platform. For details, see documentation provided for your platform. This property is required for WIM-based provisioning, Linux Kickstart and autoYaST provisioning, and SCCM-based provisioning.
`Image.ISO.UserName`	Specifies the user name to access the CIFS share in the format *username@domain*. For Dell iDRAC integrations where the image is located on a CIFS share that requires authentication to access.
`Image.ISO.Password`	Specifies the password associated with the `Image.ISO.UserName` property. For Dell iDRAC integrations where the image is located on a CIFS share that requires authentication to access.
`Image.Network.Letter`	Specifies the drive letter to which the WIM image path is mapped on the provisioned machine. The default value is K.
`Image.WIM.Path`	Specifies the UNC path to the WIM file from which an image is extracted during WIM-based provisioning. The path format is \\*server**share$* format, for example `\\lab-ad\dfs$`.
`Image.WIM.Name`	Specifies the name of the WIM file as located by the `Image.WIM.Path` property.
`Image.WIM.Index`	Specifies the index used to extract the correct image from the WIM file.
`Image.Network.User`	Specifies the user name with which to map the WIM image path (`Image.WIM.Path`) to a network drive on the provisioned machine. This is typically a domain account with access to the network share.
`Image.Network.Password`	Specifies the password associated with the `Image.Network.User` property.

Table 8-15. Required Custom Properties for WIM Blueprints (continued)

Custom Property	Description
`VirtualMachine.Admin.Owner`	Specifies the user name of the machine owner.
`VMware.VirtualCenter.OperatingSystem`	Specifies the vCenter Server guest operating system version (`VirtualMachineGuestOsIdentifier`) with which vCenter Server creates the machine. This operating system version must match the operating system version to be installed on the provisioned machine. Administrators can create property groups using one of several property sets, for example, `VMware[OS_Version]Properties`, that are predefined to include the correct `VMware.VirtualCenter.OperatingSystem` values. This property is for virtual provisioning. For related information, see the enumeration type `VirtualMachineGuestOsIdentifier` in vSphere API/SDK Documentation. For a list of currently accepted values, see the vCenter Server documentation.

Optional custom properties are also available for WIM blueprints.

Table 8-16. Common Custom Properties for WIM Blueprints

Custom Property	Description
`SysPrep.`*Section.Key* - `SysPrep.GuiUnattended.AdminPassword` - `SysPrep.GuiUnattended.EncryptedAdminPassword` - `SysPrep.GuiUnattended.TimeZone`	Specifies information to be added to the SysPrep answer file on machines during the WinPE stage of provisioning. Information that already exists in the SysPrep answer file is overwritten by these custom properties. *Section* represents the name of the section of the SysPrep answer file, for example GuiUnattended or UserData. *Key* represents a key name in the section. For example, to set the time zone of a provisioned machine to West Pacific Standard Time, define the custom property `GuiUnattended.UserData.TimeZone` and set the value to 275. For a full list of sections, keys, and accepted values, see the System Preparation Utility for Windows documentation. The following *Section.Key* combinations can be specified for WIM-based provisioning: - GuiUnattended - AdminPassword - EncryptedAdminPassword - TimeZone - UserData - ProductKey - FullName - ComputerName - OrgName - Identification - DomainAdmin - DomainAdminPassword - JoinDomain - JoinWorkgroup
`Sysprep.Identification.DomainAdmin`	Specifies a user name with administrator-level access to the target domain in Active Directory. Do not include the user domain in the credentials that you send to vCloud Director or vCloud Air.
`Sysprep.Identification.DomainAdminPassword`	Specifies the password to associate with the `Sysprep.Identification.DomainAdmin` property.
`Sysprep.Identification.JoinDomain`	Specifies the name of the domain to join in Active Directory.
`Sysprep.Identification.JoinWorkgroup`	Specifies the name of the workgroup to join if not using a domain.
`SysPrep.UserData.ComputerName`	Specifies a machine name, for example lab-client005.
`SysPrep.UserData.FullName`	Specifies the full name of a user.
`SysPrep.UserData.OrgName`	Specifies the organization name of the user.

Table 8-16. Common Custom Properties for WIM Blueprints (continued)

Custom Property	Description
`SysPrep.UserData.ProductKey`	Specifies the Windows product key.
`VirtualMachine.Admin.ThinProvision`	Determines whether thin provisioning is used on ESX compute resources. Disk provisioning is abstracted from the underlying storage. Set to True to use thin provisioning. Set to False to use standard provisioning. This property is for virtual provisioning.

Custom Properties for vCloud Air and vCloud Director Blueprints

You can add certain custom properties to a vCloud Air or vCloud Director machine component definition in a blueprint.

For vSphere machine components with associated NSX, use network, security, and load balancing setting in the user interface. For machine components that do not have a **Network** or **Security** tab, you can add network and security custom properties, such as `VirtualMachine.Network0.Name`, to their **Properties** tab in the design canvas. NSX network, security, and load balancer properties are only applicable to vSphere machines.

Table 8-17. Custom Properties for vCloud Air and vCloud Director Machine Components in the Design Canvas

Custom Property	Description
`Sysprep.Identification.DomainAdmin`	Specifies a user name with administrator-level access to the target domain in Active Directory. Do not include the user domain in the credentials that you send to vCloud Director or vCloud Air.
`Sysprep.Identification.DomainAdminPassword`	Specifies the password to associate with the `Sysprep.Identification.DomainAdmin` property.
`Sysprep.Identification.JoinDomain`	Specifies the name of the domain to join in Active Directory.
`VirtualMachine.DiskN.IsFixed`	Deactivates the editing of a specific disk when reconfiguring a machine. Set to True to deactivate display of the edit capacity option for a specific volume. The True value is case-sensitive. The *N* value is the 0-based index of the disk. Alternatively, you can set the `VirtualMachine.Disk`*`N`*`.IsFixed` custom property to True in the VirtualMachineProperties table in the database or use the Repository API to specify a URI value such as `.../Repository/Data/ManagementModelEntities.svc/VirtualMachines(guid'60D93A8A-F541-4CE0-A6C6-78973AC0F1D2')/VirtualMachineProperties`.

Table 8-17. Custom Properties for vCloud Air and vCloud Director Machine Components in the Design Canvas (continued)

Custom Property	Description
`VirtualMachine.DiskN.StorageReservationPolicy`	Specifies the storage reservation policy to use to find storage for disk *N*. Also assigns the named storage reservation policy to a volume. To use this property, substitute the volume number for *N* in the property name and specify a storage reservation policy name as the value. This property is equivalent to the storage reservation policy name specified on the blueprint. Disk numbering must be sequential. This property is valid for all Virtual and vCloud reservations. This property is not valid for Physical, Amazon, or OpenStack reservations.
`VirtualMachine.EULA.AcceptAll`	Set to true to specify that all the EULAs for the VM templates of the vCloud Air or vCloud Director endpoints are accepted during provisioning.
`VirtualMachine.NetworkN.Name`	Specifies the name of the network to connect to, for example the network device *N* to which a machine is attached. This is equivalent to a network interface card (NIC). By default, a network is assigned from the network paths available on the reservation on which the machine is provisioned. Also see `VirtualMachine.NetworkN.AddressType`. You can ensure that a network device is connected to a specific network by setting the value of this property to the name of a network on an available reservation. For example, if you give properties for N= 0 and 1, you get 2 NICs and their assigned value, provided the network is selected in the associated reservation. `VirtualMachine.Network`*N* custom properties are specific to blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks. For an example of how to use this custom property to dynamically set `VirtualMachine.Network0.Name` based on a consumer's selection from a list of predefined available networks, see the Adding a Network Selection Drop-Down in vRA 7 blog post. For related information, see Custom Properties for Networking and Security.

Table 8-17. Custom Properties for vCloud Air and vCloud Director Machine Components in the Design Canvas (continued)

Custom Property	Description
`VirtualMachine.NetworkN.AddressType`	Specifies how IP address allocation is supplied to the network provider, where Network*N* is the network number, starting with 0. The following values are available: - DHCP - Static - MANUAL (available for vCloud Air and vCloud Director only) This property is available for configuring vCloud Air, vCloud Director, and vSphere machine components in the blueprint. Also see `VirtualMachine.NetworkN.Name`. This property is not supported for on-demand NAT or on-demand routed networks.
`VirtualMachine.Reconfigure.DisableHotCpu`	Set to true to specify that the reconfigure machine action restarts the specified machine. By default, the reconfigure machine action does not restart the machine. Performing a hot add of CPU, memory, or storage causes the reconfigure machine action to fail and not to restart the machine unless the `Hot Add` setting is enabled in vSphere for the machine or template. You can add `VirtualMachine.Reconfigure.DisableHotCpu=true` to a machine component in a vRealize Automation blueprint to deactivate the `Hot Add` setting and force the machine to restart regardless of the vSphere `Hot Add` setting. The custom property is only available for machine types that support hardware reconfiguration, which are vSphere, vCloud Air, and vCloud Director.
`VCloud.Lease.Sync.TimeBufferMins`	Specifies a threshold integer value for a compute resource such that lease synchronization between vCloud Director and vRealize Automation only occur for vCloud Director or vCloud Air-provisioned machines that are set to expire in vCloud Director or vCloud Air in that time period. If a conflict is found, the lease value is synchronized to match the lease length defined in vRealize Automation. The default `VCloud.Lease.Sync.TimeBufferMins` value is 720 minutes, which is 12 hours. If `VCloud.Lease.Sync.TimeBufferMins` is not present, the default value is used. For example, if the default values are used, vRealize Automation runs the lease synchronization check workflow every 45 minutes, which is the workflow default, and only the leases of machines that are set to expire within 12 hours are changed to match the lease length defined in vRealize Automation.

Table 8-17. Custom Properties for vCloud Air and vCloud Director Machine Components in the Design Canvas (continued)

Custom Property	Description
`VCloud.Owner.UseEndpointAccount`	Set to true to assign the endpoint account as the vCloud Air or vCloud Director machine owner for provisioning and import operations. For change ownership operations, the owner is not changed on the endpoint. If not specified or set to false, the vRealize Automation owner is the machine owner.
`VCloud.Template.MakeIdenticalCopy`	Set to true to clone an identical copy of the vCloud Air or vCloud Director template for machine provisioning. The machine is provisioned as an identical copy of the template. Settings specified in the template, including storage path, supersede settings specified in the blueprint. The only changes from the template are the names of the cloned machines, which are generated from the machine prefix specified in the blueprint. vCloud Air or vCloud Director machines that are provisioned as identical copies can use networks and storage profiles that are not available in the vRealize Automation reservation. To avoid having unaccounted reservation allocations, verify that the storage profile or network specified in the template is available in the reservation.

Table 8-17. Custom Properties for vCloud Air and vCloud Director Machine Components in the Design Canvas (continued)

Custom Property	Description
`VMware.SCSI.Sharing`	Specifies the sharing mode of the machine's VMware SCSI bus. Possible values are based on the `VirtualSCSISharing` ENUM value and include noSharing, physicalSharing, and virtualSharing. The `VMware.SCSI.Sharing` property is not available for use with the CloneWorkflow provisioning workflow. If you specify the CloneWorkflow provisioning workflow when configuring your machine component in the blueprint design canvas, you cannot use the `VMware.SCSI.Sharing` property.
`VMware.SCSI.Type`	For vCloud Air, vCloud Director, or vSphere machine components in blueprints, specifies the SCSI machine type using one of the following case-sensitive values: ■ buslogic Use BusLogic emulation for the virtual disk. ■ lsilogic Use LSILogic emulation for the virtual disk (default). ■ lsilogicsas Use LSILogic SAS 1068 emulation for the virtual disk. ■ pvscsi Use para-virtualization emulation for the virtual disk. ■ none Use if a SCSI controller does not exist for this machine. The `VMware.SCSI.Type` property is not available for use with the CloneWorkflow provisioning workflow. If you specify the CloneWorkflow provisioning workflow when configuring your machine component in the blueprint design canvas, you cannot use the `VMware.SCSI.Type` property.

Custom Properties for Networking and Security

The vRealize Automation custom properties for networking specify configuration for a specific network device on a machine.

For vSphere machine components with associated NSX, use network, security, and load balancing setting in the user interface. For machine components that do not have a **Network** or **Security** tab, you can add network and security custom properties, such as `VirtualMachine.Network0.Name`, to their **Properties** tab in the design canvas. NSX network, security, and load balancer properties are only applicable to vSphere machines.

Network assignments are performed during machine allocation. vRealize Automation retrieves network information from the blueprint. If you want to assign more than one network, use the `VirtualMachine.NetworkN.Name` custom property on your machine blueprint. If you do not provide custom properties, allocation only assigns one network which is picked using a round robin method in conjunction with the selected reservation.

Note With the exception of the following properties, the properties in the table do not apply to Amazon Web Services:

- `agent.download.url`
- `software.agent.service.url`
- `software.ebs.url`

Note Network-specific custom properties that point to on-demand networks are not supported. For example, you cannot use network custom properties, such as `VirtualMachine.Network0.NetworkProfileName`, for on-demand NAT and on-demand routed network components.

By default, a machine has one network device configured with the `VirtualMachine.Network0.Name` property. You can configure additional network devices by using the `VirtualMachine.NetworkN.Name` custom property, where *N* is the network number.

The numbering of network properties must be sequential, starting with 0. For example, if you specify custom properties for only `VirtualMachine.Network0` and `VirtualMachine.Network2`, the properties for `VirtualMachine.Network2` are ignored, because the preceding network, `VirtualMachine.Network1`, was not specified.

Although general support for vCloud Networking and Security has ended, the VCNS custom properties continue to be valid for NSX purposes. See the Knowledge Base article 2144733.

Table 8-18. Custom Properties for Networking Configuration

Custom Property	Description
`agent.download.url`	When using port forwarding, specifies the private IP address of your Amazon AWS tunnel machine and port for your software agent file, for example **`https://`*`Private_IP:1443`*`/software-service/resources/nobel-agent.jar`**. Specifies the URL for the VPN agent on your deployment. The URL format is `https://` *`Private_IP`*`:1443/software-service//resources/noble-agent.jar` You can add this property, in combination with `software.agent.service.url` and `software.ebs.url`, to a reservation or the compute resource endpoint. You can also use this property to specify a private address and port when using PAT or NAT translation and port forwarding.
`NSX.Edge.ApplianceSize`	Specifies the allowed NSX edge appliance size types for the provisioned machine or deployment. The options are: ■ `compact` For small deployments, POCs, and single service use. ■ CPU = 1 ■ RAM = 512 MB ■ Disk = 512 MB ■ `large` For small to medium or multi-tenant deployments. ■ CPU = 2 ■ RAM = 1 GB ■ Disk = 512 MB ■ `quadlarge` For high throughput equal-cost multi-path routing (ECMP) or high performance firewall deployments. ■ CPU = 4 ■ RAM = 1 GB ■ Disk = 512 MB ■ `xlarge` For L7 load balancing and dedicated core deployments. ■ CPU = 6 ■ RAM = 8 GB ■ Disk = 4.5GB (4GB Swap) For related information, see System Requirements for NSX.

Table 8-18. Custom Properties for Networking Configuration (continued)

Custom Property	Description
`NSX.Edge.HighAvailability`	When set to true (`NSX.Edge.HighAvailability`=true), enables high availability (HA) mode on the NSX edge machine that is deployed from the blueprint. When used with `NSX.Edge.HighAvailability.PortGroup`=*port_group_name*, this property allows you to configure an NSX edge during blueprint authoring. You can add this property to an NSX load balancer component in the vRealize Automation blueprint or to the vRealize Automation blueprint itself. Must be used in conjunction with `NSX.Edge.HighAvailability.PortGroup`= *port_group_name*.
`NSX.Edge.HighAvailability.PortGroup`	Creates an internal interface or internal vNIC attached to the specified port group name, for example `NSX.Edge.HighAvailability.PortGroup`=`VM Network` where `VM Network` is an HA (high availability) distributed (vLAN-backed) or NSX logical switch port group. NSX HA mode requires at least one internal network interface, or vNIC. When used with `NSX.Edge.HighAvailability`=true, this property allows you to configure high availability (HA) an NSX edge during blueprint authoring. When using one arm load balancer with HA enabled, you must specify a separate port group for the HA. **Note** The specified port group network cannot be a member of the reservation pool, as the property's use of the port group conflicts with the normal deployment's use of the port group, resulting in the following error: `Portgroup must be unique within an Edge...` Must be used in conjunction with `NSX.Edge.HighAvailability`=true.

Table 8-18. Custom Properties for Networking Configuration (continued)

Custom Property	Description
`NSX.Validation.Disable.Single.Edge.Uplink`	When set to true, the NSX validation that checks for the following conditions is deactivated: ■ All on-demand NAT networks on the blueprint source the same external network. ■ All on-demand routed networks on the blueprint that use the load balancer VIP source the same external network. ■ All on-demand load balancer components on the blueprint have VIPs on the same external network or on-demand networks backed by the same external network. Disabling this validation check can result in a deployment that succeeds but in which some network components might be inaccessible. If not present or if set to false, the validation check is enabled (default). A single NSX edge can only support one external network as its uplink network. Multiple IPs from the same external network are supported. While a blueprint can contain any number of external or on-demand network components, NSX only supports one external network as the uplink network. This property can only be specified at the blueprint level. It cannot be specified on a component in the blueprint canvas.
`NSX.Validation.Disable.Blueprint.NSXT`	When set to true, all NSX-T validation is deactivated for the blueprint **Finish** action. If not present or if set to false, the NSX-T validation check is enabled (default). For example, if you have overlapping subnets in the blueprint, an error message appears when you click **Finish** in the blueprint and the overlap prevents you from finishing the blueprint, although you can save it. If you want to finish the blueprint, you can add `NSX.Validation.Disable.Blueprint.NSXT` by using the **Blueprint Properties** page and then finish the blueprint. The property only deactivates NSX-T validations for the blueprint **Finish** action.
`software.agent.service.url`	When using port forwarding, specifies the private IP address of your Amazon AWS tunnel machine and port for the vRealize Automation software service API, for example **`https://`*`Private_IP:1443`*`/software-service/api`**. You can add this property, in combination with `software.ebs.url` and `agent.download.url`, to a reservation or the compute resource endpoint. You can also use this property to specify a private address and port when using PAT or NAT and port forwarding.

Table 8-18. Custom Properties for Networking Configuration (continued)

Custom Property	Description
`software.ebs.url`	When using port forwarding, specifies the private IP address of your Amazon AWS tunnel machine and port for the vRealize Automation event broker service, for example **`https://`*`Private_IP:1443`*`/event-broker-service/api`**. You can add this property, in combination with `software.agent.service.url` and `agent.download.url`, to a reservation or the compute resource endpoint. You can also use this property to specify a private address and port when using PAT or NAT and port forwarding.
`VirtualMachine.NetworkN.Address`	Specifies the IP address of network device *N* in a machine provisioned with a static IP address. For Amazon, see `Amazon.elasticIpAddress.ipAddress` .
`VirtualMachine.NetworkN.MacAddressType`	Indicates whether the MAC address of network device *N* is generated or user-defined (static). This property is available for cloning. The default value is generated. If the value is static, you must also use `VirtualMachine.NetworkN.MacAddress` to specify the MAC address. `VirtualMachine.Network`*`N`* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks.
`VirtualMachine.NetworkN.MacAddress`	Specifies the MAC address of a network device *N*. This property is available for cloning. If the value of `VirtualMachine.NetworkN.MacAddressType` is generated, this property contains the generated address. If the value of `VirtualMachine.NetworkN.MacAddressType` is static, this property specifies the MAC address. For virtual machines provisioned on ESX server hosts, the address must be in the range specified by VMware. For details, see vSphere documentation. `VirtualMachine.Network`*`N`* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks.

Table 8-18. Custom Properties for Networking Configuration (continued)

Custom Property	Description
`VirtualMachine.NetworkN.Name`	Specifies the name of the network to connect to, for example the network device *N* to which a machine is attached. This is equivalent to a network interface card (NIC). By default, a network is assigned from the network paths available on the reservation on which the machine is provisioned. Also see `VirtualMachine.NetworkN.AddressType`. You can ensure that a network device is connected to a specific network by setting the value of this property to the name of a network on an available reservation. For example, if you give properties for N= 0 and 1, you get 2 NICs and their assigned value, provided the network is selected in the associated reservation. `VirtualMachine.Network`*N* custom properties are specific to blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks. For an example of how to use this custom property to dynamically set `VirtualMachine.Network0.Name` based on a consumer's selection from a list of predefined available networks, see the Adding a Network Selection Drop-Down in vRA 7 blog post.
`VirtualMachine.NetworkN.PortID`	Specifies the port ID to use for network device *N* when using a dvPort group with a vSphere distributed switch. `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks.

Table 8-18. Custom Properties for Networking Configuration (continued)

Custom Property	Description
`VirtualMachine.NetworkN.NetworkProfileName`	Specifies the name of a network profile from which to assign a static IP address to network device *N* or from which to obtain the range of static IP addresses that can be assigned to network device *N* of a cloned machine, where *N*=0 for the first device, 1 for the second, and so on. The network profile that the property points to is used to allocate an IP address. The property determines the network that the machine attaches to, based on the reservation. Changing this property value after the network is assigned has no effect on the expected IP address values for the designated machines. With WIM-based provisioning for virtual machines, you can use this property to specify a network profile and network interface or you can use the Network section of the Virtual Reservation page. The following attributes of the network profile are available to enable static IP assignment in a cloning blueprint: ■ `VirtualMachine.NetworkN.SubnetMask` ■ `VirtualMachine.NetworkN.Gateway` ■ `VirtualMachine.NetworkN.PrimaryDns` ■ `VirtualMachine.NetworkN.SecondaryDns` ■ `VirtualMachine.NetworkN.PrimaryWins` ■ `VirtualMachine.NetworkN.SecondaryWins` ■ `VirtualMachine.NetworkN.DnsSuffix` ■ `VirtualMachine.NetworkN.DnsSearchSuffixes` `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. You cannot use this custom property to define an on-demand NAT or on-demand routed network profile name. Because on-demand network profile names are generated at allocation time (during provisioning), their names are unknown when creating or editing the blueprint. To specify NSX on-demand network information, use the applicable network component in the blueprint design canvas for your vSphere machine components.

Table 8-18. Custom Properties for Networking Configuration (continued)

Custom Property	Description
■ `VirtualMachine.NetworkN.SubnetMask` ■ `VirtualMachine.NetworkN.Gateway` ■ `VirtualMachine.NetworkN.PrimaryDns` ■ `VirtualMachine.NetworkN.SecondaryDns` ■ `VirtualMachine.NetworkN.PrimaryWins` ■ `VirtualMachine.NetworkN.SecondaryWins` ■ `VirtualMachine.NetworkN.DnsSuffix` ■ `VirtualMachine.NetworkN.DnsSearchSuffixes`	Configures attributes of the network profile specified in `VirtualMachine.NetworkN.NetworkProfileName`. `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation.
`VCNS.LoadBalancerEdgePool.Names.`*name*	Specifies the NSX load balancing pools to which the virtual machine is assigned during provisioning. The virtual machine is assigned to all service ports of all specified pools. The value is an *edge/pool* name or a list of *edge/pool* names separated by commas. Names are case-sensitive. **Note** You can add a machine IP address to an existing load balancer by using the `VCNS.LoadBalancerEdgePool.Names` custom property. vRealize Automation and NSX use the first member of the specified edge load balancer pool to determine the new member port and monitor port settings. However, NSX 6.2 does not require that the member port setting be specified. To avoid provisioning failure when using `VCNS.LoadBalancerEdgePool.Names` with NSX 6.2 to add a machine to an existing pool, you must specify a port value for the first member of the load balancer pool in NSX. Appending a name allows you to create multiple versions of a custom property. For example, the following properties might list load balancing pools set up for general use and machines with high, moderate, and low performance requirements: ■ `VCNS.LoadBalancerEdgePool.Names` ■ `VCNS.LoadBalancerEdgePool.Names.moderate` ■ `VCNS.LoadBalancerEdgePool.Names.high` ■ `VCNS.LoadBalancerEdgePool.Names.low`
`VCNS.SecurityGroup.Names.`*name*	Specifies the NSX security group or groups to which the virtual machine is assigned during provisioning. The value is a security group name or a list of names separated by commas. Names are case-sensitive. Appending a name allows you to create multiple versions of the property, which can be used separately or in combination. For example, the following properties can list security groups intended for general use, for the sales force, and for support: ■ `VCNS.SecurityGroup.Names` ■ `VCNS.SecurityGroup.Names.sales` ■ `VCNS.SecurityGroup.Names.support`

Table 8-18. Custom Properties for Networking Configuration (continued)

Custom Property	Description
`VCNS.SecurityTag.Names.`*name*	Specifies the NSX security tag or tags to which the virtual machine is associated during provisioning. The value is a security tag name or a list of names separated by commas. Names are case-sensitive. Appending a name allows you to create multiple versions of the property, which can be used separately or in combination. For example, the following properties can list security tags intended for general use, for the sales force, and for support: ■ `VCNS.SecurityTag.Names` ■ `VCNS.SecurityTag.Names.sales` ■ `VCNS.SecurityTag.Names.support`
`VMware.Endpoint.NSX.HideDiscoveredSecurityObjects`	Set to true to hide newly discovered security objects in the active tenant for the NSX endpoints to which the security objects are associated. Otherwise, all new security objects are available to all tenants after data collection, provided that the security object is for an endpoint in which you have a reservation. This option allows you to prevent users from accessing security objects when you want to assign those objects to a single tenant or to mask from all tenants. Set to false to toggle back to global, which enables all new security objects to be available to all tenants after data collection, provided that the security object is for an endpoint in which you have a reservation. To take effect, the fabric administrator adds the `VMware.Endpoint.NSX.HideDiscoveredSecurityObjects` custom property to the associated NSX endpoint that is associated to a vSphere endpoint. The setting applies to the next inventory data collection. Existing security objects remain unchanged. To change the tenancy setting of a security object that has already been data-collected, such as existing security objects after upgrading to the current vRealize Automation release, you can edit the security object's Tenant ID setting programmatically by using the vRealize Automation REST API or vRealize CloudClient. The available Tenant ID settings for the NSX endpoint are as follows: ■ "`<global>`" - the security object is available to all tenants. This is the default setting for existing security objects after upgrade to this release and for all new security objects that you create. ■ "`<unscoped>`" - the security object is not available to any tenants. Only the system administrator can access the security object. This is an ideal setting when defining security objects that are to eventually be assigned to a specific tenant.

Table 8-18. Custom Properties for Networking Configuration (continued)

Custom Property	Description
	- "*tenant_id_name*" - the security object is only available to a single, named tenant. For related information, see Controlling Tenant Access for Security Objects in vRealize Automation.

Custom Properties and Property Groups for Containers

You can add predefined property groups to a containers component in a vRealize Automation blueprint. When machines are provisioned by using a blueprint that contain these properties, the provisioned machine is registered as a Docker Container host machine.

Containers for vRealize Automation supplied the following two property groups of container-specific custom properties. When you add a container component to a blueprint you can add these property groups to the container to register provisioned machines as container hosts.

- Container host properties with certificate authentication
- Container host properties with user/password authentication

These property groups are visible in vRealize Automation when you select **Administration > Property Dictionary > Property Groups**.

Because property groups are shared by all tenants, if you are working in a multi-tenant environment, consider cloning and customizing your properties. By uniquely naming property groups and properties in the groups, you can edit them to define custom values for use in a specific tenant.

The most commonly used properties are `Container.Auth.PublicKey` and `Container.Auth.PrivateKey` in which the container administrator provides the client certificate for authenticating with the container host.

Table 8-19. Containers Custom Properties

Property	Description
`containers.ipam.driver`	For use with containers only. Specifies the IPAM driver to be used when adding a Containers network component to a blueprint. The supported values depend on the drivers installed in the container host environment in which they are used. For example, a supported value might be `infoblox` or `calico` depending on the IPAM plug-ins that are installed on the container host.
`containers.network.driver`	For use with containers only. Specifies the network driver to be used when adding a Containers network component to a blueprint. The supported values depend on the drivers installed in the container host environment in which they are used. By default, Docker-supplied network drivers include bridge, overlay, and macvlan, while Virtual Container Host (VCH)-supplied network drivers include the bridge driver. Third-party network drivers such as `weave` and `calico` might also be available, depending on what network plug-ins are installed on the container host.
`Container`	For use with containers only. The default value is `App.Docker` and is required. Do not modify this property.
`Container.Auth.User`	For use with containers only. Specifies the user name for connecting to the Containers host.
`Container.Auth.Password`	For use with containers only. Specifies either the password for the user name or the public or private key password to be used. Encrypted property value is supported.
`Container.Auth.PublicKey`	For use with containers only. Specifies the public key for connecting to the Containers host.
`Container.Auth.PrivateKey`	For use with containers only. Specifies private key for connecting to the Containers host. Encrypted property value is supported.
`Container.Connection.Protocol`	For use with containers only. Specifies the communication protocol. The default value is `API` and is required. Do not modify this property.
`Container.Connection.Scheme`	For use with containers only. Specifies the communication scheme. The default is `https`.
`Container.Connection.Port`	For use with containers only. Specifies the Containers connection port. The default is 2376.

Table 8-19. Containers Custom Properties (continued)

Property	Description
`Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.MachineActivated`	For use with containers only. Specifies the event broker property to expose all Containers properties and is used for registering a provisioned host. The default value is `Container*` and is required. Do not modify this property.
`Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.Disposing`	For use with containers only. Specifies the event broker property to expose all Containersproperties above and is used for unregistering a provisioned host. The default value is `Container*` and is required. Do not modify this property.

Custom Properties for PXE Provisioning

PXE is the only provisioning method supported for Cisco UCS Manager. You can use the network bootstrap program with vRealize Automation custom properties to initiate WIM, SCCM, or Linux Kickstart provisioning. You can also use custom properties to call your own PowerShell scripts. Linux Kickstart provisioning does not require custom properties.

Custom Properties for Provisioning With PowerShell Scripts

You can use these properties for calling PowerShell scripts.

Table 8-20. Custom Properties for Calling PowerShell Scripts

Custom Property	Description
`Pxe.Setup.ScriptName`	Specifies a custom EPI PowerShell script to run on the machine before it is started by using the PXE network boot program. The value is the name assigned to the script when it is uploaded to the model manager, for example `setup.ps1`.
`Pxe.Clean.ScriptName`	Specifies the name of an EPI PowerShell script installed in the vRealize Automation Model Manager, to run on the machine after it is provisioned. The value is the name assigned to the script when it is uploaded to the Model Manager, for example `clean.ps1`.

Custom Properties For PXE and SCCM Provisioning

You can use these properties for PXE and SCCM provisioning.

Table 8-21. Custom Properties for PXE and SCCM Provisioning

Custom Property	Description
`SCCM.Collection.Name`	Specifies the name of the SCCM collection that contains the operating system deployment task sequence.
`SCCM.Server.Name`	Specifies the fully qualified domain name of the SCCM server on which the collection resides, for example lab-sccm.lab.local.

Table 8-21. Custom Properties for PXE and SCCM Provisioning (continued)

Custom Property	Description
`SCCM.Server.SiteCode`	Specifies the site code of the SCCM server.
`SCCM.Server.UserName`	Specifies a user name with administrator-level access to the SCCM server.
`SCCM.Server.Password`	Specifies the password associated with the `SCCM.Server.UserName` property.
`SCCM.CustomVariable.`	Specifies the value of a custom variable, where *Name* is the name of any custom variable to be made available to the SCCM task sequence after the provisioned machine is registered with the SCCM collection. The value is determined by your choice of custom variable. If your integration requires it, you can use `SCCM.RemoveCustomVariablePrefix` to remove the `SCCM.CustomVariable.` prefix from your custom variable.

Custom Properties For PXE and WIM Provisioning

You can use these properties for PXE and WIM provisioning.

Table 8-22. Custom Properties for PXE and WIM Provisioning

Custom Property	Description
`Image.Network.Letter`	Specifies the drive letter to which the WIM image path is mapped on the provisioned machine. The default value is K.
`Image.WIM.Path`	Specifies the UNC path to the WIM file from which an image is extracted during WIM-based provisioning. The path format is \\server\share$ format, for example `\\lab-ad\dfs$`.
`Image.WIM.Name`	Specifies the name of the WIM file as located by the `Image.WIM.Path` property.
`Image.WIM.Index`	Specifies the index used to extract the correct image from the WIM file.
`Image.Network.User`	Specifies the user name with which to map the WIM image path (`Image.WIM.Path`) to a network drive on the provisioned machine. This is typically a domain account with access to the network share.
`Image.Network.Password`	Specifies the password associated with the `Image.Network.User` property.

Table 8-22. Custom Properties for PXE and WIM Provisioning (continued)

Custom Property	Description
`SysPrep.`*Section.Key* - `SysPrep.GuiUnattended.AdminPassword` - `SysPrep.GuiUnattended.EncryptedAdminPassword` - `SysPrep.GuiUnattended.TimeZone`	Specifies information to be added to the SysPrep answer file on machines during the WinPE stage of provisioning. Information that already exists in the SysPrep answer file is overwritten by these custom properties. *Section* represents the name of the section of the SysPrep answer file, for example GuiUnattended or UserData. *Key* represents a key name in the section. For example, to set the time zone of a provisioned machine to West Pacific Standard Time, define the custom property `GuiUnattended.UserData.TimeZone` and set the value to 275. For a full list of sections, keys, and accepted values, see the System Preparation Utility for Windows documentation. The following *Section.Key* combinations can be specified for WIM-based provisioning: - GuiUnattended - AdminPassword - EncryptedAdminPassword - TimeZone - UserData - ProductKey - FullName - ComputerName - OrgName - Identification - DomainAdmin - DomainAdminPassword - JoinDomain - JoinWorkgroup
`Sysprep.Identification.DomainAdmin`	Specifies a user name with administrator-level access to the target domain in Active Directory. Do not include the user domain in the credentials that you send to vCloud Director or vCloud Air.
`Sysprep.Identification.DomainAdminPassword`	Specifies the password to associate with the `Sysprep.Identification.DomainAdmin` property.
`Sysprep.Identification.JoinDomain`	Specifies the name of the domain to join in Active Directory.
`Sysprep.Identification.JoinWorkgroup`	Specifies the name of the workgroup to join if not using a domain.
`SysPrep.UserData.ComputerName`	Specifies a machine name, for example lab-client005.
`SysPrep.UserData.FullName`	Specifies the full name of a user.

Table 8-22. Custom Properties for PXE and WIM Provisioning (continued)

Custom Property	Description
`SysPrep.UserData.OrgName`	Specifies the organization name of the user.
`SysPrep.UserData.ProductKey`	Specifies the Windows product key.

Custom Properties for OVF Import

When you import an OVF to a blueprint, you can import and configure several settings as custom properties.

For related information, see Configuring a Blueprint to Provision from an OVF.

Table 8-23. Custom Properties for Blueprints In Which An OVF Is Imported

Custom Property	Description
`VMware.Ovf.Thumbprint`	If the OVF resides on an HTTPS server that has a certificate, this property stores the value of that certificate's thumbprint and is used to validate that certificate. It has no relevance when the OVF is hosted on an HTTP server. The property is automatically created when you import an OVF by using the `ImportOvfWorkflow` provisioning workflow in the blueprint component's user interface. If you create the blueprint programmatically with vRealize Automation REST APIs or vRealize CloudClient, you must manually create the property. **Note** The thumbprint can be stored in a comma-separated format to support a certificate chain. When the `VMware.Ovf.TrustAllCertificates` is present and set to true, the `VMware.Ovf.Thumbprint` property is ignored.
`VMware.Ovf.TrustAllCertificates`	When this property is present and set to true, the `Vmware.Ovf.Thumbprint` property is ignored and no certificate validation is performed when you import an OVF by using the `ImportOvfWorkflow` provisioning workflow.
`VMware.Ovf.Configuration.`*X*	An OVF can contain user-configurable properties, for example a property that sets the root password of a VM provisioned from the OVF. When you import an OVF into a blueprint, the user-configurable properties that are defined in the OVF are parsed and converted into custom properties of the form `Vmware.Ovf.Configuration.`*X*, where *X* is the name of the user-configurable property from the OVF.

Custom Properties for vRealize Automation Guest Agent

If you have installed the vRealize Automation guest agent in your templates for cloning or in your WinPE, you can use custom properties to run custom scripts within the guest operating system of a provisioned machine after the machine is fully deployed.

For related information, see Using vRealize Automation Guest Agent in Provisioning.

Table 8-24. Custom Properties for Customizing Provisioned Machines with a Guest Agent

Custom Property	Description
`VirtualMachine.Admin.AddOwnerToAdmins`	Set to True (default) to add the machine's owner, as specified by the `VirtualMachine.Admin.Owner` property, to the local administrators group on the machine. This property is not available for provisioning by cloning.
`VirtualMachine.Admin.AllowLogin`	Set to True (default) to add the machine owner to the local remote desktop users group, as specified by the `VirtualMachine.Admin.Owner` property.
`VirtualMachine.Admin.UseGuestAgent`	If the guest agent is installed as a service on a template for cloning, set to True on the machine blueprint to enable the guest agent service on machines cloned from that template. When the machine is started, the guest agent service is started. Set to False to deactivate the guest agent. If set to False, the enhanced clone workfow will not use the guest agent for guest operating system tasks, reducing its functionality to `VMwareCloneWorkflow`. If not specified or set to anything other than False, the enhanced clone workflow sends work items to the guest agent.
`VirtualMachine.DiskN.Active`	Set to True (default) to specify that the machine's disk *N* is active. Set to False to specify that the machine's disk *N* is not active.
`VirtualMachine.DiskN.Label`	Specifies the label for a machine's disk *N*. The disk label maximum is 32 characters. Disk numbering must be sequential. When used with a guest agent, specifies the label of a machine's disk *N* inside the guest operating system.
`VirtualMachine.DiskN.Letter`	Specifies the drive letter or mount point of a machine's disk *N*. The default is C. For example, to specify the letter D for Disk 1, define the custom property as `VirtualMachine.Disk1.Letter` and enter the value D. Disk numbering must be sequential. When used in conjunction with a guest agent, this value specifies the drive letter or mount point under which an additional disk *N* is mounted by the guest agent in the guest operating system.
`VirtualMachine.DiskN.FS`	For use with Windows guest agent (gugent). Specifies the file system of the machine's disk *N*. The options are NTFS (default), FAT and FAT32. For example usage, see the `10_setupdisks.bat` Windows agent script.

Table 8-24. Custom Properties for Customizing Provisioned Machines with a Guest Agent (continued)

Custom Property	Description
`VirtualMachine.DiskN.FileSystem`	For use with Linux guest agent (gugent). Specifies the file system of the machine's disk *N*. The options are ext3, ext4, and XFS. For example usage, see the `30_DiskSetup.sh` Linux agent script.
`VirtualMachine.Admin.CustomizeGuestOSDelay`	Specifies the time to wait after customization is complete and before starting the guest operating system customization. The value must be in HH:MM:SS format. If the value is not set, the default value is one minute (00:01:00). If you choose not to include this custom property, provisioning can fail if the virtual machine reboots before guest agent work items are completed, causing provisioning to fail.
`VirtualMachine.Customize.WaitComplete`	Set to True to prevent the provisioning workflow from sending work items to the guest agent until all customizations arecomplete. Set to False to allow work items to be created before customization is complete.
`VirtualMachine.SoftwareN.Name`	Specifies the descriptive name of a software application *N* or script to install or run during provisioning. This is an optional and information-only property. It serves no real function for the enhanced clone workflow or the guest agent but it is useful for a custom software selection in a user interface or for software use reporting.
`VirtualMachine.SoftwareN.ScriptPath`	Specifies the full path to an application's install script. The path must be a valid absolute path as seen by the guest operating system and must include the name of the script filename. You can pass custom property values as parameters to the script by inserting {*CustomPropertyName*} in the path string. For example, if you have a custom property named `ActivationKey` whose value is 1234, the script path is `D:\InstallApp.bat -key {ActivationKey}`. The guest agent runs the command `D:\InstallApp.bat -key 1234`. Your script file can then be programmed to accept and use this value. Insert {Owner} to pass the machine owner name to the script. You can also pass custom property values as parameters to the script by inserting {*YourCustomProperty*} in the path string. For example, entering the value **`\\vra-scripts.mycompany.com\scripts\changeIP.bat`** runs the `changeIP.bat` script from a shared location, but entering the value **`\\vra-scripts.mycompany.com\scripts\changeIP.bat {VirtualMachine.Network0.Address}`** runs the changeIP script but also passes the value of the `VirtualMachine.Network0.Address` property to the script as a parameter.

Table 8-24. Custom Properties for Customizing Provisioned Machines with a Guest Agent (continued)

Custom Property	Description
`VirtualMachine.ScriptPath.Decrypt`	Allows vRealize Automation to obtain an encrypted string that is passed as a properly formatted `VirtualMachine.SoftwareN.ScriptPath` custom property statement to the gugent command line. You can provide an encrypted string, such as your password, as a custom property in a command-line argument. This allows you to store encrypted information that the guest agent can decrypt and understand as a valid command-line argument. For example, the `VirtualMachine.Software0.ScriptPath = c:\dosomething.bat` *`password`* custom property string is not secure as it contains an actual password. To encrypt the password, you can create a vRealize Automation custom property, for example `MyPassword = password`, and enable encryption by selecting the available check box. The guest agent decrypts the **`[MyPassword]`** entry to the value in the custom property `MyPassword` and runs the script as `c:\dosomething.bat password`. ■ Create custom property **`MyPassword =`** ***`password`*** where *password* is the value of your actual password. Enable encryption by selecting the available check box. ■ Set custom property `VirtualMachine.ScriptPath.Decrypt` as **`VirtualMachine.ScriptPath.Decrypt = true`**. ■ Set custom property `VirtualMachine.Software0.ScriptPath` as **`VirtualMachine.Software0.ScriptPath = c:\dosomething.bat [MyPassword]`**. If you set `VirtualMachine.ScriptPath.Decrypt` to false, or do not create the `VirtualMachine.ScriptPath.Decrypt` custom property, then the string inside the square brackets ([and]) is not decrypted.
`VirtualMachine.SoftwareN.ISOName`	Specifies the path and filename of the ISO file relative to the datastore root. The format is *`/folder_name/subfolder_name/file_name`*`.iso`. If a value is not specified, the ISO is not mounted.
`VirtualMachine.SoftwareN.ISOLocation`	Specifies the storage path that contains the ISO image file to be used by the application or script. Format the path as it appears on the host reservation, for example `netapp-1:it_nfs_1`. If a value is not specified, the ISO is not mounted.

Custom Properties for BMC BladeLogic Configuration Manager Integration

vRealize Automation includes custom properties that you can use to provide additional controls for BMC BladeLogic Configuration Manager integration.

Table 8-25. Custom Properties Required for BMC BladeLogic Configuration Manager Integrations

Custom Property	Description
`VirtualMachine.EPI.Type`	Specifies the type of external provisioning infrastructure.
`VirtualMachine.Admin.Owner`	Specifies the user name of the machine owner.
`BMC.Software.Install`	Set to True to enable BMC BladeLogic Configuration Manager integration.
`EPI.Server.Name`	Specifies the name of the external provisioning infrastructure server, for example, the name of the server hosting BMC BladeLogic. If at least one general BMC EPI agent was installed without specifying a BMC BladeLogic Configuration Manager host, this value directs the request to the desired server. If only dedicated BMC EPI agents for specific BMC BladeLogic Configuration Manager hosts were installed, this value must exactly match the server name configured for one of these agents.
`BMC.Service.Profile`	Specifies the name of the default authentication profile on the BMC BladeLogic server.
`BMC.Software.BatchLocation`	Specifies the location in BMC BladeLogic configuration where software jobs are deployed. This value must match the appropriate value of `Vrm.Software.IdNNNN`. For example, a valid value could be `/Application Deployment`.
`VMware.VirtualCenter.OperatingSystem`	Specifies the vCenter Server guest operating system version (`VirtualMachineGuestOsIdentifier`) with which vCenter Server creates the machine. This operating system version must match the operating system version to be installed on the provisioned machine. Administrators can create property groups using one of several property sets, for example, `VMware[OS_Version]Properties`, that are predefined to include the correct `VMware.VirtualCenter.OperatingSystem` values. This property is for virtual provisioning. For related information, see the enumeration type `VirtualMachineGuestOsIdentifier` in vSphere API/SDK Documentation. For a list of currently accepted values, see the vCenter Server documentation.

Custom Properties To Make BMC BladeLogic Configuration Manager Software Jobs Available

Configure BMC BladeLogic Configuration Manager jobs for vRealize Automation integrations. Make all software jobs available to machine requesters to select from, or specify a software job to apply to all machines provisioned from the blueprint.

Table 8-26. Custom Properties to Make Software Jobs Available

Custom Property	Description
`LoadSoftware`	Set to True to enable software install options.

`Vrm.Software.Id`*NNNN*	Specifies a software job or policy to be applied to all machines provisioned from the blueprint. Set the value to `job_type=job_path`, where `job_type` is the numeral that represents the BMC BladeLogic job type and `job_path` is the location of the job in BMC BladeLogic, for example `4=/Utility/putty`. *NNNN* is a number from 1000 to 1999. The first property must start with 1000 and increment in numerical order for each additional property.
	`1 – AuditJob`
	`2 – BatchJob`
	`3 – ComplianceJob`
	`4 – DeployJob`
	`5 – FileDeployJob`
	`6 – NSHScriptJob`
	`7 – PatchAnalysisJob`
	`8 – SnapshotJob`

Optional Custom Properties for BMC BladeLogic Configuration Manager Integrations

You can also use optional custom properties that are commonly used with BMC BladeLogic Configuration Manager blueprints.

Table 8-27. Optional Custom Properties for BMC BladeLogic Configuration Manager Integrations

Property	Definition
`BMC.AddServer.Delay`	Specifies the number of seconds to wait before adding the machine to BMC BladeLogic Configuration Manager. The default is 30.
`BMC.AddServer.Retry`	Specifies the number of seconds to wait before retrying if the first attempt to add the machine to BMC BladeLogic Configuration Manager is unsuccessful. The default is 100.

Custom Properties for HP Server Automation Integration

vRealize Automation includes custom properties that you can use to provide additional controls for HP Server Automation integration. Some custom properties are required for HP Server Automation integration. Other custom properties are optional.

Required Custom Properties for HP Server Automation Integration

Certain custom properties are required for a blueprint to work with HP Server Automation.

Table 8-28. Required Custom Properties for HP Server Automation Integration

Property	Definition
`VMware.VirtualCenter.OperatingSystem`	Specifies the vCenter Server guest operating system version (`VirtualMachineGuestOsIdentifier`) with which vCenter Server creates the machine. This operating system version must match the operating system version to be installed on the provisioned machine. Administrators can create property groups using one of several property sets, for example, `VMware[OS_Version]Properties`, that are predefined to include the correct `VMware.VirtualCenter.OperatingSystem` values. This property is for virtual provisioning.
`VirtualMachine.EPI.Type`	Specifies the type of external provisioning infrastructure.
`EPI.Server.Name`	Specifies the name of the external provisioning infrastructure server, for example, the name of the server hosting BMC BladeLogic. If at least one general BMC EPI agent was installed without specifying a BMC BladeLogic Configuration Manager host, this value directs the request to the desired server.
`Opsware.Software.Install`	Set to True to allow HP Server Automation to install software.
`Opsware.Server.Name`	Specifies the fully qualified name of the HP Server Automation server.
`Opsware.Server.Username`	Specifies the user name provided when a password file in the agent directory was created, for example opswareadmin. This user name requires administrative access to the HP Server Automation instance.
`Opsware.BootImage.Name`	Specifies the boot image value as defined in HP Server Automation for the 32-bit WinPE image, for example winpe32. The property is not required when provisioning by cloning.
`Opsware.Customer.Name`	Specifies a customer name value as defined in HP Server Automation, for example MyCompanyName.
`Opsware.Facility.Name`	Specifies a facility name value as defined in HP Server Automation, for example Cambridge.
`Opsware.Machine.Password`	Specifies the default local administrator password for an operating system sequence WIM image such as `Opsware.OSSequence.Name` as defined in HP Server Automation, for example P@ssword1.
`Opsware.OSSequence.Name`	Specifies the operating system sequence name value as defined in HP Server Automation, for example Windows 2008 WIM.
`Opsware.Realm.Name`	Specifies the realm name value as defined in HP Server Automation, for example Production.

Table 8-28. Required Custom Properties for HP Server Automation Integration (continued)

Property	Definition
`Opsware.Register.Timeout`	Specifies the time, in seconds, to wait for creation of a provisioning job to complete.
`VirtualMachine.CDROM.Attach`	Set to False to provision the machine without a CD-ROM device. The default is True.
`Linux.ExternalScript.Name`	Specifies the name of an optional customization script, for example `config.sh`, that the Linux guest agent runs after the operating system is installed. This property is available for Linux machines cloned from templates on which the Linux agent is installed.
`Linux.ExternalScript.LocationType`	Specifies the location type of the customization script named in the `Linux.ExternalScript.Name` property. This can be either local or nfs.
`Linux.ExternalScript.Path`	Specifies the local path to the Linux customization script or the export path to the Linux customization on the NFS server. The value must begin with a forward slash and not include the file name, for example `/scripts/linux/config.sh`.

Optional Custom Properties for HP Server Automation Integration

Certain custom properties are optional for a blueprint to work with HP Server Automation.

Table 8-29. Optional Custom Properties for HP Server Automation Integration

Property	Definition
`Opsware.ProvFail.Notify`	(Optional) Specifies the notification email address for HP Server Automation to use in the event of provisioning failure, for example provisionfail@lab.local.
`Opsware.ProvFail.Notify`	(Optional) Specifies the HP Server Automation user to whom ownership is assigned if provisioning fails.
`Opsware.ProvSuccess.Notify`	(Optional) Specifies the notification email address for HP Server Automation to use if provisioning is successful.
`Opsware.ProvSuccess.Owner`	(Optional) Specifies the HP Server Automation user to whom ownership is assigned if provisioning is successful.

Custom Properties That Make HP Server Automation Software Jobs Available

Depending on how your fabric administrator configures HP Server Automation jobs for vRealize Automation integration, you might have a choice between making all software jobs available to machine requesters to select, or you can specify jobs to apply to all machines provisioned from your blueprint.

Table 8-30. Custom Properties to Make Software Jobs Available

Property	Definition
`LoadSoftware`	Set to True to enable software install options.
`Vrm.Software.Id`	(Optional) Specifies an HP Server Automation policy to be applied to all machines provisioned from the blueprint. *NNNN* is a number from 1000 to 1999. The first property must start with 1000 and increment in numerical order for each additional property.

Custom Properties Grouped by Name

You can use custom properties to provide additional vRealize Automation controls.

Custom properties have been grouped here by name. To explore custom properties grouped by function, see Custom Properties Grouped by Function.

Custom Properties Underscore (_)

A list of vRealize Automation custom properties that begin with an underscore (_).

Table 8-31. Custom Properties Underscore (_) Table

Property	Description
`_debug_deployment`	Except for scale operations which allow partially successful deployments, the default behavior is to destroy the entire deployment if any of the individual resources fail to provision. You can override the default behavior by setting the `_debug_deployment` custom property value to true. If provisioning fails, the debugging custom property stops the resources from being rolled back so you can identify which of the components failed to provision successfully. In other words, by setting `_debug_deployment` to true, you can more easily debug customization and first-boot (for example, agent) issues because the setting ensures that machines are not destroyed after a provisioning failure. Otherwise, the setting doesn't directly change anything about the provisioning process, or affect guest agent or customization (for example, settings our outcomes relative to a vCenter customization spec). Note: A failed catalog item is normally inaccessible because it is immediately rolled back on failure. But when `_debug_deployment` is set to true, vRealize Automation treats the otherwise failed deployment as partially successful, which enables its accessibility. To apply the custom property to a blueprint, add `_debug_deployment` to the **Blueprint Properties** page using the **Properties** tab when you create or edit a blueprint. The `_debug_deployment` property is consumed at the software provisioning level, not the guest agent or machine provisioning level. You can also configure vRealize Automation to not delete virtual machines after deployment failure by using settings in the `VRMAgent.exe.config` file.
`_deploymentName`	When added to a blueprint, this property allows you to specify a custom name for the deployment by setting the value of `_deploymentName` to your custom string. If more than one instance of this deployment is provisioned in a single request, your custom name becomes a prefix. If you want users to specify their own deployment names, set this custom property to allow override. The following two caveats are required for usage: ■ You must add this property at the blueprint level, not at the component level. For example, when creating or editing a blueprint, click the **Properties** tab and then select **Custom Properties > New** to add the `_deploymentName` property to the blueprint. Do not add the property to a machine or other component in the blueprint. ■ You must add this property as a separate property and not as a member of a property group.

Custom Properties A

A list of vRealize Automation custom properties that begin with the letter A.

Table 8-32. Custom Properties A Table

Property	Description
`AD.Lookup.Department`	Specifies the cost center value that is included in a notification email sent to approvers. This property value must be specified in the blueprint.
`agent.download.url`	When using port forwarding, specifies the private IP address of your Amazon AWS tunnel machine and port for your software agent file, for example **`https://`*`Private_IP:1443`*`/software-service/resources/nobel-agent.jar`**. Specifies the URL for the VPN agent on your deployment. The URL format is `https://` *`Private_IP`*`:1443/software-service//resources/noble-agent.jar` You can add this property, in combination with `software.agent.service.url` and `software.ebs.url`, to a reservation or the compute resource endpoint. You can also use this property to specify a private address and port when using PAT or NAT translation and port forwarding.
`amazon.AmazonEC2Config.ServiceURL`	Specifies the Amazon configuration service URL for Amazon GovCloud, for example `amazon.AmazonEC2Config.ServiceURL`=https://ec2.us-gov-west-1.amazonaws.com.
`amazon.ElasticLoadBalancingConfig.ServiceURL`	Specifies the Amazon load balancer configuration service URL for Amazon GovCloud, for example `amazon.ElasticLoadBalancingConfig.ServiceURL`=https://elasticloadbalancing.us-gov-west-1.amazonaws.com.
`Amazon.ElasticLoadBalancer.Names`	Assigns machines that are provisioned by a blueprint to the elastic load balancers that match the specified values. This property is valid for vSphere, Amazon, and Hyper-V configurations.
`Amazon.Extensions.UserData`	Specifies the name of an Amazon user data script to be run during the first boot cycle when an instance is launched. The property supports string substitution from other custom properties to allow for dynamic requests. You can add the property either to the overall vRealize Automation blueprint or to an AWS machine component in the blueprint. For information about Amazon user data scripts, see the Running Commands on Your Linux Instance at Launch topic in *Amazon Elastic Compute Cloud* product documentation. You can pass a series of custom properties to the `Amazon.Extensions.UserData` property by including them in a file whose name begins with `Amazon.CustomProperty.Shell`.

Table 8-32. Custom Properties A Table (continued)

Property	Description
`amazon.IAMInstanceProfile.ARN`	Specifies the AWS Identity and Access Management (IAM) instance profile Amazon Resource Names (ARNs) when requesting an AWS instance. When you add this property, for example `amazon.IAMInstanceProfile.ARN` = *IAM Instance Profile ARN(s) value*, to a blueprint and then request provisioning from the catalog, the provisioned Amazon virtual machine or instance contains the specified IAM role. The DEM reads and includes the property specification, for example amazon.IAMInstanceProfile.ARN = IAM Instance Profile ARN(s) value, in the Amazon RunInstanceRequest workflow.
`Amazon.Instance.Id`	Specifies the Amazon instance ID of a machine provisioned on an Amazon EC2 endpoint. This property is valid for vSphere and Amazon configurations.
`Amazon.Instance.GroupName`	Specifies the name of the existing AWS placement group for the associated Amazon endpoint. The placement group must exist in the target availability zone prior to vRealize Automation data collection of the endpoint. Add the `Amazon.Instance.GroupName` custom property to a blueprint to specify which AWS placement group is used during machine provisioning.
`Amazon.elasticIpAddress.ipAddress`	Specifies the Amazon IP address where *ipAddress* is the specific IP address to assign to the instance.
`Amazon.Placement.Tenancy`	Set to = dedicated to specify that the AWS connection be specific to a dedicated tenant. This property is valid for use with VPC subnets.
`Amazon.Storage.Encrypt`	If set to true, specifies whether the Amazon EBS storage disks attached to the EC2 machine should be encrypted or not encrypted. Default is false. The property only applies to new EBS volume encryptions. Volumes that are part of an Amazon Machine Image (AMI) definition maintain their AMI settings regardless of this property.
`Amazon.Storage.iops`	Specifies the input/output operations per second (IOPS) for the associated storage device. Currently, this property is only supported when the `Amazon.Storage.Type` property value is io1. For more information, see Amazon EBS volume types documentation. Add the `Amazon.Storage.iops` custom property to a blueprint to specify the IOPS. The io1 storage type is the only AWS storage type in which you can set IOPS.

Table 8-32. Custom Properties A Table (continued)

Property	Description
`Amazon.Storage.Type`	Specifies the Amazon EBS volume type to use for disk storage relative to the associated Amazon endpoint. All disks are provisioned with the specified type. You cannot specify a different volume type for each disk. Set the property value to one of the API Names values provided in Amazon EBS volume types documentation, for example io1 or gp2. Add the `Amazon.Storage.Type` custom property to a blueprint to specify the EBS volume type to use during machine provisioning.
`Azure.Windows.ScriptPath`	Specifies the path to the downloaded script that configures tunneling for Windows-based systems. Update the path as appropriate for your deployment.
`Azure.Linux.ScriptPath`	Specifies the path to the downloaded script that configures tunneling for Linux-based systems. Update the path as appropriate for your deployment.

Custom Properties B

A list of vRealize Automation custom properties that begin with the letter B.

Table 8-33. Custom Properties B Table

Property	Definition
`BMC.AddServer.Delay`	Specifies the number of seconds to wait before adding the machine to BMC BladeLogic Configuration Manager. The default is 30.
`BMC.AddServer.Retry`	Specifies the number of seconds to wait before retrying if the first attempt to add the machine to BMC BladeLogic Configuration Manager is unsuccessful. The default is 100.
`BMC.Service.Profile`	Specifies the name of the default authentication profile on the BMC BladeLogic server.
`BMC.Software.BatchLocation`	Specifies the location in BMC BladeLogic configuration where software jobs are deployed. This value must match the appropriate value of `Vrm.Software.IdNNNN`. For example, a valid value could be `/Application Deployment`.
`BMC.Software.Install`	Set to True to enable BMC BladeLogic Configuration Manager integration.

Custom Properties C

A list of vRealize Automation custom properties that begin with the letter C.

Table 8-34. Custom Properties C Table

Property	Definition
`Cisco.Organization.Dn`	Specifies the distinguished name of the Cisco UCS Manager organization in which Cisco UCS machines provisioned by the business group are placed, for example org-root/org-Engineering. If the specified organization does not exist in the Cisco UCS Manager instance that is managing the machine, provisioning fails. This property is available for business groups only.
`CloneFrom`	Specifies the name of an existing machine or virtualization platform object to clone from, for example a template in vCenter Server such as Win2k8tmpl.
`CloneSpec`	Specifies the name of a customization specification on a cloned machine, for example a predefined `SysPrep` object in vCenter Server such as Win2k Customization Spec. The default value is specified on the blueprint.
`Command.DiskPart.Options`	When you use WIM-based virtual provisioning on ESX server hosts, set to Align=64 to use the recommended alignment parameters when you format and partition the machine's disk. This property is not available for physical provisioning.
`Command.FormatDisk.Options`	When you use WIM-based virtual provisioning on ESX server hosts, set to /A:32K to use the recommended alignment parameters when you format and partition the machine's disk. This property is not available for physical provisioning.
`containers.ipam.driver`	For use with containers only. Specifies the IPAM driver to be used when adding a Containers network component to a blueprint. The supported values depend on the drivers installed in the container host environment in which they are used. For example, a supported value might be `infoblox` or `calico` depending on the IPAM plug-ins that are installed on the container host. This property name and value are case-sensitive. The property value is not validated when you add it. If the specified driver does not exist on the container host at provisioning time, an error message is returned and provisioning fails.
`containers.network.driver`	For use with containers only. Specifies the network driver to be used when adding a Containers network component to a blueprint. The supported values depend on the drivers installed in the container host environment in which they are used. By default, Docker-supplied network drivers include bridge, overlay, and macvlan, while Virtual Container Host (VCH)-supplied network drivers include the bridge driver. Third-party network drivers such as `weave` and `calico` might also be available, depending on what network plug-ins are installed on the container host. This property name and value are case-sensitive. The property value is not validated when you add it. If the specified driver does not exist on the container host at provisioning time, an error message is returned and provisioning fails.
`Container`	For use with containers only. The default value is `App.Docker` and is required. Do not modify this property.

Table 8-34. Custom Properties C Table (continued)

Property	Definition
`Container.Auth.User`	For use with containers only. Specifies the user name for connecting to the Containers host.
`Container.Auth.Password`	For use with containers only. Specifies either the password for the user name or the public or private key password to be used. Encrypted property value is supported.
`Container.Auth.PublicKey`	For use with containers only. Specifies the public key for connecting to the Containers host.
`Container.Auth.PrivateKey`	For use with containers only. Specifies private key for connecting to the Containers host. Encrypted property value is supported.
`Container.Connection.Protocol`	For use with containers only. Specifies the communication protocol. The default value is `API` and is required. Do not modify this property.
`Container.Connection.Scheme`	For use with containers only. Specifies the communication scheme. The default is `https`.
`Container.Connection.Port`	For use with containers only. Specifies the Containers connection port. The default is 2376.
`Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.MachineActivated`	For use with containers only. Specifies the event broker property to expose all Containers properties and is used for registering a provisioned host. The default value is `Container*` and is required. Do not modify this property.
`Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.Disposing`	For use with containers only. Specifies the event broker property to expose all Containersproperties above and is used for unregistering a provisioned host. The default value is `Container*` and is required. Do not modify this property.

Custom Properties E

A list of vRealize Automation custom properties that begin with the letter E.

Table 8-35. Custom Properties E Table

Property	Definition
`EPI.Server.Collection`	Specifies the name of the Citrix provisioning collection to which the machine is to be registered.
`EPI.Server.Name`	Specifies the name of the external provisioning infrastructure server, for example, the name of the server hosting BMC BladeLogic. If at least one general BMC EPI agent was installed without specifying a BMC BladeLogic Configuration Manager host, this value directs the request to the desired server. If only dedicated BMC EPI agents for specific BMC BladeLogic Configuration Manager hosts were installed, this value must exactly match the server name configured for one of these agents. Specifies the name of the server hosting HP Server Automation. If at least one general Opsware EPI agent was installed without specifying a server automation server, this value directs the request to the desired server. If only dedicated EPI agents for specific HP server automation servers were installed, this value must exactly match the server name configured for one of these agents. If at least one general EPI agent of the appropriate type (`VirtualMachine.EPI.Type`) was installed without specifying a server, this value directs the request to the desired server. If only dedicated EPI agents for specific servers of the appropriate type were installed, this value must exactly match the server name configured for one of these agents.
`EPI.Server.Port`	Specifies the port on which to contact the provisioning server. If you are using a Citrix provisioning server, omit to specify the default port value of 54321.
`EPI.Server.Site`	Specifies the name of the Citrix provisioning site that contains the collection and store identified by the `EPI.Server.Collection` and `EPI.Server.Store` properties, for example site1.
`EPI.Server.Store`	Specifies the name of the Citrix provisioning store that contains the vDisk identified by the `EPI.Server.VDiskName` property, for example store1.
`EPI.Server.VDiskName`	Specifies the name of the Citrix provisioning vDisk from which to provision, for example disk1.
`ext.policy.activedirectory.customizationWorkflowTag`	The tag that you added to a custom vRealize Orchestrator workflow. The Active Directory policy searches for custom workflows with the specified tag and, if found, uses the tagged workflow when an Active Directory record is created.
`ext.policy.activedirectory.customizationDeleteWorkflowTag`	The tag that you added to a custom vRealize Orchestrator workflow. The Active Directory policy searches for custom workflows with the specified tag and, if found, uses the tagged workflow when an Active Directory record is deleted.
`ext.policy.activedirectory.domain`	The domain that you want to user rather than the domain in the current Active Directory policy. Overrides `ext.policy.activedirectory.system.domain` value that is specified in the Active Directory policy.

Table 8-35. Custom Properties E Table (continued)

Property	Definition
`ext.policy.activedirectory.endpoint.id`	The policy identifier to use to specify a policy or override policy. The ID that you provide must be for an existing Active Directory policy. Overrides `ext.policy.activedirectory.system.endpoint.id`value that is specified in the Active Directory policy.
`ext.policy.activedirectory.id`	The user-readable ID for the Active Directory policy. Overrides the `ext.policy.activedirectory.system.id` value that is specified in the Active Directory policy.
`ext.policy.activedirectory.ignore`	Indicates that the machine is not added to Active Directory by a policy. It skips the Active Directory policy.
`ext.policy.activedirectory.machineName`	The name of the machine in Active Directory that you want to use rather than the name in the current Active Directory policy. Overrides `ext.policy.activedirectory.system.machineName` value that is specified in the Active Directory policy.
`ext.policy.activedirectory.orgunit`	The organizational unit that you want to use rather than the organizational unit in the current Active Directory policy. Overrides the `ext.policy.activedirectory.system.orgunit` value that is specified in the Active Directory policy.
`ext.policy.activedirectory.system.domain`	System property for the domain of the machine in Active Directory. If you modify this property, which is used by the defined policies, you can deactivate the policy. Use `ext.policy.activedirectory.domain` to override the policy value.
`ext.policy.activedirectory.system.endpoint.id`	System property for the name of the Active Directory vRealize Orchestrator endpoint. If you modify this property, which is used by the defined policies, you can deactivate the policy. Use `ext.policy.activedirectory.endpoint.id` to override the policy value.
`ext.policy.activedirectory.system.id`	System property for the user-readable ID for the Active Directory policy. If you modify this property, which is used by the defined policies, you can deactivate the policy. Use `ext.policy.activedirectory.id` to override the policy value.
`ext.policy.activedirectory.system.machineName`	System property for the name of the machine in Active Directory. If you modify this property, which is used by the defined policies, you can deactivate the policy. Use `ext.policy.activedirectory.machineName` to override the policy value.
`ext.policy.activedirectory.system.orgunit`	System property for the distinguished name of the Active Directory organizational unit. If you modify this property, which is used by the defined policies, you can deactivate the policy. Use `ext.policy.activedirectory.orgunit` to override the policy value.

Custom Properties H

A list of vRealize Automation custom properties that begin with the letter H.

Table 8-36. Custom Properties H Table

Property	Definition
`Hostname`	Specifies the host machine name, overriding the generated machine name contained in the `VirtualMachine.Admin.Name` property. If `Hostname` is not used, the `VirtualMachine.Admin.Name` value is used as the machine name. The maximum number of allowed characters for the `Hostname` value is 15.
`Hyperv.Network.Type`	Specifies the network adapter type of the virtual machine. This property is valid for use with Hyper-V (SCVMM) only. When the value is set to synthetic, specifies that the blueprint be allowed to provision a Generation-2 machine on a Hyper-V (SCVMM) 2012 R2 resource. Generation-2 provisioning also requires that the blueprint includes the `Scvmm.Generation2 = true` property setting. The legacy value is not compatible with WinXP or Server 2003 x64 guest operating systems. The default value is synthetic.

Custom Properties I

A list of vRealize Automation custom properties that begin with the letter I.

Table 8-37. Custom Properties I Table

Property	Definition
`Image.ISO.Location`	Values for this property are case sensitive. Specifies the location of the ISO image from which to boot. The format of this value depends on your platform. For details, see documentation provided for your platform. This property is required for WIM-based provisioning, Linux Kickstart and autoYaST provisioning, and SCCM-based provisioning. For virtual provisioning with vCenter Server, this specifies the name of a datastore in the instance that will be accessible to the provisioning compute resource. For virtual provisioning with XenServer, this specifies the name of a storage repository. For physical provisioning, this specifies the HTTP URL of the web-accessible location of the image.
`Image.ISO.Name`	Values for this property are case sensitive. Specifies the name of the ISO image from which to boot, for example /ISO/Microsoft/WinPE.iso. The format of this value depends on your platform. For details, see documentation provided for your platform. This property is required for WIM-based provisioning, Linux Kickstart and autoYaST provisioning, and SCCM-based provisioning. For virtual provisioning with vCenter Server, this svalue specifies the path to the image in the datastore specified by `Image.ISO.Location`. The value must use forward slashes and begin with a forward slash. For virtual provisioning with XenServer, this value specifies the name of the image in the storage repository specified by `Image.ISO.Location`. In virtual provisioning with Hyper-V, this value specifies the full local path to the image. For physical provisioning, this value specifies the file name of the image.
`Image.ISO.UserName`	Specifies the user name to access the CIFS share in the format *username@domain*. For Dell iDRAC integrations where the image is located on a CIFS share that requires authentication to access.

Table 8-37. Custom Properties I Table (continued)

Property	Definition
`Image.ISO.Password`	Specifies the password associated with the `Image.ISO.UserName` property. For Dell iDRAC integrations where the image is located on a CIFS share that requires authentication to access.
`Image.WIM.Path`	Specifies the UNC path to the WIM file from which an image is extracted during WIM-based provisioning. The path format is \\*server**share$* format, for example `\\lab-ad\dfs$`.
`Image.WIM.Name`	Specifies the name of the WIM file as located by the `Image.WIM.Path` property.
`Image.WIM.Index`	Specifies the index used to extract the correct image from the WIM file.
`Image.Network.User`	Specifies the user name with which to map the WIM image path (`Image.WIM.Path`) to a network drive on the provisioned machine. This is typically a domain account with access to the network share.
`Image.Network.Password`	Specifies the password associated with the `Image.Network.User` property.
`Image.Network.Letter`	Specifies the drive letter to which the WIM image path is mapped on the provisioned machine. The default value is K.
`Infrastructure.Admin.MachineObjectOU`	Specifies the organizational unit (OU) of the machine. When machines are placed in the required OU by the business group OU setting, this property is not required.
`Infrastructure.Admin.ADUser`	Specifies the domain administrator user ID. This identifier is used to query Active Directory users and groups when an anonymous bind cannot be used.
`Infrastructure.Admin.ADPassword`	Specifies the password associated with the `Infrastructure.Admin.ADUser` domain administrator user ID.
`Infrastructure.Admin.DefaultDomain`	Specifies the default domain on the machine.
`Infrastructure.ResourcePool.Name`	Specifies the resource pool to which the machine belongs, if any. The default is the value specified in the reservation from which the machine was provisioned.

Custom Properties L

A list of vRealize Automation custom properties that begin with the letter L.

Table 8-38. Custom Properties L Table

Property	Description
`Linux.ExternalScript.LocationType`	Specifies the location type of the customization script named in the `Linux.ExternalScript.Name` property. This can be either local or nfs. You must also specify the script location using the `Linux.ExternalScript.Path` property. If the location type is nfs, also use the `Linux.ExternalScript.Server` property.
`Linux.ExternalScript.Name`	Specifies the name of an optional customization script, for example `config.sh`, that the Linux guest agent runs after the operating system is installed. This property is available for Linux machines cloned from templates on which the Linux agent is installed. If you specify an external script, you must also define its location by using the `Linux.ExternalScript.LocationType` and `Linux.ExternalScript.Path` properties.
`Linux.ExternalScript.Path`	Specifies the local path to the Linux customization script or the export path to the Linux customization on the NFS server. The value must begin with a forward slash and not include the file name, for example `/scripts/linux/config.sh`.
`Linux.ExternalScript.Server`	Specifies the name of the NFS server, for example lab-ad.lab.local, on which the Linux external customization script named in `Linux.ExternalScript.Name` is located.
`LoadSoftware`	Set to True to enable software install options.

Custom Properties M

A list of vRealize Automation custom properties that begin with the letter M.

Table 8-39. Custom Properties M Table

Property	Description
`MaximumProvisionedMachines`	Specifies the maximum number of linked clones for one machine snapshot. The default is unlimited.
`Machine.SSH`	Set to True to enable the **Connect Using SSH** option on the vRealize Automation Items page for Linux machines provisioned from this blueprint. If set to True and the **Connect using RDP or SSH** machine operation is enabled in the blueprint, all Linux machines that are provisioned from the blueprint display the **Connect Using SSH** option to entitled users. The **Connect Using SSH** option requires that your browser has a plug-in that supports SSH, for example the FireSSH SSH terminal client for Mozilla Firefox and Google Chrome. When the plug-in is present, selecting **Connect Using SSH** displays an SSH console and prompts for your administrator credentials.

Custom Properties N

A list of vRealize Automation custom properties that begin with the letter N.

Table 8-40. Custom Properties N Table

Property	Description
`NSX.Edge.ApplianceSize`	Specifies the allowed NSX edge appliance size types for the provisioned machine or deployment. The options are: ■ `compact` For small deployments, POCs, and single service use. ■ CPU = 1 ■ RAM = 512 MB ■ Disk = 512 MB ■ `large` For small to medium or multi-tenant deployments. ■ CPU = 2 ■ RAM = 1 GB ■ Disk = 512 MB ■ `quadlarge` For high throughput equal-cost multi-path routing (ECMP) or high performance firewall deployments. ■ CPU = 4 ■ RAM = 1 GB ■ Disk = 512 MB ■ `xlarge` For L7 load balancing and dedicated core deployments. ■ CPU = 6 ■ RAM = 8 GB ■ Disk = 4.5GB (4GB Swap) For related information, see System Requirements for NSX.
`NSX.Edge.HighAvailability`	When set to true (`NSX.Edge.HighAvailability`=true), enables high availability (HA) mode on the NSX edge machine that is deployed from the blueprint. When used with `NSX.Edge.HighAvailability.PortGroup`=*port_group_name*, this property allows you to configure an NSX edge during blueprint authoring. You can add this property to an NSX load balancer component in the vRealize Automation blueprint or to the vRealize Automation blueprint itself. Must be used in conjunction with `NSX.Edge.HighAvailability.PortGroup`= *port_group_name*.

Table 8-40. Custom Properties N Table (continued)

Property	Description
`NSX.Edge.HighAvailability.PortGroup`	Creates an internal interface or internal vNIC attached to the specified port group name, for example `NSX.Edge.HighAvailability.PortGroup=VM Network` where `VM Network` is an HA (high availability) distributed (vLAN-backed) or NSX logical switch port group. NSX HA mode requires at least one internal network interface, or vNIC. When used with `NSX.Edge.HighAvailability`=true, this property allows you to configure high availability (HA) an NSX edge during blueprint authoring. When using one arm load balancer with HA enabled, you must specify a separate port group for the HA. **Note** The specified port group network cannot be a member of the reservation pool, as the property's use of the port group conflicts with the normal deployment's use of the port group, resulting in the following error: `Portgroup must be unique within an Edge...` Must be used in conjunction with `NSX.Edge.HighAvailability`=true.

Table 8-40. Custom Properties N Table (continued)

Property	Description
`NSX.Validation.Disable.Single.Edge.Uplink`	When set to true, the NSX validation that checks for the following conditions is deactivated: ■ All on-demand NAT networks on the blueprint source the same external network. ■ All on-demand routed networks on the blueprint that use the load balancer VIP source the same external network. ■ All on-demand load balancer components on the blueprint have VIPs on the same external network or on-demand networks backed by the same external network. Disabling this validation check can result in a deployment that succeeds but in which some network components might be inaccessible. If not present or if set to false, the validation check is enabled (default). A single NSX edge can only support one external network as its uplink network. Multiple IPs from the same external network are supported. While a blueprint can contain any number of external or on-demand network components, NSX only supports one external network as the uplink network. This property can only be specified at the blueprint level. It cannot be specified on a component in the blueprint canvas.
`NSX.Validation.Disable.Blueprint.NSXT`	When set to true, all NSX-T validation is deactivated for the blueprint **Finish** action. If not present or if set to false, the NSX-T validation check is enabled (default). For example, if you have overlapping subnets in the blueprint, an error message appears when you click **Finish** in the blueprint and the overlap prevents you from finishing the blueprint, although you can save it. If you want to finish the blueprint, you can add `NSX.Validation.Disable.Blueprint.NSXT` by using the **Blueprint Properties** page and then finish the blueprint. The property only deactivates NSX-T validations for the blueprint **Finish** action.

Custom Properties O

A list of vRealize Automation custom properties that begin with the letter O.

Table 8-41. Custom Properties O Table

Property	Description
`Opsware.BootImage.Name`	Specifies the boot image value as defined in HP Server Automation for the 32-bit WinPE image, for example winpe32. The property is not required when provisioning by cloning.
`Opsware.Customer.Name`	Specifies a customer name value as defined in HP Server Automation, for example MyCompanyName.
`Opsware.Facility.Name`	Specifies a facility name value as defined in HP Server Automation, for example Cambridge.
`Opsware.Machine.Password`	Specifies the default local administrator password for an operating system sequence WIM image such as `Opsware.OSSequence.Name` as defined in HP Server Automation, for example P@ssword1.
`Opsware.OSSequence.Name`	Specifies the operating system sequence name value as defined in HP Server Automation, for example Windows 2008 WIM.
`Opsware.ProvFail.Notify`	(Optional) Specifies the notification email address for HP Server Automation to use in the event of provisioning failure, for example provisionfail@lab.local.
`Opsware.ProvFail.Owner`	(Optional) Specifies the HP Server Automation user to whom ownership is assigned if provisioning fails.
`Opsware.ProvSuccess.Notify`	(Optional) Specifies the notification email address for HP Server Automation to use if provisioning is successful.
`Opsware.ProvSuccess.Owner`	(Optional) Specifies the HP Server Automation user to whom ownership is assigned if provisioning is successful.
`Opsware.Realm.Name`	Specifies the realm name value as defined in HP Server Automation, for example Production.
`Opsware.Register.Timeout`	Specifies the time, in seconds, to wait for creation of a provisioning job to complete.
`Opsware.Server.Name`	Specifies the fully qualified name of the HP Server Automation server.
`Opsware.Server.Username`	Specifies the user name provided when a password file in the agent directory was created, for example opswareadmin. This user name requires administrative access to the HP Server Automation instance.
`Opsware.Software.Install`	Set to True to allow HP Server Automation to install software.

Custom Properties P

A list of vRealize Automation custom properties that begin with the letter P.

Table 8-42. Custom Properties P Table

Property	Description
`Plugin.AdMachineCleanup.Delete`	Set to True to delete the accounts of destroyed machines, instead of disabling them.
`Plugin.AdMachineCleanup.Execute`	Set to True to enable the Active Directory cleanup plug-in. By default, each machine's account is deactivated when it is destroyed.
`Plugin.AdMachineCleanup.MoveToOu`	Moves the account of destroyed machines to a new Active Directory organizational unit. The value is the organization unit to which you are moving the account. This value must be in *ou=OU, dc=dc* format, for example ou=trash,cn=computers,dc=lab,dc=local.
`Plugin.AdMachineCleanup.UserName`	Specifies an Active Directory account user name with sufficient privileges to perform Active Directory actions such as delete, deactivate, rename, or move Active Directory accounts. The value must be in *domain\username* format, for example lab\administrator. This property is required if the vRealize Automation manager service does not have these rights in a domain, which can occur when you provision machines in more than one domain.
`Plugin.AdMachineCleanup.Password`	Specifies the password associated to the `Plugin.AdMachineCleanup.UserName` property.
`Plugin.AdMachineCleanup.Domain`	Specifies the Active Directory domain name that contains the machine account to be destroyed.
`Plugin.AdMachineCleanup.RenamePrefix`	Renames the accounts of destroyed machines by adding a prefix. The value is the prefix string to prepend, for example destroyed_.
`Pxe.Clean.ScriptName`	Specifies the name of an EPI PowerShell script installed in the vRealize Automation Model Manager, to run on the machine after it is provisioned. The value is the name assigned to the script when it is uploaded to the Model Manager, for example `clean.ps1`.
`Pxe.Setup.ScriptName`	Specifies a custom EPI PowerShell script to run on the machine before it is started by using the PXE network boot program. The value is the name assigned to the script when it is uploaded to the model manager, for example `setup.ps1`.

Custom Properties R

A list of vRealize Automation custom properties that begin with the letter R.

Table 8-43. Custom Properties R Table

Property	Description
`ReservationPolicyID`	Specifies the reservation policy ID, not the reservation policy name. For example, the name that is returned by the vRealize Orchestrator property `getApplicableReservationPolicies` is the reservation policy name, not the reservation policy ID.

Custom Properties S

A list of vRealize Automation custom properties that begin with the letter S.

Table 8-44. Custom Properties S Table

Property	Description
`SysPrep.`*Section.Key* - `SysPrep.GuiUnattended.AdminPassword` - `SysPrep.GuiUnattended.EncryptedAdminPassword` - `SysPrep.GuiUnattended.TimeZone`	Specifies information to be added to the SysPrep answer file on machines during the WinPE stage of provisioning. Information that already exists in the SysPrep answer file is overwritten by these custom properties. *Section* represents the name of the section of the SysPrep answer file, for example GuiUnattended or UserData. *Key* represents a key name in the section. For example, to set the time zone of a provisioned machine to West Pacific Standard Time, define the custom property `GuiUnattended.UserData.TimeZone` and set the value to 275. For a full list of sections, keys, and accepted values, see the System Preparation Utility for Windows documentation. The following *Section.Key* combinations can be specified for WIM-based provisioning: - GuiUnattended - AdminPassword - EncryptedAdminPassword - TimeZone - UserData - ProductKey - FullName - ComputerName - OrgName - Identification - DomainAdmin - DomainAdminPassword - JoinDomain - JoinWorkgroup
`Sysprep.Identification.DomainAdmin`	Specifies a user name with administrator-level access to the target domain in Active Directory. Do not include the user domain in the credentials that you send to vCloud Director or vCloud Air.
`Sysprep.Identification.DomainAdminPassword`	Specifies the password to associate with the `Sysprep.Identification.DomainAdmin` property.
`Sysprep.Identification.JoinDomain`	Specifies the name of the domain to join in Active Directory.
`Sysprep.Identification.JoinWorkgroup`	Specifies the name of the workgroup to join if not using a domain.
`SysPrep.UserData.ComputerName`	Specifies a machine name, for example lab-client005.
`SysPrep.UserData.FullName`	Specifies the full name of a user.

Table 8-44. Custom Properties S Table (continued)

Property	Description
`SysPrep.UserData.OrgName`	Specifies the organization name of the user.
`SysPrep.UserData.ProductKey`	Specifies the Windows product key.
`SCCM.Collection.Name`	Specifies the name of the SCCM collection that contains the operating system deployment task sequence.
`SCCM.CustomVariable.`*Name*	Specifies the value of a custom variable, where *Name* is the name of any custom variable to be made available to the SCCM task sequence after the provisioned machine is registered with the SCCM collection. The value is determined by your choice of custom variable. If your integration requires it, you can use `SCCM.RemoveCustomVariablePrefix` to remove the `SCCM.CustomVariable.` prefix from your custom variable.
`SCCM.Server.Name`	Specifies the fully qualified domain name of the SCCM server on which the collection resides, for example lab-sccm.lab.local.
`SCCM.Server.SiteCode`	Specifies the site code of the SCCM server.
`SCCM.Server.UserName`	Specifies a user name with administrator-level access to the SCCM server.
`SCCM.Server.Password`	Specifies the password associated with the `SCCM.Server.UserName` property.
`SCCM.RemoveCustomVariablePrefix`	Set to *true* to remove the prefix `SCCM.CustomVariable.` from SCCM custom variables you created by using the custom property `SCCM.CustomVariable.`*Name*.
`Scvmm.Generation2`	When set to true, specifies that the blueprint be allowed to provision a Generation-2 machine on a Hyper-V (SCVMM) 2012 R2 resource. Generation-2 provisioning also requires that the blueprint includes the `Hyperv.Network.Type = synthetic` property setting.
`Snapshot.Policy.AgeLimit`	Sets the age limit, in days, for snapshots that can be applied to machines. This property applies to vSphere provisioning. When a snapshot exceeds the age limit, the Apply option is no longer available. When the snapshot age limit is reached, the snapshot remains but you can no longer revert to it. You can delete the snapshot using the vSphere client.

Table 8-44. Custom Properties S Table (continued)

Property	Description
`Snapshot.Policy.Limit`	Sets the number of snapshots allowed per machine. The default setting is one snapshot per machine. This property applies to vSphere provisioning. When set to 0, the blueprint option to create a snapshot is hidden for all users except for support and manager roles. Snapshots are shown in a hierarchical structure. ■ Depth – Maximum is 31. ■ Width – There is no limit.
`software.agent.service.url`	When using port forwarding, specifies the private IP address of your Amazon AWS tunnel machine and port for the vRealize Automation software service API, for example **`https://`*`Private_IP:1443`*`/software-service/api`**. You can add this property, in combination with `software.ebs.url` and `agent.download.url`, to a reservation or the compute resource endpoint. You can also use this property to specify a private address and port when using PAT or NAT and port forwarding.
`software.agent.task.timeout.seconds`	Specifies the timeout period, in seconds, for software scripts that are executing on agents. By default, the timeout period for software scripts that are executing on agents is 6 hours.
`software.ebs.url`	When using port forwarding, specifies the private IP address of your Amazon AWS tunnel machine and port for the vRealize Automation event broker service, for example **`https://`*`Private_IP:1443`*`/event-broker-service/api`**. You can add this property, in combination with `software.agent.service.url` and `agent.download.url`, to a reservation or the compute resource endpoint. You can also use this property to specify a private address and port when using PAT or NAT and port forwarding.

Table 8-44. Custom Properties S Table (continued)

Property	Description
`software.http.proxyHost`	Specifies the host name, or address, of the proxy server. For software content properties to use the proxy server, you must use both `software.http.proxyHost` and `software.http.proxyPort`. **Note** You can use the software proxy settings to define a content property type value for a software component. Content properties are URLs that are downloaded by the agent. The agent uses the variable as a file path to the locally downloaded file. However, you can use the software proxy settings to download through the proxy host instead of from the URL.
`software.http.proxyPassword`	Specifies the password for the user name with which to authenticate to the proxy server. Use in combination with `software.http.proxyUser`. The `software.http.proxyPassword` setting is required if you use the `software.http.proxyUser` setting. **Note** You can use the software proxy settings to define a content property type value for a software component. Content properties are URLs that are downloaded by the agent. The agent uses the variable as a file path to the locally downloaded file. However, you can use the software proxy settings to download through the proxy host instead of from the URL.
`software.http.proxyPort`	Specifies the port number of the proxy server. For software content properties to use the proxy server, you must use both `software.http.proxyHost` and `software.http.proxyPort`. There is no default `software.http.proxyPort` value. **Note** You can use the software proxy settings to define a content property type value for a software component. Content properties are URLs that are downloaded by the agent. The agent uses the variable as a file path to the locally downloaded file. However, you can use the software proxy settings to download through the proxy host instead of from the URL.

Table 8-44. Custom Properties S Table (continued)

Property	Description
`software.http.proxyUser`	Specifies the user name with which to authenticate to the proxy server. Use in combination with `software.http.proxyPassword`. The `software.http.proxyUser` setting is optional. The `software.http.proxyPassword` setting is required if you use the `software.http.proxyUser` setting. **Note** You can use the software proxy settings to define a content property type value for a software component. Content properties are URLs that are downloaded by the agent. The agent uses the variable as a file path to the locally downloaded file. However, you can use the software proxy settings to download through the proxy host instead of from the URL.
`software.http.noProxyList`	Specifies a list of hosts, and optional ports, that cannot use the proxyHost. The original content property downloads directly from URLs that match the patterns in the list. The `software.http.noProxyList` setting is only applicable if the proxy server is configured. For example, for the following comma separated list: `"buildweb.eng.vmware.com,confluence.eng.vmware.com:443,*.eng.vmware.com:80"` The following statements apply: ■ Any URL whose HOST is "buildweb.eng.vmware.com" cannot use the proxy server. ■ Any URL whose HOST is "confluence.eng.vmware.com" and the whose PORT is 443 cannot use the proxy server. ■ Any URL whose HOST is anything under the "eng.vmware.com" namespace and whose PORT is 80 cannot use the proxy server. **Note** You can use the software proxy settings to define a content property type value for a software component. Content properties are URLs that are downloaded by the agent. The agent uses the variable as a file path to the locally downloaded file. However, you can use the software proxy settings to download through the proxy host instead of from the URL.

Custom Properties V

A list of vRealize Automation custom properties that begin with the letter V.

Although general support for vCloud Networking and Security has ended, the VCNS custom properties continue to be valid for NSX purposes. See the Knowledge Base article 2144733.

Table 8-45. Custom Properties V Table

Property	Description
`VbScript.PreProvisioning.Name`	Specifies the full path of a Visual Basic script to be run before a machine is provisioned. For example, `%System-Drive%\Program Files(x86)\VMware\vCAC Agents\EPI_Agent\Scripts \SendEmail.vbs`. The script file must reside on the system on which the Visual Basic script EPI agent is installed.
`VbScript.PostProvisioning.Name`	Specifies the full path of a Visual Basic script to be run after a machine is provisioned. For example, `%System-Drive%\Program Files(x86)\VMware\vCAC Agents\EPI_Agent\Scripts \SendEmail.vbs`. The script file must reside on the system on which the Visual Basic script EPI agent is installed.
`VbScript.UnProvisioning.Name`	Specifies the full path of a Visual Basic script to be run when a machine is destroyed. For example, `%System-Drive%\Program Files (x86)\VMware\vCAC Agents\EPI_Agent\Scripts \SendEmail.vb`. The script file must reside on the system on which the Visual Basic script EPI agent is installed.
`VCloud.Lease.Sync.TimeBufferMins`	Specifies a threshold integer value for a compute resource such that lease synchronization between vCloud Director and vRealize Automation only occur for vCloud Director or vCloud Air-provisioned machines that are set to expire in vCloud Director or vCloud Air in that time period. If a conflict is found, the lease value is synchronized to match the lease length defined in vRealize Automation. The default `VCloud.Lease.Sync.TimeBufferMins` value is 720 minutes, which is 12 hours. If `VCloud.Lease.Sync.TimeBufferMins` is not present, the default value is used. For example, if the default values are used, vRealize Automation runs the lease synchronization check workflow every 45 minutes, which is the workflow default, and only the leases of machines that are set to expire within 12 hours are changed to match the lease length defined in vRealize Automation.
`VCloud.Owner.UseEndpointAccount`	Set to true to assign the endpoint account as the vCloud Air or vCloud Director machine owner for provisioning and import operations. For change ownership operations, the owner is not changed on the endpoint. If not specified or set to false, the vRealize Automation owner is the machine owner.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VCloud.Template.MakeIdenticalCopy`	Set to true to clone an identical copy of the vCloud Air or vCloud Director template for machine provisioning. The machine is provisioned as an identical copy of the template. Settings specified in the template, including storage path, supersede settings specified in the blueprint. The only changes from the template are the names of the cloned machines, which are generated from the machine prefix specified in the blueprint. vCloud Air or vCloud Director machines that are provisioned as identical copies can use networks and storage profiles that are not available in the vRealize Automation reservation. To avoid having unaccounted reservation allocations, verify that the storage profile or network specified in the template is available in the reservation.
`VCNS.LoadBalancerEdgePool.Names.`*name*	Specifies the NSX load balancing pools to which the virtual machine is assigned during provisioning. The virtual machine is assigned to all service ports of all specified pools. The value is an *edge/pool* name or a list of *edge/pool* names separated by commas. Names are case-sensitive. **Note** You can add a machine IP address to an existing load balancer by using the `VCNS.LoadBalancerEdgePool.Names` custom property. vRealize Automation and NSX use the first member of the specified edge load balancer pool to determine the new member port and monitor port settings. However, NSX 6.2 does not require that the member port setting be specified. To avoid provisioning failure when using `VCNS.LoadBalancerEdgePool.Names` with NSX 6.2 to add a machine to an existing pool, you must specify a port value for the first member of the load balancer pool in NSX. Appending a name allows you to create multiple versions of a custom property. For example, the following properties might list load balancing pools set up for general use and machines with high, moderate, and low performance requirements: ■ `VCNS.LoadBalancerEdgePool.Names` ■ `VCNS.LoadBalancerEdgePool.Names.moderate` ■ `VCNS.LoadBalancerEdgePool.Names.high` ■ `VCNS.LoadBalancerEdgePool.Names.low`
`VCNS.SecurityGroup.Names.`*name*	Specifies the NSX security group or groups to which the virtual machine is assigned during provisioning. The value is a security group name or a list of names separated by commas. Names are case-sensitive. Appending a name allows you to create multiple versions of the property, which can be used separately or in combination. For example, the following properties can list security groups intended for general use, for the sales force, and for support: ■ `VCNS.SecurityGroup.Names` ■ `VCNS.SecurityGroup.Names.sales` ■ `VCNS.SecurityGroup.Names.support`

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VCNS.SecurityGroup.Names`.*blueprint_name*	When using NSX, specifies the Edge Pool with which to associate the blueprint.
`VCNS.SecurityTag.Names.`*name*	Specifies the NSX security tag or tags to which the virtual machine is associated during provisioning. The value is a security tag name or a list of names separated by commas. Names are case-sensitive. Appending a name allows you to create multiple versions of the property, which can be used separately or in combination. For example, the following properties can list security tags intended for general use, for the sales force, and for support: ■ `VCNS.SecurityTag.Names` ■ `VCNS.SecurityTag.Names.sales` ■ `VCNS.SecurityTag.Names.support`
`VirtualMachine.Admin.UseGuestAgent`	If the guest agent is installed as a service on a template for cloning, set to True on the machine blueprint to enable the guest agent service on machines cloned from that template. When the machine is started, the guest agent service is started. Set to False to deactivate the guest agent. If set to False, the enhanced clone workfow will not use the guest agent for guest operating system tasks, reducing its functionality to `VMwareCloneWorkflow`. If not specified or set to anything other than False, the enhanced clone workflow sends work items to the guest agent. This property does not apply to Amazon Web Services provisioning.
`VirtualMachine.Admin.NameCompletion`	Specifies the domain name to include in the fully qualified domain name of the machine that the RDP or SSH files generate for the user interface options **Connect Using RDP** or **Connect Using SSH** option. For example, set the value to myCompany.com to generate the fully qualified domain name *`my-machine-name`*`.myCompany.com` in the RDP or SSH file.
`VirtualMachine.Admin.ConnectAddress`	Specifies the RDP connection address of the machine to which an RDP file is downloaded when the user interface option **Connect Using RDP** is used or attached to automatic emails. Do not use in a blueprint or property group unless you require the user to be prompted and you have not supplied a default value.
`VirtualMachine.Admin.ConnectAddress.Regex`	Used by a vRealize Automation administrator to define a regular expression to match an IP address for terminal connections, such as an RDP connection. If matched, the IP address is saved under the `VirtualMachine.Admin.ConnectAddress` custom property. Otherwise, the first available IP address is designated. For example, setting the property value to `10.10.0.` allows selection of an IP address from a 10.10.0.* subnet that is assigned to the virtual machine. If the subnet has not been assigned, the property is ignored. This property is available for use with OpenStack.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.Admin.ThinProvision`	Determines whether thin provisioning is used on ESX compute resources. Disk provisioning is abstracted from the underlying storage. Set to True to use thin provisioning. Set to False to use standard provisioning. This property is for virtual provisioning.
`VirtualMachine.Admin.CustomizeGuestOSDelay`	Specifies the time to wait after customization is complete and before starting the guest operating system customization. The value must be in HH:MM:SS format. If the value is not set, the default value is one minute (00:01:00). If you choose not to include this custom property, provisioning can fail if the virtual machine reboots before guest agent work items are completed, causing provisioning to fail. This property does not apply to Amazon Web Services provisioning.
`VirtualMachine.Admin.Datastore.Cluster.ResourceLeaseDurationSec`	When provisioning to multiple VMs and using SDRS, specifies a value in seconds, in the range of 30 to 3600, for reserving storage resources during the `RecommendDataStore` API call. You can add this property to a business group or blueprint or when you request provisioning. The lease lock is only applied to the datastore that is used by the deployment, not all datastores in the storage cluster. The lease lock is released when provisioning either completes or fails. If not specified, no lock is applied to the storage resources at provisioning time. Because of memory size considerations, requesting more than 10 VMs simultaneously can cause provisioning failures.
`VirtualMachine.Admin.NetworkInterfaceType`	Indicates the network adapter type that is supported and emulated by the guest operating system. Use to create a new virtual machine and assign a specific adapter type for a template cloning operation. Use to modify the network settings of a newly provisioned virtual machine. The following options are available: ■ E1000 (default) ■ VirtIO ■ RTL8139 ■ RTL8139 VirtIO
`VirtualMachine.Admin.Name`	Specifies the generated machine name for vSphere, for example CodyVM01. When creating custom workflows or plug-ins for customizing a virtual machine name, set this property to match the name of the virtual machine. This is an internal input property for the agent to name the virtual machine. **Note** This property is for vSphere only. The value specified in the blueprint has no effect on this property. This property is not intended to be used to prompt the user. Use the `HostName` property to prompt the user. If the property is set at runtime, the container name that is created in the hypervisor might not match the item record name.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.Admin.UUID`	Specifies the UUID of the machine. The guest agent records the value when the machine is created. The value becomes read-only. The value in the blueprint or property group has no effect on this property.
`VirtualMachine.Admin.AgentID`	Specifies the UUID of the guest agent. The guest agent recorsd the value when the machine is created. The value becomes read-only. The value in the blueprint or property group has no effect on this property.
`VirtualMachine.Admin.Owner`	Specifies the user name of the machine owner.
`VirtualMachine.Admin.Approver`	Specifies the user name of the group manager who approved the machine request.
`VirtualMachine.Admin.Description`	Specifies the description of the machine as entered or modified by its owner or an administrator.
`VirtualMachine.Admin.EncryptPasswords`	If set to True, specifies that the administrator passwords are encrypted.
`VirtualMachine.Admin.AdministratorEmail`	Specifies the manager email addresses or Active Directory accounts for the business group of the provisioning blueprint. Multiple email addresses are separated by a comma, for example `AlbertAdmin@VMware.com,WeiLeeMgr@VMware.com`.
`VirtualMachine.Admin.TotalDiskUsage`	Specifies the total disk space in GB that the machine uses, including all disks as specified by the `VirtualMachine.DiskN.Size` properties and the swap file as specified by the `VMware.Memory. Reservation` property. You specify the value in GB, but the disk space is stored by vRealize Automation in MB.
`VirtualMachine.Admin.Hostname`	Informs the administrator which host is used for provisioning the machine on the endpoint. The specified value is implemented on the machine and is populated during data collection. For example, if the compute resource of a machine is changed, a proxy agent updates the value of the machine's `VirtualMachine.Admin.Hostname` property. **Note** This is an internal output property from the agent that is populated during the data collection process and identifies the host on which a machine resides.
`VirtualMachine.Admin.ClusterName`	Informs the administrator which cluster contains the compute resource for the machine to use. **Note** This is an internal output property from the agent that is populated during the data collection process and identifies the cluster in which a machine resides.
`VirtualMachine.Admin.ApplicationID`	List the application IDs that can be assigned to a machine.

Table 8-45. Custom Properties V Table (continued)

Property	Description
VirtualMachine.Admin.AddOwnerToAdmins	Set to True (default) to add the machine's owner, as specified by the VirtualMachine.Admin.Owner property, to the local administrators group on the machine. This property is not available for provisioning by cloning.
VirtualMachine.Admin.AllowLogin	Set to True (default) to add the machine owner to the local remote desktop users group, as specified by the VirtualMachine.Admin.Owner property.
VirtualMachine.Admin.DiskInterfaceType	Indicates the type of disk drivers. The following disk drivers are supported: ■ IDE (default) ■ VirtIO This property is for virtual provisioning.
VirtualMachine.Admin.EagerZero	When set to true, specifies that the machine disks are provisioned using the VMware provisioning format of eager zero. Thick provision eager zero is a type of thick virtual disk that supports clustering features such as fault tolerance. Space required for the virtual disk is allocated at creation time. In contrast to the flat format, the data remaining on the physical device is zeroed out when the virtual disk is created. It might take much longer to create disks in this format than to create other types of disks.
VirtualMachine.Admin.ForceHost	Specifies the name of the ESX host. The property is only honored if VirtualMachine.Admin.HostSelectionPolicy is set to EXACT_MATCH. **Note** This property is for vSphere only. When provisioning against a vSphere cluster, you can use the VirtualMachine.Admin.ForceHost property to specify the host on which a machine is to be provisioned. This property is used only if DRS is not set to automatic for the cluster. If the cluster has DRS enabled and is set to Automatic, vSphere relocates the provisioned machine when the machine is restarted.
VirtualMachine.Admin.HostSelectionPolicy	Optionally set to EXACT_MATCH to require the machine to be placed on the host specified by the VirtualMachine.Admin.ForceHost property. If the host is unavailable, the request results in a failure. If a host is not specified, the next best available host is selected. If set to EXACT_MATCH, an error occurs if the specified host does not have enough memory or is in maintenance mode. **Note** This property applies to vSphere only.
VirtualMachine.Agent.CopyToDisk	Set to True (default) to copy the guest agent executable file to %System–Drive%\VRM\Build\Bin on the machine's disk.
VirtualMachine.Agent.GuiRunOnce	Set to True to include guest agent execution in the SysPrep.inf run once section. Set to False for the Linux agent to stop the provisioning workflow.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.Agent.Reboot`	Set to True (default) to specify that the guest agent restarts the machine following installation of the guest operating system.
`VirtualMachine.CDROM.Attach`	Set to False to provision the machine without a CD-ROM device. The default is True.
`VirtualMachine.CPU.Count`	Specifies the number of CPUs, for example 2, allocated to a machine. The default is the value specified by the CPU setting on the blueprint. **Note** This custom property value is overridden by the CPU value on the blueprint when the machine is first provisioned.
`VirtualMachine.Customize.WaitComplete`	Set to True to prevent the provisioning workflow from sending work items to the guest agent until all customizations arecomplete. Set to False to allow work items to be created before customization is complete. This property does not apply to Amazon Web Services provisioning.
`VirtualMachine.Core.Count`	If set to a value greater than zero, specifies the number of cores per socket when provisioning the virtual machine. You can use this property on a blueprint to specify cores per virtual socket or total number of sockets. For example, your licensing terms might restrict software that is licensed per socket or available operating systems only recognize so many sockets and additional CPUs must be provisioned as additional cores.
`VirtualMachine.DiskN.Letter`	Specifies the drive letter or mount point of a machine's disk *N*. The default is C. For example, to specify the letter D for Disk 1, define the custom property as `VirtualMachine.Disk1.Letter` and enter the value D. Disk numbering must be sequential. When used in conjunction with a guest agent, this value specifies the drive letter or mount point under which an additional disk *N* is mounted by the guest agent in the guest operating system.
`VirtualMachine.DiskN.IsFixed`	Deactivates the editing of a specific disk when reconfiguring a machine. Set to True to deactivate display of the edit capacity option for a specific volume. The True value is case-sensitive. The *N* value is the 0-based index of the disk. Alternatively, you can set the `VirtualMachine.Disk`*`N`*`.IsFixed` custom property to True in the VirtualMachineProperties table in the database or use the Repository API to specify a URI value such as `.../Repository/Data/ManagementModelEntities.svc/VirtualMachines(guid'60D93A8A-F541-4CE0-A6C6-78973AC0F1D2')/VirtualMachineProperties`.
`VirtualMachine.DiskN.Label`	Specifies the label for a machine's disk *N*. The disk label maximum is 32 characters. Disk numbering must be sequential. When used with a guest agent, specifies the label of a machine's disk *N* inside the guest operating system.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.DiskN.Active`	Set to True (default) to specify that the machine's disk *N* is active. Set to False to specify that the machine's disk *N* is not active.
`VirtualMachine.DiskN.FS`	For use with Windows guest agent (gugent). Specifies the file system of the machine's disk *N*. The options are NTFS (default), FAT and FAT32. For example usage, see the `10_setupdisks.bat` Windows agent script.
`VirtualMachine.DiskN.FileSystem`	For use with Linux guest agent (gugent). Specifies the file system of the machine's disk *N*. The options are ext3, ext4, and XFS. For example usage, see the `30_DiskSetup.sh` Linux agent script.
`VirtualMachine.DiskN.Percent`	Specifies the percentage of the disk *N* to be formatted by a guest agent for the machine's use. That machine cannot use the remaining portion of the disk.
`VirtualMachine.DiskN.StorageReservationPolicy`	Specifies the storage reservation policy to use to find storage for disk *N*. Also assigns the named storage reservation policy to a volume. To use this property, substitute the volume number for *N* in the property name and specify a storage reservation policy name as the value. This property is equivalent to the storage reservation policy name specified on the blueprint. Disk numbering must be sequential. This property is valid for all Virtual and vCloud reservations. This property is not valid for Physical, Amazon, or OpenStack reservations. You can use `VirtualMachine.DiskN.StorageReservationPolicyMode` to prevent provisioning from failing if there is insufficient space on the datastores in a storage reservation policy. Use this custom property to allow vRealize Automation to select a datastore outside the specified storage reservation policy in cases where there is not sufficient space remaining on the datastores in the policy.
`VirtualMachine.DiskN.StorageReservationPolicyMode`	Allocates disk *N* to the best available storage reservation policy.
`VirtualMachine.DiskN.Storage`	Specifies the datastore on which to place the machine disk *N*, for example DATASTORE01. This property is also used to add a single datastore to a linked clone blueprint. *N* is the index (starting at 0) of the volume to assign. Enter the name of the datastore to assign to the volume. This is the datastore name as it appears in the Storage Path on the Edit Compute Resource page. Disk numbering must be sequential.
`VirtualMachine.EPI.Type`	Specifies the type of external provisioning infrastructure. Set to BMC for BMC BladeLogic integration. Set to CitrixProvisioning for Citrix provisioning server integration.
`VirtualMachine.EULA.AcceptAll`	Set to true to specify that all the EULAs for the VM templates of the vCloud Air or vCloud Director endpoints are accepted during provisioning.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.Host.TpmEnabled`	Limits virtual machine placement to hosts that have a Trust Protection Module (TPM) device installed and recognized by ESX and vSphere. The default value is False. All hosts in a cluster must have a Trust Protection Module device installed. If no acceptable hosts or clusters are found, the machine cannot be provisioned until this property is removed.
`VirtualMachine.Memory.Size`	Specifies the size of the machine's memory in MB, such as 1024. The default is the value specified by the memory setting on the blueprint. **Note** This custom property setting is overridden by the memory setting on the blueprint when the machine is first provisioned.
`VirtualMachine.NetworkN.Address`	Specifies the IP address of network device *N* in a machine provisioned with a static IP address. `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks.
`VirtualMachine.NetworkN.AdditionAddress`*M*	Defines additional *M* IP address allocated for an OpenStack instance for network *N*, excluding the IP address set specified by the `VirtualMachine.Network`*N*`.Address`. property. More addresses are displayed on the Network tab in the Additional Addresses column. This property is used by OpenStack machine state data collection. While this property is only data-collected by the OpenStack endpoint, it is not specific to OpenStack and can be used for lifecycle extensibility by other endpoint types. This property is not supported for on-demand NAT or on-demand routed networks.
`VirtualMachine.NetworkN.AddressType`	Specifies how IP address allocation is supplied to the network provider, where Network*N* is the network number, starting with 0. The following values are available: ■ DHCP ■ Static ■ MANUAL (available for vCloud Air and vCloud Director only) The MANUAL value also requires that you specify an IP address. This property is available for configuring vCloud Air, vCloud Director, and vSphere machine components in the blueprint. Also see `VirtualMachine.NetworkN.Name`. This property is not supported for on-demand NAT or on-demand routed networks.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.NetworkN.MacAddressType`	Indicates whether the MAC address of network device *N* is generated or user-defined (static). This property is available for cloning. The default value is generated. If the value is static, you must also use `VirtualMachine.NetworkN.MacAddress` to specify the MAC address. `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks.
`VirtualMachine.NetworkN.MacAddress`	Specifies the MAC address of a network device *N*. This property is available for cloning. If the value of `VirtualMachine.NetworkN.MacAddressType` is generated, this property contains the generated address. If the value of `VirtualMachine.NetworkN.MacAddressType` is static, this property specifies the MAC address. For virtual machines provisioned on ESX server hosts, the address must be in the range specified by VMware. For details, see vSphere documentation. `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.NetworkN.Name`	Specifies the name of the network to connect to, for example the network device *N* to which a machine is attached. This is equivalent to a network interface card (NIC). By default, a network is assigned from the network paths available on the reservation on which the machine is provisioned. Also see `VirtualMachine.NetworkN.AddressType`. You can ensure that a network device is connected to a specific network by setting the value of this property to the name of a network on an available reservation. For example, if you give properties for N= 0 and 1, you get 2 NICs and their assigned value, provided the network is selected in the associated reservation. `VirtualMachine.Network`*N* custom properties are specific to blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks. For an example of how to use this custom property to dynamically set `VirtualMachine.Network0.Name` based on a consumer's selection from a list of predefined available networks, see the Adding a Network Selection Drop-Down in vRA 7 blog post.
`VirtualMachine.NetworkN.PortID`	Specifies the port ID to use for network device *N* when using a dvPort group with a vSphere distributed switch. `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. This property is not supported for on-demand NAT or on-demand routed networks.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.NetworkN.NetworkProfileName`	Specifies the name of a network profile from which to assign a static IP address to network device *N* or from which to obtain the range of static IP addresses that can be assigned to network device *N* of a cloned machine, where *N*=0 for the first device, 1 for the second, and so on. The network profile that the property points to is used to allocate an IP address. The property determines the network that the machine attaches to, based on the reservation. Changing this property value after the network is assigned has no effect on the expected IP address values for the designated machines. With WIM-based provisioning for virtual machines, you can use this property to specify a network profile and network interface or you can use the Network section of the Virtual Reservation page. The following attributes of the network profile are available to enable static IP assignment in a cloning blueprint: ■ `VirtualMachine.NetworkN.SubnetMask` ■ `VirtualMachine.NetworkN.Gateway` ■ `VirtualMachine.NetworkN.PrimaryDns` ■ `VirtualMachine.NetworkN.SecondaryDns` ■ `VirtualMachine.NetworkN.PrimaryWins` ■ `VirtualMachine.NetworkN.SecondaryWins` ■ `VirtualMachine.NetworkN.DnsSuffix` ■ `VirtualMachine.NetworkN.DnsSearchSuffixes` `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. You cannot use this custom property to define an on-demand NAT or on-demand routed network profile name. Because on-demand network profile names are generated at allocation time (during provisioning), their names are unknown when creating or editing the blueprint. To specify NSX on-demand network information, use the applicable network component in the blueprint design canvas for your vSphere machine components.

Table 8-45. Custom Properties V Table (continued)

Property	Description
■ `VirtualMachine.NetworkN.SubnetMask` ■ `VirtualMachine.NetworkN.Gateway` ■ `VirtualMachine.NetworkN.PrimaryDns` ■ `VirtualMachine.NetworkN.SecondaryDns` ■ `VirtualMachine.NetworkN.PrimaryWins` ■ `VirtualMachine.NetworkN.SecondaryWins` ■ `VirtualMachine.NetworkN.DnsSuffix` ■ `VirtualMachine.NetworkN.DnsSearchSuffixes`	Configures attributes of the network profile specified in `VirtualMachine.NetworkN.NetworkProfileName`. `VirtualMachine.Network`*N* custom properties are specific to individual blueprints and machines. When a machine is requested, network and IP address allocation is performed before the machine is assigned to a reservation. Because blueprints are not guaranteed to be allocated to a specific reservation, do not use this property on a reservation. When specifying values for multiple DNS search suffixes using `VirtualMachine.NetworkN.DnsSearchSuffixes`, you can use commas to separate values for a Windows deployment. These properties are not supported for on-demand NAT or on-demand routed networks.
`VirtualMachine.Rdp.File`	Specifies the file that contains RDP settings to be used when opening an RDP link to the machine. Can be used together with, or as an alternative to, `VirtualMachine.Rdp.Setting`*N*. The file must be located in the *vRA_installation_dir*\Server\Website \Rdp folder. You must create the `Rdp` directory. For related information, see `VirtualMachine.Rdp.SettingN`.
`VirtualMachine.Rdp.SettingN`	Specifies the RDP settings to be used when opening an RDP link to the machine. *N* is a unique number used to distinguish one RDP setting from another. For example, to specify the RDP authentication level so that no authentication requirement is specified, define the custom property `VirtualMachine.Rdp.Setting1` and set the value to authentication level:i:3. For information about available RDP settings, and their correct syntax, see Microsoft Windows RDP documentation such as RDP Settings for Remote Desktop Services in Windows Server. For related information, see `VirtualMachine.Rdp.File`.
`VirtualMachine.Reconfigure.DisableHotCpu`	Set to true to specify that the reconfigure machine action restarts the specified machine. By default, the reconfigure machine action does not restart the machine. Performing a hot add of CPU, memory, or storage causes the reconfigure machine action to fail and not to restart the machine unless the `Hot Add` setting is enabled in vSphere for the machine or template. You can add `VirtualMachine.Reconfigure.DisableHotCpu=true` to a machine component in a vRealize Automation blueprint to deactivate the `Hot Add` setting and force the machine to restart regardless of the vSphere `Hot Add` setting. The custom property is only available for machine types that support hardware reconfiguration, which are vSphere, vCloud Air, and vCloud Director.
`VirtualMachine.Request.Layout`	Specifies the property layout to be used in the virtual machine request page. The value must match the name of the layout to be used.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.Software.Execute`	When set to True, supports guest agents for Amazon Web Services provisioning. Use this property with the `VirtualMachine.SoftwareN.Name` and `VirtualMachine.SoftwareN.ScriptPath` custom properties to configure and use guest agents in Amazon Web Services provisioning.
`VirtualMachine.SoftwareN.Name`	Specifies the descriptive name of a software application *N* or script to install or run during provisioning. This is an optional and information-only property. It serves no real function for the enhanced clone workflow or the guest agent but it is useful for a custom software selection in a user interface or for software use reporting.
`VirtualMachine.SoftwareN.ScriptPath`	Specifies the full path to an application's install script. The path must be a valid absolute path as seen by the guest operating system and must include the name of the script filename. You can pass custom property values as parameters to the script by inserting {*CustomPropertyName*} in the path string. For example, if you have a custom property named `ActivationKey` whose value is 1234, the script path is `D:\InstallApp.bat -key {ActivationKey}`. The guest agent runs the command `D:\InstallApp.bat -key 1234`. Your script file can then be programmed to accept and use this value. You can also pass custom property values as parameters to the script by inserting {*YourCustomProperty*} in the path string. For example, entering the value **`\\vra-scripts.mycompany.com\scripts\changeIP.bat`** runs the `changeIP.bat` script from a shared location, but entering the value **`\\vra-scripts.mycompany.com\scripts\changeIP.bat {VirtualMachine.Network0.Address}`** runs the changeIP script but also passes the value of the `VirtualMachine.Network0.Address` property to the script as a parameter. Insert {Owner} to pass the machine owner name to the script.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.ScriptPath.Decrypt`	Allows vRealize Automation to obtain an encrypted string that is passed as a properly formatted `VirtualMachine.SoftwareN.ScriptPath` custom property statement to the gugent command line. You can provide an encrypted string, such as your password, as a custom property in a command-line argument. This allows you to store encrypted information that the guest agent can decrypt and understand as a valid command-line argument. For example, the `VirtualMachine.Software0.ScriptPath = c:\dosomething.bat` *`password`* custom property string is not secure as it contains an actual password. To encrypt the password, you can create a vRealize Automation custom property, for example `MyPassword = password`, and enable encryption by selecting the available check box. The guest agent decrypts the **`[MyPassword]`** entry to the value in the custom property `MyPassword` and runs the script as `c:\dosomething.bat password`. ■ Create custom property **`MyPassword =`** ***`password`*** where *password* is the value of your actual password. Enable encryption by selecting the available check box. ■ Set custom property `VirtualMachine.ScriptPath.Decrypt` as **`VirtualMachine.ScriptPath.Decrypt = true`**. ■ Set custom property `VirtualMachine.Software0.ScriptPath` as **`VirtualMachine.Software0.ScriptPath = c:\dosomething.bat [MyPassword]`**. If you set `VirtualMachine.ScriptPath.Decrypt` to false, or do not create the `VirtualMachine.ScriptPath.Decrypt` custom property, then the string inside the square brackets ([and]) is not decrypted.
`VirtualMachine.SoftwareN.ISOName`	Specifies the path and filename of the ISO file relative to the datastore root. The format is *`/folder_name/subfolder_name/file_name`*`.iso`. If a value is not specified, the ISO is not mounted.
`VirtualMachine.SoftwareN.ISOLocation`	Specifies the storage path that contains the ISO image file to be used by the application or script. Format the path as it appears on the host reservation, for example `netapp-1:it_nfs_1`. If a value is not specified, the ISO is not mounted.
`VirtualMachine.Storage.Name`	Identifies the storage path on which the machine resides. The default is the value specified in the reservation that was used to provision the machine.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VirtualMachine.Storage.AllocationType`	Stores collected groups to a single datastore. A distributed environment stores disks round-robin style. Specify one of the following values: ■ Collected Keep all disks together. ■ Distributed Allow disks to be placed on any datastore or datastore cluster that is available in the reservation. For an example of how to use the `VirtualMachine.Storage.AllocationType` property to create datastore clusters, see the Keeping Multiple Disks Together blog post.
`VirtualMachine.Storage.Cluster.Automation.Enabled`	If set to True, the storage cluster automation on the machine is enabled. If set to False, then storage cluster automation is deactivated on the machine. The storage cluster automation type is determined by the `VirtualMachine.Storage.Cluster.Automation.Behavior` custom property.
`VirtualMachine.Storage.Cluster.Automation.Behavior`	Specifies an SDRS behavior type when `VirtualMachine.Storage.Cluster.Automation.Enabled` is set to True. The available behavior type values are automated or manual. The `VirtualMachine.Storage.Cluster.Automation.Enabled` and `VirtualMachine.Storage.Cluster.Automation.Behavior` properties are set after the machine is provisioned and after inventory data collection is finished. If automation is deactivated, `VirtualMachine.Storage.Cluster.Automation.Behavior` is not present on the machine.
`VirtualMachine.Storage.ReserveMemory`	Set to True to manage vSwap storage allocation to ensure availability and set allocation in the reservation. vSwap allocation is considered when you create or reconfigure a virtual machine. vSwap allocation checking is only available for vSphere endpoints. **Note** If you do not specify the `VirtualMachine.Storage.ReserveMemory` custom property when you create or provision the machine from vRealize Automation, swap space availability is not ensured. If you add the property for an already provisioned machine, and the allocated reservation is full, the storage allocated in the reservation might exceed the actual allocated storage.
`VirtualMachine.VDI.Type`	Specifies the type of virtual desktop infrastructure. For XenDesktop provisioning, set to XenDesktop.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VMware.AttributeN.Name`	Specifies the name of an attribute in vRealize Orchestrator. For example, it specifies the value of the attribute used in the `VMware.AttributeN.Name` property. Replace the letter *N* with a number, starting with 0 and increasing for each attribute to set.
`VMware.AttributeN.Value`	Specifies the value of the attribute used in the `VMware.AttributeN.Name` property. Replace the letter *N* with a number, starting with 0 and increasing for each attribute to set.
`VMware.Endpoint.Openstack.IdentityProvider.Domain.Name`	Allows vRealize Automation to support required Keystone V3 domain-name authentication. If Keystone V3 is in effect, you can use the property to designate a specific domain for the OpenStack endpoint to authenticate with a Keystone V3 OpenStack identity provider. ■ For new endpoints, add the custom property to designate a specific domain. ■ For upgraded or migrated endpoints, add the custom property only if data collection fails after upgrade or migration.
`VMware.Endpoint.Openstack.IdentityProvider.Version`	Specifies the version of OpenStack identity provider (Keystone) to use when authenticating an OpenStack endpoint. Configure a value of **3** to authenticate with Keystone V3 OpenStack identity provider. If you use any other value, or do not use this custom property, authentication defaults to Keystone V2.
`VMware.Endpoint.Openstack.Release`	Deprecated. Specifies the OpenStack release, for example Havana or Icehouse, when creating an OpenStack endpoint. Required for 6.2.1, 6.2.2, and 6.2.3 OpenStack provisioning.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VMware.Endpoint.NSX.HideDiscoveredSecurityObjects`	Set to true to hide newly discovered security objects in the active tenant for the NSX endpoints to which the security objects are associated. Otherwise, all new security objects are available to all tenants after data collection, provided that the security object is for an endpoint in which you have a reservation. This option allows you to prevent users from accessing security objects when you want to assign those objects to a single tenant or to mask from all tenants. Set to false to toggle back to global, which enables all new security objects to be available to all tenants after data collection, provided that the security object is for an endpoint in which you have a reservation. To take effect, the fabric administrator adds the `VMware.Endpoint.NSX.HideDiscoveredSecurityObjects` custom property to the associated NSX endpoint that is associated to a vSphere endpoint. The setting applies to the next inventory data collection. Existing security objects remain unchanged. To change the tenancy setting of a security object that has already been data-collected, such as existing security objects after upgrading to the current vRealize Automation release, you can edit the security object's Tenant ID setting programmatically by using the vRealize Automation REST API or vRealize CloudClient. The available Tenant ID settings for the NSX endpoint are as follows: ■ "`<global>`" - the security object is available to all tenants. This is the default setting for existing security objects after upgrade to this release and for all new security objects that you create. ■ "`<unscoped>`" - the security object is not available to any tenants. Only the system administrator can access the security object. This is an ideal setting when defining security objects that are to eventually be assigned to a specific tenant. ■ "*`tenant_id_name`*" - the security object is only available to a single, named tenant.
`VMware.Hardware.Version`	Specifies the VM hardware version to be used for vSphere settings. Supported values are currently vmx-04, vmx-07, vmx-08, vmx-09 and vmx-10. This property is applicable for VM Create and VM Update workflows and is available only for basic workflow blueprints.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VMware.VirtualCenter.OperatingSystem`	Specifies the vCenter Server guest operating system version (`VirtualMachineGuestOsIdentifier`) with which vCenter Server creates the machine. This operating system version must match the operating system version to be installed on the provisioned machine. Administrators can create property groups using one of several property sets, for example, `VMware[OS_Version]Properties`, that are predefined to include the correct `VMware.VirtualCenter.OperatingSystem` values. This property is for virtual provisioning. When this property has a non-Windows value, the **Connect Using RDP** user interface option is deactivated. The property can be used in a virtual, cloud or physical blueprint. For related information, see the enumeration type `VirtualMachineGuestOsIdentifier` in vSphere API/SDK Documentation. For a list of currently accepted values, see the vCenter Server documentation.
`VMware.SCSI.Type`	For vCloud Air, vCloud Director, or vSphere machine components in blueprints, specifies the SCSI machine type using one of the following case-sensitive values: ■ buslogic Use BusLogic emulation for the virtual disk. ■ lsilogic Use LSILogic emulation for the virtual disk (default). ■ lsilogicsas Use LSILogic SAS 1068 emulation for the virtual disk. ■ pvscsi Use para-virtualization emulation for the virtual disk. ■ none Use if a SCSI controller does not exist for this machine. The `VMware.SCSI.Type` property is not available for use with the CloneWorkflow provisioning workflow. If you specify the CloneWorkflow provisioning workflow when configuring your machine component in the blueprint design canvas, you cannot use the `VMware.SCSI.Type` property.
`VMware.SCSI.Sharing`	Specifies the sharing mode of the machine's VMware SCSI bus. Possible values are based on the `VirtualSCSISharing` ENUM value and include noSharing, physicalSharing, and virtualSharing. If you specify the CloneWorkflow provisioning workflow when configuring your machine component in the blueprint design canvas, the `VMware.SCSI.Sharing` property is not available. The `VMware.SCSI.Sharing` property is not available for use with the CloneWorkflow provisioning workflow. If you specify the CloneWorkflow provisioning workflow when configuring your machine component in the blueprint design canvas, you cannot use the `VMware.SCSI.Sharing` property.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VMware.Memory.Reservation`	Defines the amount of reserved memory for the virtual machine in MB, for example 1024. Setting this value also reduces the size of the virtual machine swap file on disk by the amount specified.
`VMware.Network.Type`	Specifies the network to connect the VM as specified in the reservation. The network adapter on the machine must be connected to a unique network. The following adapter type values are available: - Flexible (default) - VirtualPCNet32 (not compatible with vSphere). - E1000 or VirtualE1000 - VMXNET or VirtualVMXNET - VMXNET2 - VMXNET3 Set to E1000 when provisioning Windows 32-bit virtual machines on ESX server hosts to ensure that machines are created with the correct network adapter. This property is not used for physical provisioning.
`VMware.Ovf.Thumbprint`	If the OVF resides on an HTTPS server that has a certificate, this property stores the value of that certificate's thumbprint and is used to validate that certificate. It has no relevance when the OVF is hosted on an HTTP server. The property is automatically created when you import an OVF by using the `ImportOvfWorkflow` provisioning workflow in the blueprint component's user interface. If you create the blueprint programmatically with vRealize Automation REST APIs or vRealize CloudClient, you must manually create the property. **Note** The thumbprint can be stored in a comma-separated format to support a certificate chain. When the `VMware.Ovf.TrustAllCertificates` is present and set to true, the `VMware.Ovf.Thumbprint` property is ignored.
`VMware.Ovf.TrustAllCertificates`	When this property is present and set to true, the `Vmware.Ovf.Thumbprint` property is ignored and no certificate validation is performed when you import an OVF by using the `ImportOvfWorkflow` provisioning workflow.
`VMware.Ovf.Configuration.`*X*	An OVF can contain user-configurable properties, for example a property that sets the root password of a VM provisioned from the OVF. When you import an OVF into a blueprint, the user-configurable properties that are defined in the OVF are parsed and converted into custom properties of the form `Vmware.Ovf.Configuration.`*X*, where *X* is the name of the user-configurable property from the OVF.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VMware.VCenterOrchestrator.EndpointName`	Overrides a specified endpoint setting or specifies that a particular endpoint be used during the vRealize Automation IaaS provisioning process. The value of this property can be set to an applicable vRealize Orchestrator endpoint, such as external VRO, available in the environment.
`VMware.VirtualCenter.Folder`	Specifies the name of the inventory folder in the data center in which to put the virtual machine. The default is VRM, which is also the vSphere folder in which vRealize Automation places provisioned machines if the property is not used. This value can be a path with multiple folders, for example `production\email servers`. A proxy agent creates the specified folder in vSphere if the folder does not exist. Folder names are case-sensitive. This property is available for virtual provisioning.
`VDI.Server.Website`	Specifies the server name of the Citrix Web interface site to use in connecting to the machine. If the value of `VDI.Server.Name` is a XenDesktop farm, this property must have an appropriate value or the machine owner cannot connect to the machine using XenDesktop. If this property is not specified, the `VDI.Server.Name` property determines the desktop delivery controller to connect to, which must be the name of a server that hosts a desktop delivery controller. **Note** If the Citrix Web Interface (WI) has been replaced with StoreFront (SF), you can use this property instead of `VDI.Server.Name` to connect to the XenDesktop server. An example value is `VDI.Server.Website=sqa-xddc-7.sqa.local/Citrix/StoreWeb`. See `VDI.Server.Name` for more information.

Table 8-45. Custom Properties V Table (continued)

Property	Description
`VDI.Server.Name`	Specifies the server name, which hosts the desktop delivery controller, to register with, or the name of a XenDesktop farm that contains desktop delivery controllers with which to register. If the value is a XenDesktop farm name, the `VDI.Server.Website` property value must be the URL of an appropriate Citrix web interface site to use in connecting to the machine. If the value is a server name, and at least one general XenDesktop VDI agent was installed without specifying a desktop delivery controller server, this value directs the request to the desired server. If the value is a server name, and only dedicated XenDesktop VDI agents for specific DDC servers were installed, this value must exactly match the server name configured for a dedicated agent. **Note** For more information about how to make StoreFront the default page in IIS, see Citrix documentation. See also `VDI.Server.Website`. **Note** Changes in the Citrix web interface protocol have impacted how the `VDI.Server.Name` default value is recognized. The value of the `VDI.Server.Name` property is used as the default connection string to open the Citrix web interface when users connect to a virtual desktop. It is always the DNS/IP of the XD server. If that value does not connect to the Citrix interface, you are unable to access your VMs. However, you can use the `VDI.Server.Website` custom property when the Citrix web interface is hosted on a server other than the XenDesktop server. When this property is present on the VM, it is used instead of `VDI.Server.Name`.
`VDI.Server.Group`	For XenDesktop 5, specifies the name of the XenDesktop group to add machines to and the name of the catalog to which the group belongs, in the *group_name;catalog_name* format. For XenDesktop 4, specifies the name of the XenDesktop group to which machines are to be added. XenDesktop 4 preassigned groups are supported.
`VDI.ActiveDirectory.Interval`	Specifies an optional interval value in time span format for virtual desktop infrastructure machine Active Directory registration check. The default value is 00:00:15 (15 seconds).
`VDI.ActiveDirectory.Timeout`	Specifies an optional timeout value to wait before retrying Active Directory registration. The default value is 00:00:15 (30 minutes).
`VDI.ActiveDirectory.Delay`	Specifies an optional delay time value in time span format between successfully adding a machine to Active Directory and initiation of XenDesktop registration. The default value is 00:00:05 (5 seconds).

Table 8-45. Custom Properties V Table (continued)

Property	Description
`Vrm.DataCenter.Location`	Allows you to use a blueprint to provision machines on more than one compute resource. You can add the `Vrm.DataCenter.Location` property to a blueprint, or enable the **Display Location on Request** option in the blueprint, to require that the user supply a data center location when they request machine provisioning. **Note** If you enable the **Display Location on Request** option on the blueprint, you do not need to also add the custom property. Data center locations are configured in a `DataCenterLocations.xml` file, which provides the location values that are applied to compute resources. For related information about adding data center locations, see *Scenario: Add Datacenter Locations for Cross Region Deployments*. Because the `Vrm.DataCenter.Location` property cannot access the contents of the `DatacenterLocations.xml` file, you must rely on users to provide property values that match the locations provided in the `DataCenterLocations.xml` file. Use this property if you want to use the data center location value as input to an external action for another custom property.
`Vrm.DataCenter.Policy`	Specifies whether provisioning must use a compute resource associated with a particular location, or if any location is suitable. To enable this feature, you must add data center to a location file. Associate each compute resource with a location. Set to Exact (default) to provision a requested machine on a compute resource associated with the location specified on the blueprint. The request fails if no reservations match the requested location. If the property is not present, the Exact default is used. Set to NonExact to provision a requested machine on a compute resource with sufficient capacity and associated with the location specified on the blueprint. If that compute resource is not available, then use the next available compute resource with sufficient capacity without regard to location.
`Vrm.ProxyAgent.Uri`	Allows you to override the default `Vrm.ProxyAgent.Uri` value that is derived from the VMPS endpoint address in the vRealize Automation Manager Service configuration file. The configuration setting is often set to the local machine but you might want to set it to the virtual IP (VIP) address. You can specify the `Vrm.ProxyAgent.Uri` custom property on a blueprint. Sample syntax is as follows: `Vrm.ProxyAgent.Uri`=https://*loadbalancer-vip*/VMPS2Proxy

Table 8-45. Custom Properties V Table (continued)

Property	Description

`Vrm.Software.Id`*NNNN*
This row is specific to BMC BladeLogic.
	Specifies a software job or policy to be applied to all machines provisioned from the blueprint. Set the value to `job_type=job_path`, where `job_type` is the numeral that represents the BMC BladeLogic job type and `job_path` is the location of the job in BMC BladeLogic, for example `4=/Utility/putty`. *NNNN* is a number from 1000 to 1999. The first property must start with 1000 and increment in numerical order for each additional property.
	`1 – AuditJob`
	`2 – BatchJob`
	`3 – ComplianceJob`
	`4 – DeployJob`
	`5 – FileDeployJob`
	`6 – NSHScriptJob`
	`7 – PatchAnalysisJob`
	`8 – SnapshotJob`

`Vrm.Software.Id`*NNNN* This row is specific to HP Server Automation.	(Optional) Specifies an HP Server Automation policy to be applied to all machines provisioned from the blueprint. *NNNN* is a number from 1000 to 1999. The first property must start with 1000 and increment in numerical order for each additional property.

Custom Properties X

A list of vRealize Automation custom properties that begin with the letter X.

Table 8-46. Custom Properties X Table

Property	Description
`Xen.Platform.Viridian`	For virtual provisioning, set to False when you provision Windows virtual machines on a XenServer host or pool. The default is True. This property is not used in physical provisioning.

Using the Property Dictionary

You can use the property dictionary to define new custom property definitions and property groups.

You define a property to support a specific data type and a display control style within that data type. You can also create reusable property groups to simplify adding multiple properties.

Using Property Definitions

Many custom properties are supplied with vRealize Automation. You can also define new properties to create unique custom properties and provide greater control for provisioning machines.

When you add a property to a blueprint or reservation, you can determine if a user must be prompted for a property value and if the property value must be encrypted.

You can specify how a property is rendered, for example if should display as a checkbox or as a drop-down menu with values obtained from a custom vRealize Orchestrator workflow.

You can also use properties to control how your custom workflows function. For information about using vRealize Automation Designer to define and work with custom workflows, see *Life Cycle Extensibility*.

Best Practices for Naming Property Definitions

To avoid naming conflicts with supplied vRealize Automation custom properties, use a standard and meaningful prefix for all property names that you create. Use a prefix such as a company or feature name followed by a dot for all new property names. VMware reserves all property names that do not contain a dot (.). Property names that do not follow this recommendation may conflict with vRealize Automation custom properties. In that event, the vRealize Automation custom property takes precedence over property definitions that you create.

General Procedures

The following steps describe the general procedure for creating and using new property definitions:

1 Create a new property definition and associate it with a data type that allows for a specific type of content, such as boolean or integer content. Use a standard naming convention for the new property name such as *my_grouping_prefix.my_property_name*.

2 Associate a property definition with a display type, such as a check box or drop-down menu. Available display types are derived from the selected data type.

3 Add the property to a blueprint either individually or as part of a property group.

   Add the property to a blueprint and specify if the property value must be encrypted.

   Add the property to a blueprint and specify if the user should be prompted to specify a property value.

4 As a machine requestor, specify required values as prompted.

Using vRealize Orchestrator Script Actions

You can populate the property value in a drop-down menu by using vRealize Orchestrator script actions. Using vRealize Orchestrator script actions also enables you to populate a drop-down menu value based on the values specified for another property.

You can use the `vra content list --type property-definition` vRealize CloudClient command to list all property definitions in the current vRealize Automation tenant. You can also use the `vra content list --type property-group` vRealize CloudClient command to list all property groups.

For a tutorial on creating dynamic property definitions to filter the options that are available to users, see the How to use dynamic property definitions blog post.

Limitations

If you create a property definition where `Data type` equals `String`, `Display as` equals `Dropdown`, and you use a vRealize Orchestrator action that returns properties that populate the drop-down list, the list is in random order. You cannot specify the order.

Create and Test Custom Property Definitions

You create a custom properties definition that determines how the custom property appears in vRealize Automation. You can add the custom property to a blueprint so that you can verify that the property displays the check box, drop-down menu, or other control type as expected.

To create and test the custom property definitions, you need a blueprint that is already entitled to you or to a test user account to which you have access. This test blueprint allows you to create the custom property, add it to a blueprint, and then verify that the custom property has the expected appearance. After you validate the custom property, you can add it to your production blueprints as needed.

Prerequisites

- Verify that you have a blueprint to which you are adding the action. See Configure a Machine Blueprint.
- Verity that the blueprint is entitled to you so that you can test the custom properties in the blueprint. See Entitle Users to Services, Catalog Items, and Actions.
- Log in to vRealize Automation as a **tenant administrator** or **fabric administrator**.

Procedure

1 Create Custom Property Definitions

   You create custom property definitions that determine how the custom property appears in vRealize Automation. You can validate the custom property in a test blueprint before adding it to your production blueprints.

2 Add a Custom Property to a Blueprint

   You can add custom properties to many parts of vRealize Automation, including approval policies, business groups, endpoints, and reservation policies. However, only the machine blueprints support the display options that you configure as property definitions. Adding a custom property to a blueprint as a simple way to verify that the custom property appears in the user interface as you designed it in the property definition.

3 Verify the Custom Property in the Catalog Request Form

   As creator of the custom property definitions that run vRealize Orchestrator actions, you should test your custom properties to ensure that the correct values appear in the request form.

Create Custom Property Definitions

You create custom property definitions that determine how the custom property appears in vRealize Automation. You can validate the custom property in a test blueprint before adding it to your production blueprints.

- Create a Property Definition

  You can create property definitions to allow for additional levels of vRealize Automation customization. When you create a property definition, you specify a data type for the property, for example a string or a boolean type.

- Create a Custom Property That Validates Against a Regular Expression

  You create custom property definition that evaluates a regular expression when you want service catalog users to provide validated data on the catalog request form.

- Create a vRealize Orchestrator Action Custom Property Definition

  You create a custom property definition that includes a vRealize Orchestrator action so that you can add the custom property to a blueprint. The action runs when the service catalog user is configuring the custom property in the request form. The action retrieves the data that is displayed in the form.

- Bind Custom Properties to Create a Parent-Child Relationship

  To create a parent-child relationship between custom properties, you bind the parent to the child. When you add the parent and child custom properties to a blueprint, the requesting user selects a value for the parent property. The selected parent value determines the possible values for the child property.

Create a Property Definition

You can create property definitions to allow for additional levels of vRealize Automation customization. When you create a property definition, you specify a data type for the property, for example a string or a boolean type.

To avoid conflict with supplied vRealize Automation custom properties, use a naming format of *my_prefix.my_property_name1*. For example, use a standard prefix such as company name, followed by a dot (.), followed by a descriptive name.

Properties that you create that do not follow this recommendation may conflict with vRealize Automation-supplied custom properties. In that event, the vRealize Automation custom properties take precedence over properties that you create.

Prerequisites

Log in to vRealize Automation as a **tenant administrator** or **fabric administrator**.

Procedure

1 Select **Administration > Property Dictionary > Property Definitions**.

2 Click **New** (+).

3. Enter the new property definition name in the **Name** text box.

   Use a standard naming convention for the new property name such as *my_grouping_prefix.my_property_name*.

   The **Name** value is stored internally as the property identifier (ID).

4. Accept the generated value in the **Label** text box.

   The **Label** value is automatically populated with the value that you entered in the **Name** text box. If you enter a **Label** value first, the **Name** text box is populated with the same value.

   The **Label** value is displayed in the user interface when requesting properties, for example when adding a property to a blueprint, as the property name.

   The **Label** value can contain a wider range of characters than the **Name** value.

5. In the **Visibility** section, select **All tenants** or **This tenant** to determine where the property is to be available.

   If you are logged in with only tenant administrator privileges, then only **This tenant** is available. If you are logged in with only fabric administrator privileges, then only **All tenants** is available.

   You cannot change the **All tenants** or **This tenant** setting after you create the item.

6. (Optional) Enter a property description in the **Description** text box.

   Describe the intent of the property definition and any other helpful information about the property.

7. (Optional) Enter a value in the **Display order** text box.

   The number that you enter controls how the property name appears on the request form. The following ordering rules apply:

   - The display order applies only to properties that are configured with **Prompt User** or **Show in Request Form** settings.
   - All properties with a display order appear before properties with no display order.
   - Properties with a display order are sorted from lowest to highest value. Negative numbers are allowed.
   - All properties are ordered alphabetically, with all display order properties appearing before non-display order properties.
   - If two properties have the same display order value, they are sorted alphabetically.

8 Select a property definition data type from the **Data type** drop-down menu.

Table 8-47. Data type

Data type	Display as
Boolean	Allows for a boolean value. The Display as options are **Checkbox** and **Yes/No**.
Datetime	Allows for a value entered in a date and time format. The Display as option is **Date Time Picker**.
Decimal	Allows for an integer or decimal value. The Display as options are **Dropdown**, **Slider**, and **Textbox**.
Integer	Allows for an integer value. The Display as options are **Dropdown**, **Slider**, and **Textbox**.
Secure String	Allows for secure or encrypted content such as a password. The Display as option is either a password that requires confirmation or **Textbox**.
String	Allows for a string value. The Display as options are **Dropdown**, **Email**, **Hyperlink**, **Textarea**, and **Textbox**.

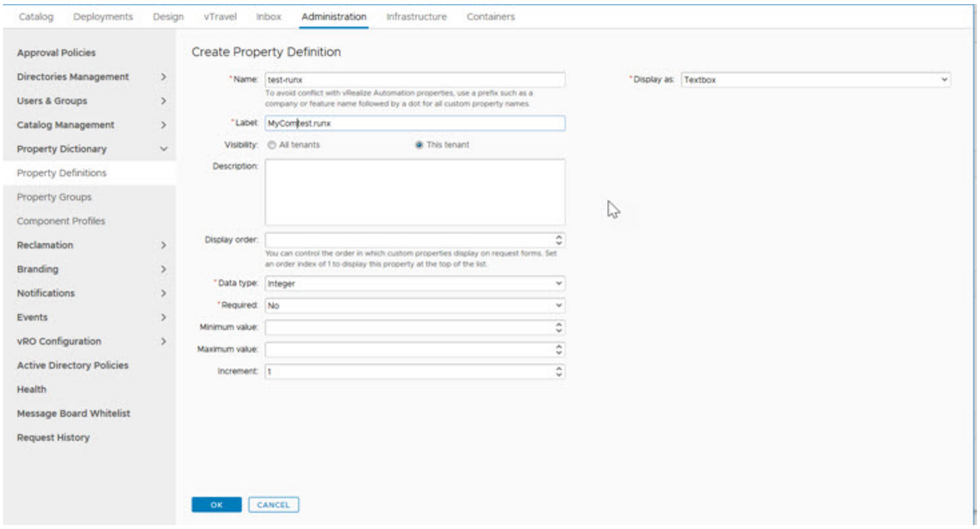
9 If the **Required** option is available, select **Yes** or **No** from the drop-down menu to specify if a value must be supplied for this property.

10 Select a display control type for this property in the **Display as** drop-down menu. Available options are derived from your **Data type** selection as shown in the above table.

Table 8-48. Display as

Display as	Description
CheckBox	Provides a single check box control.
Date Time Picker	Provides a date and time control that adheres to a *YYYY-MM-DD* or *MM/DD/YYYY* format and a time in *HH:MM* format, 24-hour clock or followed by AM or PM.
Dropdown	Provides a drop-down menu control.
Email	Provides an email control.
Hyperlink	Displays a link with the property display name as the link text and the property value as the URL.
Slider	Provides a slider control for a range of values.
Testarea	Provides a text area in which to enter or display information.
Textbox	Provides a text box in which to enter a value.
Yes/No	Specifies a Yes or No value.

11 Based on your data type, make any additional selections as presented in the user interface. A sample page is shown below:



Create Property Definition

*Name: test-runx

To avoid conflict with vRealize Automation properties, use a prefix such as a company or feature name followed by a dot for all custom property names.

*Label: MyComtest.runx

Visibility: ○ All tenants ● This tenant

Description:

Display order:

You can control the order in which custom properties display on request forms. Set an order index of 1 to display this property at the top of the list.

*Data type: Integer

*Required: No

Minimum value:

Maximum value:

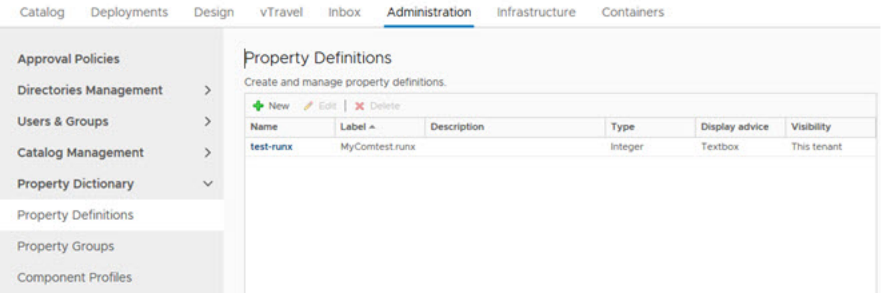
Increment: 1

*Display as: Textbox

12 Click **OK**.

Results

The property is created and available on the Property Definitions page.



Property Definitions

Create and manage property definitions.

Name	Label ▲	Description	Type	Display advice	Visibility
test-runx	MyComtest.runx		Integer	Textbox	This tenant

Create a Custom Property That Validates Against a Regular Expression

You create custom property definition that evaluates a regular expression when you want service catalog users to provide validated data on the catalog request form.

For example, to add an alphanumeric text box where the requesting user provides an application or function name that is limited to five to ten characters with no special characters. For this scenario, you use a regular expression custom property configured to something similar to `^[a-zA-Z0-9]{5,10}$`.

Prerequisites

- Ensure that you have a regular expression that validates the provided values as expected.
- Log in to vRealize Automation as a **tenant administrator** or **fabric administrator**.

Procedure

1 Select **Administration > Property Dictionary > Property Definitions**.

2 Click the **New** icon (✚).

3 Enter the options.

Option	Description
Name	Enter a value using a standard naming convention for the new property name such as `my_grouping_prefix.my_property_name`.
Label	The label is populated based on the name. You can change the label to provide a more readable name.
Visibility	The action custom properties are only available in the current tenant. To make them available in another tenant, you must configure them when you are logged in to that tenant.
Description	Describe the intent of the property definition and any other helpful information about the property.
Display order	The number that you enter controls how the property name appears on the request form. The following ordering rules apply: ■ The display order applies only to properties that are configured with **Prompt User** or **Show in Request Form** settings. ■ All properties with a display order appear before properties with no order index. ■ Properties with a display order are sorted from lowest to highest order index value. You can use negative numbers. ■ All properties are ordered alphabetically, with all display ordered properties appearing before non-display ordered properties. ■ If two properties have the same display order value, then they are sorted alphabetically.
Data type	Select **String** in the drop-down menu.
Required	Select **No** in the drop-down menu.
Display as	Select **Textbox** in the drop-down menu.
Valid user input	Enter the regular expression.

4 Enter a value in the testing text box to verify that the expression works.

5 Click **OK**.

Results

The custom property definition is added to the list and it is available to add to a blueprint.

What to do next

Add the custom property to a machine blueprint. See Add a Custom Property or Property Group Using the Properties Tab on a Blueprint's Machine Component.

Create a vRealize Orchestrator Action Custom Property Definition

You create a custom property definition that includes a vRealize Orchestrator action so that you can add the custom property to a blueprint. The action runs when the service catalog user is configuring the custom property in the request form. The action retrieves the data that is displayed in the form.

Prerequisites

- Review the configuration details for the custom property you are creating. See Configuration Details for the vRealize Orchestrator Action Custom Property Definitions.
- Log in to vRealize Automation as a **tenant administrator** or **fabric administrator**.

Procedure

1. Select **Administration > Property Dictionary > Property Definitions**.
2. Click the **New** icon (✚).
3. Enter the options.

Option	Description
Name	Consult the configuration details. Some of the custom properties require specific names or formats. Where you can, use a standard naming convention for the new property name such as `my_grouping_prefix.my_property_name`.
Label	The label is populated based on the name. You can change the label to provide a more readable name.
Visibility	The action custom properties are only available in the current tenant. To make them available in another tenant, you must configure them when you are logged in to that tenant.
Description	Describe the intent of the property definition and any other helpful information about the property.
Display order	The number that you enter controls where the property name appears on the request form. The following ordering rules apply: - The display order applies only to properties that are configured with **Prompt User** or **Show in Request Form** settings. - All properties with a display order appear before properties with no order index. - Properties with a display order are sorted from lowest to highest value. You can use negative numbers. - All properties are ordered alphabetically, with all display order properties appearing before non-display order properties. - If two properties have the same display order value, they are sorted alphabetically.

4. Consult the configuration details to determine what you must provide for the values.

   The following values are provided in the configuration details:

   - Data type
   - Display as
   - Values

- Action folder
- Script action
- Input parameters

5 Click **OK**.

Results

The custom property definition is added to the list and it is available to add to a blueprint.

What to do next

Add the custom property to a blueprint. Whether you add it as a machine or as a network property depends on the property. See Add a Custom Property to a Blueprint.

Bind Custom Properties to Create a Parent-Child Relationship

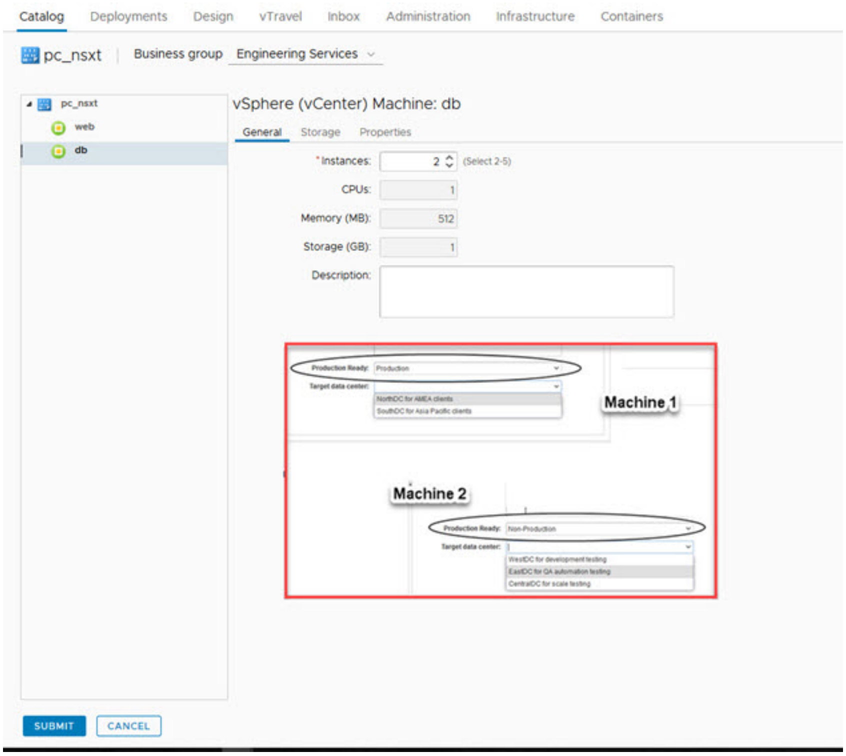
To create a parent-child relationship between custom properties, you bind the parent to the child. When you add the parent and child custom properties to a blueprint, the requesting user selects a value for the parent property. The selected parent value determines the possible values for the child property.

- The parent custom property definition can be a static list or an external value that is determined by an vRealize Orchestrator action. It provides possible input parameters to a child property definition.
- The child custom property definition must call a vRealize Orchestrator action. In the child custom property, you bind the parent custom property so that it provides an input parameter value.

For example, your development team works on production and non-production systems. You also have five data centers. Three of the data centers are your development testing data centers and the other two are where you provide services to your internal clients. To ensure that developers can deploy the same blueprint to either environment, the testing or the internal clients data centers, you create and bind two custom property definition. Using the first custom property, the requesting user can select either the production or non-production environment. Based the environment that the user selects in the request form, the second custom property displays one the following values:

- The list of three testing data centers for the non-production environments.
- The two internal clients data centers as production environments.

The following screen illustrates the catalog request page for the Machine 1 (db), with a snippet section illustrating the property to bind from the Machine 1 (db) to the property in the Machine 2 (web).



Business group: Engineering Services

vSphere (vCenter) Machine: db

*Instances: 2 (Select 2-5)

CPUs: 1

Memory (MB): 512

Storage (GB): 1

Description:

Production Ready: Production

Target data center:

- NorthDC for AMEA clients
- SouthDC for Asia Pacific clients

Production Ready: Non-Production

Target data center:

- WestDC for development testing
- EastDC for QA automation testing
- CentralDC for scale testing

The goal of this procedure is to create two custom properties that you bind in parent-child relationship. With the binding, you can select the appropriate location based on the selected production state.

Prerequisites

- For this example, create a vRealize Orchestrator action that provides data center names as location information. Name the action datacenters_prod, add an input parameter named prod as a string type, and use this sample script for the action script.

  ```
  if(prod == null) {
      return ['Empty1', 'Empty2'];
  } else if (prod.equals('nonprod')) {
      return ['WestDC for development testing', 'EastDC for QA automation testing', 'CentralDC for 
  scale testing'];
  } else {
      return ['NorthDC for AMEA clients', 'SouthDC for Asia Pacific clients'];
  }
  ```

  For information about developing workflows, and about creating and using vRealize Orchestrator script actions, see *Developing with VMware vRealize Orchestrator* in vRealize Orchestrator product documentation.

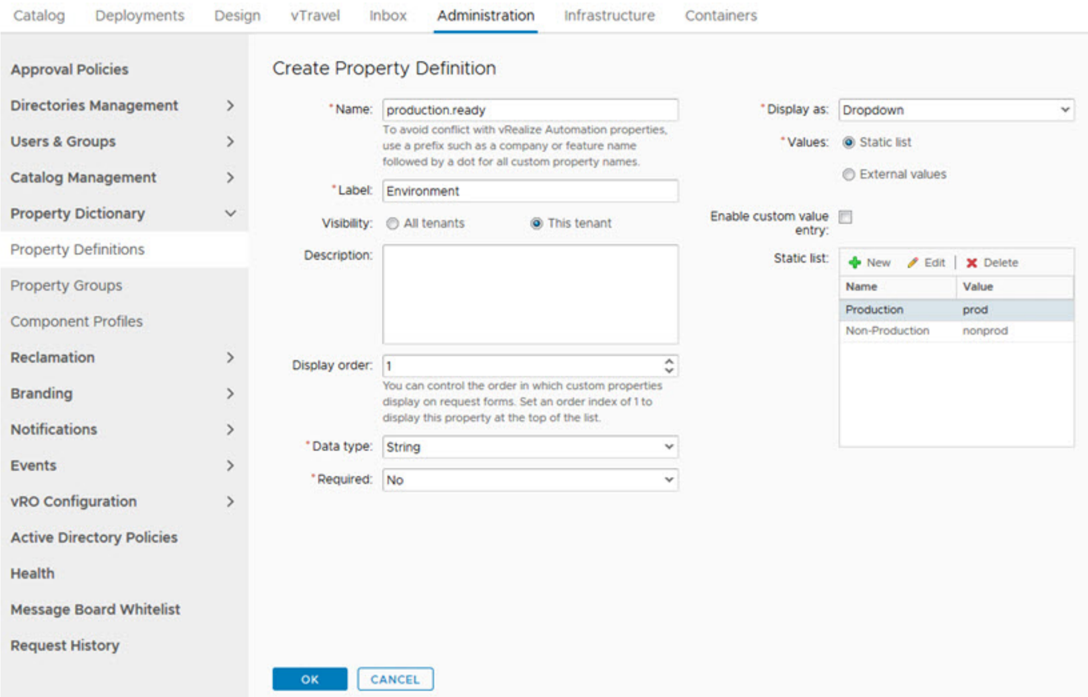
- Log in to vRealize Automation as a **tenant administrator** or **fabric administrator**.

Procedure

1 Create a custom property definition so that users can select production or non-production environment.

a Select **Administration > Property Dictionary > Property Definitions**.

b Configure the production.ready custom property.



Create Property Definition

*Name: production.ready

To avoid conflict with vRealize Automation properties, use a prefix such as a company or feature name followed by a dot for all custom property names.

*Label: Environment

Visibility: ○ All tenants ● This tenant

Description:

Display order: 1

You can control the order in which custom properties display on request forms. Set an order index of 1 to display this property at the top of the list.

*Data type: String

*Required: No

*Display as: Dropdown

*Values: ● Static list ○ External values

Enable custom value entry: □

Static list:

Name	Value
Production	prod
Non-Production	nonprod

Option	Example Values
Name	`production.ready`
Label	`Environment`
Description	`Select the production or non-production environment.`
Display order	`1` You select 1 to ensure that this custom property appears first in the blueprint.
Data type	String
Display as	Dropdown
Values	Static list
Static list values	Add the following key-pair pairs. ■ `Production` and `prod` ■ `Non-Production` and `nonprod`

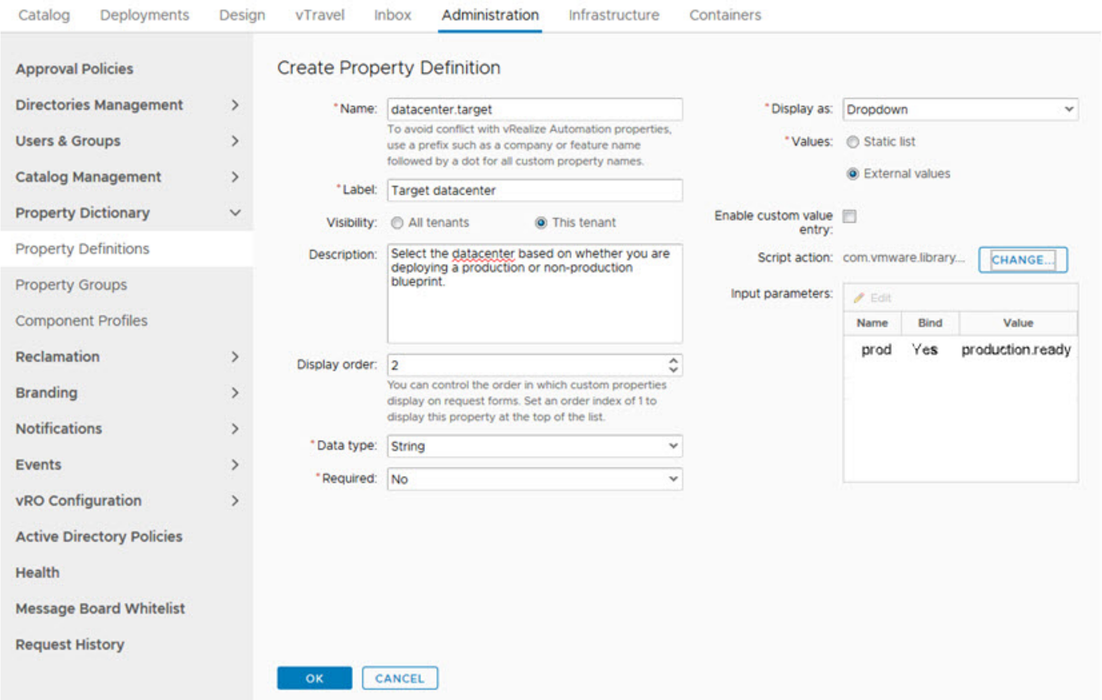
c Click **OK**.

The `production.ready` custom property is configured and ready to use.

2 Create a vRealize Orchestrator action custom property definition that runs your custom location action.

a Select **Administration > Property Dictionary > Property Definitions**.

b Configure the datacenter.target custom property.



Create Property Definition

*Name: datacenter.target

To avoid conflict with vRealize Automation properties, use a prefix such as a company or feature name followed by a dot for all custom property names.

*Label: Target datacenter

Visibility: ○ All tenants ● This tenant

Description: Select the datacenter based on whether you are deploying a production or non-production blueprint.

Display order: 2

You can control the order in which custom properties display on request forms. Set an order index of 1 to display this property at the top of the list.

*Data type: String

*Required: No

*Display as: Dropdown

*Values: ○ Static list ● External values

Enable custom value entry: □

Script action: com.vmware.library...

Input parameters:

Name	Bind	Value
prod	Yes	production.ready

Option	Example Values
Name	`datacenter.target`
Label	`Target data center`
Description	`Select the datacenter based on whether you are deploying a production or non-production blueprint.`
Display order	2 You select 2 to ensure that this custom property is listed after the `production.ready` custom property in the blueprint.
Data type	String
Display as	Dropdown
Values	External values
Script action	Click **Select** and locate your datacenters_prod action.

The input parameters table includes a prod parameter.

c In the Input parameters table, select the prod row and click **Edit**.

d Select the **Bind** check box.

e Select **production.ready** in the drop-down menu.

f Click **OK**.

g Click **OK**.

The `datacenter.target` custom property is configured and ready to use.

What to do next

- Because of the relationship between the two property definitions, add the two property definitions to a property group. See Create a Property Group.
- Add your production-datacenter property group to a blueprint. See Add a Custom Property or Property Group Using the Properties Tab on a Blueprint's Machine Component.

Add a Custom Property to a Blueprint

You can add custom properties to many parts of vRealize Automation, including approval policies, business groups, endpoints, and reservation policies. However, only the machine blueprints support the display options that you configure as property definitions. Adding a custom property to a blueprint as a simple way to verify that the custom property appears in the user interface as you designed it in the property definition.

Some custom properties are associated with the virtual machine blueprint on the **Properties** tab and some are on the **Network** tab.

- Add a Custom Property or Property Group Using the Properties Tab on a Blueprint's Machine Component

  You add a custom property as a machine custom property so that service catalog users can select of configure the values when they request the item. You can add individual properties or property groups.

- Add a Custom Property Using the Network Tab on a Blueprint's Machine Component

  Add a custom property as a network custom property so that service catalog users can select the necessary network profile value when they deploy the blueprint.

Add a Custom Property or Property Group Using the Properties Tab on a Blueprint's Machine Component

You add a custom property as a machine custom property so that service catalog users can select of configure the values when they request the item. You can add individual properties or property groups.

In this workflow, you add the custom properties to validate that they are working as expected in blueprints. You can also add custom properties to business groups, approval policies, and other components.

Prerequisites

- Verify that you created the required property definition. See Create Custom Property Definitions.

- If you are adding a property group, verify that you added the relevant property definitions to a property group. See Create a Property Group. To test the visual functions of the property definitions, you must select **Show in request** when you add the property to the group.
- If you are adding a vRealize Orchestrator action as a custom property, review the configuration details to ensure that you addrf the custom property in the correct location. See Configuration Details for the vRealize Orchestrator Action Custom Property Definitions.
- Verify that you created the blueprint to which you are adding the custom property. See Configure a Machine Blueprint.
- Log in to vRealize Automation as an **infrastructure architect**.

Procedure

1 Select **Design > Blueprints**.

2 Select the blueprint to which you are adding the custom property and click **Edit**.

3 Click the target machine component.

The configuration options for the virtual machine appear on the canvas.

4 Click the **Properties** tab, and then click the **Custom Properties** tab or the **Property Groups** tab.

- To add a custom property, click **New** and select the property definition in the drop-down menu.

Option	Description
Name	Name of the selected custom property definition.
Value	(Optional) Enter a default value.
Encrypted	When adding custom properties that run vRealize Orchestrator actions, do not encrypt the value.
Overridable	Select this option to ensure that the requesting user can select a value on the request form.
Show in request	Select this option to ensure that the requesting user can see the property and select a value on the request form.

- To add a property group, click **Add** and select the group.

5 Click **OK**.

The custom property is added to the blueprint.

6 Click **Finish**.

7 Publish the finished blueprint.

Results

The blueprint includes the custom property.

What to do next

Test the custom property in the request form. See Verify the Custom Property in the Catalog Request Form.

Add a Custom Property Using the Network Tab on a Blueprint's Machine Component

Add a custom property as a network custom property so that service catalog users can select the necessary network profile value when they deploy the blueprint.

Prerequisites

- Verify that you have the required custom property definition. See Create a vRealize Orchestrator Action Custom Property Definition.
- If you are adding a vRealize Orchestrator action as a custom property, review the configuration details to ensure that you added the custom property in the correct location. See Configuration Details for the vRealize Orchestrator Action Custom Property Definitions.
- Verify that you created the blueprint to which you are adding the custom property. See Configure a Machine Blueprint.
- Log in to vRealize Automation as an **infrastructure architect**.

Procedure

1 Select **Design > Blueprints**.

2 Select the blueprint that you want to edit.

The blueprint opens in the design canvas.

3 In the design canvas, click the virtual machine component that you want to edit.

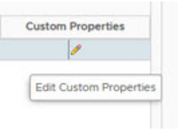
The configuration options for the virtual machine appear on the canvas.

4 Click the machine component's **Network** tab.

5 Click **New** to add a new network row.

6 In the new row, select a network and an assignment type (static IP or DHCP), specify an address if using a static IP, and click **OK**.

7 In the new row, click the Edit icon in the Custom Properties column to assign a custom property.



8 Click **New**, select the custom property, configure the options described in the following table, and click **OK**.

Option	Description
Name	Select an existing custom property name from the drop-down menu.
Value	(Optional) Enter a default value.
Encrypted	When adding custom properties that run vRealize Orchestrator actions, do not encrypt the value.
Overridable	Select this option to ensure that the requesting user can select a value on the request form.
Show in request	Select this option to ensure that the requesting user can see the property and select a value on the request form.

The network, with its configured custom property, is added to the blueprint.

9 Click **Finish**.

10 Publish the finished blueprint.

Results

The blueprint includes the custom property.

What to do next

Test the custom property in the request form. See Verify the Custom Property in the Catalog Request Form

Verify the Custom Property in the Catalog Request Form

As creator of the custom property definitions that run vRealize Orchestrator actions, you should test your custom properties to ensure that the correct values appear in the request form.

Prerequisites

- Add the custom property to the appropriate location in the blueprint. See Add a Custom Property to a Blueprint.
- Verity that the blueprint is entitled to you so that you can test the custom properties in the blueprint. See Entitle Users to Services, Catalog Items, and Actions.
- Log in to the vRealize Automation as a user with access to the test blueprint.

Procedure

1 Click **Catalog** to display the catalog items that you are entitled to use.

Published blueprints appear on the Catalog page as catalog items.

2 Click **Request** on the catalog item.

3 On the request form, click the machine to which you added the custom property.

4 On the machine's **Properties** tab, select the custom property and click the drop-down arrow.

The vRealize Orchestrator action runs and retrieves the values it is configured to display. Verify that the expected values appear.

What to do next

Add the custom property to your production blueprints where needed.

Configuration Details for the vRealize Orchestrator Action Custom Property Definitions

You create custom property definitions that run vRealize Orchestrator actions to retrieve key value pairs from external files or from vRealize Automation configuration information. You add the custom properties to blueprints so that they appear in the catalog request forms.

The service catalog user requesting the item can select a value to include in the deployment. When the user clicks the drop-down menu to select a value, the vRealize Orchestrator action runs, retrieving the data that is displayed in the menu for the user to select.

The configuration workflows for each vRealize Orchestrator actions property definition are similar, but some of the details vary. For example, there are differences in prerequisites and limitations, and where you apply the custom property in the blueprint might vary.

- Network Custom Property Definition

  You add a custom property to retrieve network names from the vRealize Automation database when you want users to select the network in the request form. The network selector custom property uses a vRealize Orchestrator action to retrieve the values.

- Reservation Policy Custom Property Definition

  You add a custom property definition to retrieve reservation policy names that are applicable to the requesting users when they select the policy in the request form. The reservation policy selector custom property definition uses a vRealize Orchestrator action to retrieve the values.

- PowerShell Script Custom Property Definition

  You add a custom property to run a PowerShell script when you want to use a script to retrieve data to populate the custom property in the request form. The PowerShell script custom property uses a vRealize Orchestrator action to run the script and retrieve the values.

- Database Query Custom Property Definition

  You add a custom property to query a database when you want to retrieve values from that database to populate the custom property on the request form. The database custom property uses a vRealize Orchestrator action to run the query and retrieve the values.

- Custom Action Custom Property Definition

  You add a custom property to retrieve data from a source using a custom vRealize Orchestrator action when you want users to select the retrieved values in the request form.

Network Custom Property Definition

You add a custom property to retrieve network names from the vRealize Automation database when you want users to select the network in the request form. The network selector custom property uses a vRealize Orchestrator action to retrieve the values.

Limitations

Plan for the following limitations when you use the network selector custom property.

- The name of the custom property must be `VirtualMachine.Network0.Name`. This name is required. You cannot create multiple property definitions for the network selector.
- The action retrieves all the network names for the requesting user without validating that it applies to the target vCenter Server instance. A service catalog user might select a network that is not applicable to the selected target. If the wrong network is selected, the catalog request fails.
- The action retrieves network names for the requesting user only. If you submit a request on behalf of other users, the networks are for you. For example, Network A and Network C are associated with Business Group 1, so the BG 1 users see only Network A and C, not B.

Prerequisites

If you use an external vRealize Orchestrator server, verity that it is set up correctly. See Configure an External vRealize Orchestrator Server.

Custom Property Configuration Values

You use these options to create the custom property.

Table 8-49. Network Custom Property Configuration Values

Option	Value
Name	You must use `VirtualMachine.Network0.Name`. See Custom Properties V.
Data type	String
Display as	Dropdown
Values	External
Action folder	com.vmware.vra.networks
Script action	getApplicableNetworks This script action is an example script. You can create specific actions for your environment.
Input parameters	No required parameters.

Blueprint Configuration

Add the custom property on the blueprint **Network** tab. See Add a Custom Property Using the Network Tab on a Blueprint's Machine Component.

Reservation Policy Custom Property Definition

You add a custom property definition to retrieve reservation policy names that are applicable to the requesting users when they select the policy in the request form. The reservation policy selector custom property definition uses a vRealize Orchestrator action to retrieve the values.

Limitations

Plan for the following limitations when you use the reservation policy selector custom property.

- The name of the custom property must be ReservationPolicyID. This name is required. You cannot create multiple property definitions for the reservation policy selector.
- The action retrieves all reservation policies applicable to the requesting user without validating that it applies to the target endpoint, for example a vCenter Server instance or some other platform. A service catalog user might select a reservation that is not applicable to the selected blueprint target system. If the user selects the wrong reservation, the catalog request fails.
- The action retrieves reservation policies for the requesting user only. If you submit a request on behalf of another user, the reservation policies are for you. For example, Reservation 1 and Reservation 3 are associated with Business Group 1, so the BG 1 users see only Reservations 1 and 3, not 2.

Prerequisites

If you use an external vRealize Orchestrator server, verity that it is set up correctly. See Configure an External vRealize Orchestrator Server.

Custom Property Configuration Values

You use these options to create the custom property.

Table 8-50. Reservation Policy Custom Property Configuration Values

Option	Value
Name	You must use ReservationPolicyID.
Data type	String
Display as	Dropdown
Values	External
Action folder	com.vmware.vra.reservations
Script action	getApplicableReservationPolicies This script action is an example script. You can create specific actions for your environment.
Input parameters	No required parameters.

Blueprint Configuration

You can add a custom property to the blueprint **Properties** tab to associate the property with the overall blueprint.

PowerShell Script Custom Property Definition

You add a custom property to run a PowerShell script when you want to use a script to retrieve data to populate the custom property in the request form. The PowerShell script custom property uses a vRealize Orchestrator action to run the script and retrieve the values.

For example, as the cloud administrator you have a PowerShell script that retrieves user IDs from the Active Directory that is registered with vRealize Automation. The intent of the script is to retrieve and display John Smith when the actual value in Active Directory is JSmith01.

An advantage to using the PowerShell script action includes a central location for the script. You can either store the script on a central server and then run it on target virtual machines, or you can store it in vRealize Orchestrator and then run it on the target machines. A central location decreases maintenance time. Storing the scripts in vRealize Orchestrator when you have backup and restore configured ensures that you can restore the scripts if a system failure occurs.

Prerequisites

Verify that you have a working PowerShell script that returns key pair values. The script must be available on an accessible server or that the script is uploaded into vRealize Orchestrator.

Custom Property Configuration Values

You use these options to create the custom property.

Table 8-51. PowerShell Script Custom Property Configuration Values

Option	Value
Name	You can use any string.
Data type	String
Display as	Dropdown
Values	External
Action folder	com.vmware.vra.powershell

Table 8-51. PowerShell Script Custom Property Configuration Values (continued)

Option	Value
Script action	Select an action based on where the PowerShell script is located. ■ If the PowerShell script is on a central server, use executeExternalPowerShellScriptOnHostByName. ■ If the PowerShell script is uploaded into vRealize Orchestrator, use executePowershellScriptFromResourceOnHostByName. These script actions are example scripts. You can create specific actions for your environment. The `Resources/Sample/vRA/PowerShell/countries.ps1` sample PowerShell script is provided in the vRealize Orchestrator client as reference for use with the executePowershellScriptFromResourceOnHostByName action.
Input parameters	Configure the input parameters based on the selected action. Define parameters ■ If you use executeExternalPowerShellScriptOnHostByName: ■ **hostName**. Name of the central server where the script is located. ■ **externalPowershellScript**. Path to the PowerShell file on the host. ■ **Arguments**. Parameters to pass to the script. You separate the arguments with commas. For example, Argument1,Arguement2. ■ If you use executePowershellScriptFromResourceOnHostByName: ■ vRealize Orchestrator. Name of the vRealize Orchestrator instance you are using as the host. ■ **scriptResourcePath**. Path to the PowerShell file on the host. ■ **scriptResourceName**. Path to the PowerShell file as an uploaded resource in vRealize Orchestrator.

Blueprint Configuration

You can add a custom property to the blueprint **Properties** tab to associate the property with the overall blueprint.

Database Query Custom Property Definition

You add a custom property to query a database when you want to retrieve values from that database to populate the custom property on the request form. The database custom property uses a vRealize Orchestrator action to run the query and retrieve the values.

The action is supported for the following databases:

- Microsoft SQL Server
- MySQL
- Oracle
- PostgreSQL

Limitations

All retrieved values are converted to strings.

Prerequisites

Verify that the vRealize Orchestrator SQL Plug-In is installed and configured to connect to the target database.

Custom Property Configuration Values

You use these options to create the custom property.

Table 8-52. Database Query Custom Property Configuration Values

Option	Value
Name	You can use any string.
Data type	String
Display as	Dropdown
Values	External
Action folder	com.vmware.vra.sql
Script action	executeSQLSelectOnDatabase This script action is an example script. You can create specific actions for your environment.
Input parameters	■ **databaseName**. Name of the database to which vRealize Orchestrator is connected. ■ **sqlSelectQuery**. The SQL select query that you are running on the database to retrieve the values. For example, select * <table name>. ■ **keyColumnName**. Name of the database column that is the key for the key pair value. ■ **valueColumnName**. Name of the database column from which you are retrieving values.

Blueprint Configuration

You can add a custom property to the blueprint **Properties** tab to associate the property with the overall blueprint.

Custom Action Custom Property Definition

You add a custom property to retrieve data from a source using a custom vRealize Orchestrator action when you want users to select the retrieved values in the request form.

Limitations

The supported scripted actions include:

- Any and Array/Any
- Array/String and Array/Properties if you select the String data type in the definition form
- Array/Number if you select the Integer or Decimal data type in the definition form

Prerequisites

Verify that you have a working vRealize Orchestrator action. For information about developing workflows and creating and using vRealize Orchestrator script actions, see *Developing with VMware vCenter Orchestrator* .

The action script must accept the input parameter values. You can configure the values as key value pairs. You can present user-readable names for less friendly identifiers using key value pairs.

Custom Property Configuration Values

You use these options to create the custom property.

Table 8-53. Custom Action Custom Property Configuration Values

Option	Value
Name	You can use any string.
Data type	Decimal, Integer, or String
Display as	Dropdown
Values	External
Action folder	Location of your custom action.
Script action	Name of your custom action.
Input parameters	Depends on your custom action.

Blueprint Configuration

Usually you add the custom property on the blueprint Properties tab. Whether you add it to the Properties tab depends on your action. See Add a Custom Property to a Blueprint.

Using Property Groups

You can create property groups to collect properties into a single unit.

Property groups are logical and reusable groups of properties, that can include property definitions that you create or custom properties that are supplied. Property groups are designed to simplify the process of adding properties to blueprints or other vRealize Automation elements for which they are available. They provide a means by which logical groupings of properties can be added more efficiently than by adding the properties individually.

A property group typically contains properties that are commonly used together. For example, you can create a property group named WimImagingProperties that contains properties commonly used for WIM-based provisioning:

- `Image.ISO.Location`
- `Image.ISO.Name`
- `Image.Network.Password`

- `Image.Network.User`
- `Image.WIM.Index`
- `Image.WIM.Name`
- `Image.WIM.Path`

You can also create a property group for vCloud Air or vCloud Director machine provisioning that contains the following properties:

- `VirtualMachine.Network0.Name`
- `VCloud.Template.MakeIdenticalCopy`
- `VMware.SCSI.Type`
- `Sysprep.Identification.DomainAdmin`
- `Sysprep.Identification.DomainAdminPassword`
- `Sysprep.Identification.JoinDomain`

You can use the `vra content list --type property-definition` vRealize CloudClient command to list all property definitions in the current vRealize Automation tenant. You can also use the `vra content list --type property-group` vRealize CloudClient command to list all property groups.

Create a Property Group

You can organize specific custom properties into property groups to more easily add multiple custom properties to blueprints.

Prerequisites

Log in to vRealize Automation as a **tenant administrator** or **fabric administrator**.

Procedure

1. Select **Administration > Property Dictionary > Property Groups**.
2. Click **New** (➕).
3. Enter the new property group name and ID.

   If you enter the **Name** value first, the **ID** text box is populated with the same value.
4. In the **Visibility** section, select **All tenants** or **This tenant** to determine where the property is to be available.

   If you are logged in with only tenant administrator privileges, then only **This tenant** is available. If you are logged in with only fabric administrator privileges, then only **All tenants** is available.

   You cannot change the **All tenants** or **This tenant** setting after you create the item.
5. (Optional) Enter a description of the property group, for example `My Cloning Properties vSphere`.

6 Click **New** and add a property to the group.

Option	Description
Name	Add a new property or select an existing property from the drop-down menu. For example, enter `VirtualMachine.Storage.ReserveMemory`.
Value	(Optional) Enter a default property value. For example, enter `True`.
Encrypted	Select this option to specify that the property value be encrypted. For example, if the value is to be a password or other secure entry, using the encrypted option hides the value characters. When adding custom properties that run vRealize Orchestrator actions, do not encrypt the value.
Show in request	Select this option to specify that the requesting user can see the property and select a value on the request form when requesting machine provisioning.

7 Click **OK** to add the property to the group.

8 Add additional properties to the group.

9 Click **OK**.

Defining Component Profile Settings

You can use component profiles to configure advanced property management capabilities in vRealize Automation blueprints. Deployers can then use the `Size` and `Image` component profiles on a blueprint to select pre-defined value sets.

You can use the `Size` and `Image` component profiles, and their specified value sets, to map to a logical grouping such as Small, Medium, and Large or Dev, Test, and Production. By using these settings, you can reduce the number of blueprints that you need to maintain.

A component profile defines settings for a vSphere machine component in a blueprint. For example, you might define a component profile for a small size virtual machine deployment. You might define another component profile for a large size machine deployment. You can use vRealize Automation to define the following component profile types:

- Size

  See Configure Component Profile Size Settings for Catalog Deployments.

- Image

  See Configure Component Profile Image Settings for Catalog Deployments.

  For related information about using component profiles in a blueprint, see Understanding and Using Blueprint Parameterization.

You can define multiple named value sets within the `Size` and `Image` component profile types and add one or more of the value sets to machine components in a blueprint. Each value set that you define for the component profile type contains the following configurable settings:

- Name that requesters see when they provision a machine
- Unique identifier for tenant
- Description
- Set of value choices for each option in the value set

You cannot define other component profile types.

When you request provisioning, you can select from available `Size` and `Image` options. When you choose one of the value sets, the corresponding property values are bound to the request.

Configure Component Profile Image Settings for Catalog Deployments

You can configure the component profile `Image` setting to control build information for vSphere machine components in the blueprint.

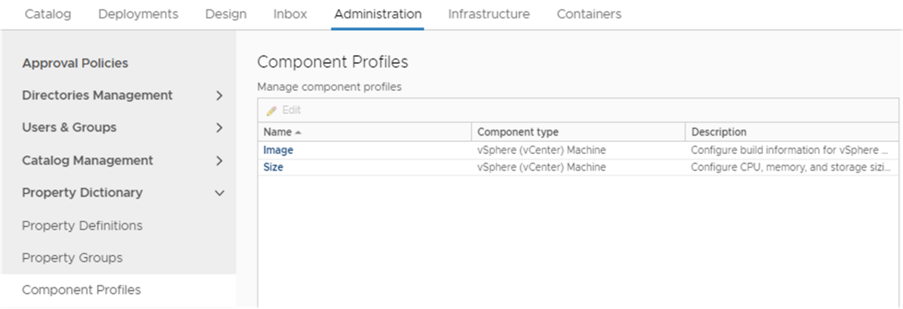
After you define value sets for the `Image` component profile, you can add one or more value sets to the component profile for a vSphere machine component in a blueprint. Users can then select an `Image` value set when they request a catalog item.

Prerequisites

Log in to vRealize Automation as an administrator with **tenant administrator** and **IaaS administrator** access rights.

Procedure

1 Select **Administration > Property Dictionary > Component Profiles**.



Component Profiles

Manage component profiles

Name	Component type	Description
Image	vSphere (vCenter) Machine	Configure build information for vSphere ...
Size	vSphere (vCenter) Machine	Configure CPU, memory, and storage sizi...

2 Click **Image** in the Name column.

Information about the supplied image component property is displayed.

3 Click the **Value Sets** tab.

4 To define a new value set click **New** and configure the `Image` settings.

   a Enter a value in the **Display name** field to append to the ValueSet delimeter, for example `CloneA`.

   b Accept the default value shown in the **Name** text box, for example **ValueSet.CloneA**, or enter a custom name.

   c Enter a description such as `Build settings for cloning scenario A` in the **Description** text box.

   d Select **Active** or **Inactive** in the **Status** drop down menu.

   Select **Active** to allow the value set to be visible in the catalog provisioning request form.

   e Select **Server** or **Desktop** as the blueprint type.

   f Select the build action to use for this value set, for example select **Clone**.

   Other action include:

   - **Create**
   - **Clone**
   - **Linked Clone**
   - **NetApp FlexClone**

   g Select the CloneWorkflow provisioning workflow.

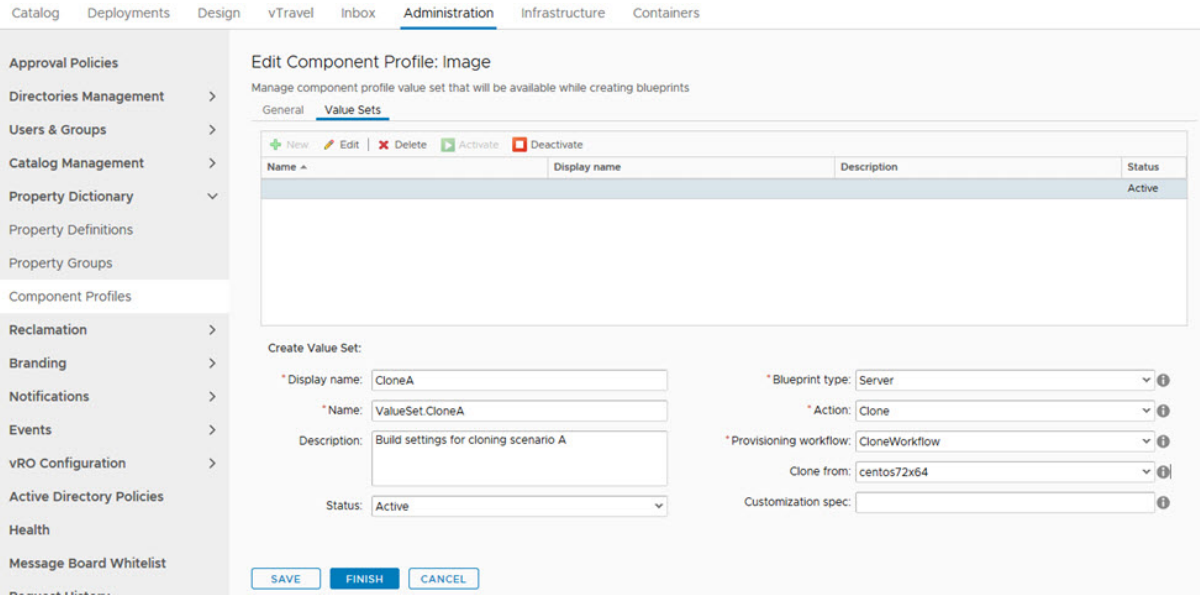
   For information about importing the image value set for an OVF, see Define an Image Value Set for a Component Profile By Using an OVF.

   h (Optional) Select a source machine to clone from, for example **centos7264**.

   i (Optional) Enter the path to a vSphere customization specification.

5 Click **Save**.

6 When you are satisfied with your settings, click **Finish**.



Edit Component Profile: Image

Manage component profile value set that will be available while creating blueprints

Name	Display name	Description	Status
			Active

Create Value Set:

- *Display name: CloneA
- *Name: ValueSet.CloneA
- Description: Build settings for cloning scenario A
- Status: Active
- *Blueprint type: Server
- *Action: Clone
- *Provisioning workflow: CloneWorkflow
- Clone from: centos72x64
- Customization spec:

What to do next

Add one or more value sets to the `Image` component profile by using the **Profiles** tab on a vSphere machine component. See Configure a Machine Blueprint and vSphere Machine Component Settings in vRealize Automation.

Configure Component Profile Size Settings for Catalog Deployments

You can configure the component profile `Size` setting to specify CPU, memory, and storage sizing for vSphere machine components in the blueprint.

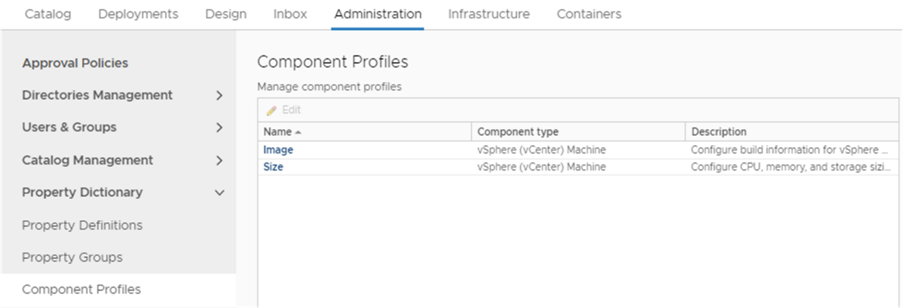
After you define value sets for the `Size` component profile, you can add one or more value sets to a component profile for a vSphere machine component in a blueprint. Users can then select a `Size` value set when they request a catalog item.

Prerequisites

Log in to vRealize Automation as an administrator with **tenant administrator** and **IaaS administrator** access rights.

Procedure

1 Select **Administration > Property Dictionary > Component Profiles**.



Component Profiles

Manage component profiles

Name	Component type	Description
Image	vSphere (vCenter) Machine	Configure build information for vSphere ...
Size	vSphere (vCenter) Machine	Configure CPU, memory, and storage sizi...

2 Click **Size** in the Name column.

Information about the supplied `Size` component profile is displayed on the **General** tab.

3 Click the **Value Sets** tab.

4 To define a new value set, for example, for a large size deployment, click **New** and configure the `Size` settings.

a Enter a value in the **Display name** field to append to the ValueSet delimeter, for example **`small_1`**.

b Accept the default value shown in the **Name** text box as **ValueSet.small_1** or enter a custom name.

c Enter a description such as **`small deployment`** in the **Description** text box.

d Select **Active** or **Inactive** in the Status drop down menu.

Select **Active** to allow the value set to be visible in the catalog provisioning request form.

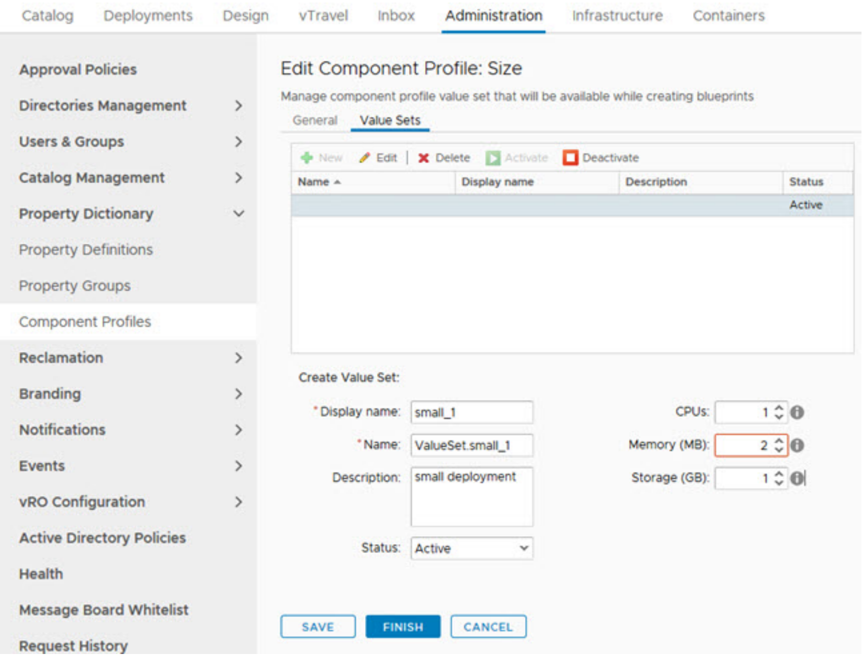
e Enter the number of virtual CPUs on which the deployment can be run, for example 1.

f Enter the amount of RAM to be used by virtual machines in the deployment, for example 2 MB.

g Enter the amount of storage to be used by virtual machines in the deployment, for example 1 GB.

5 Click **Save**.

6 When you are satisfied with your settings , click **Finish**.



Edit Component Profile: Size

Manage component profile value set that will be available while creating blueprints

Name	Display name	Description	Status
			Active

Create Value Set:

* Display name: small_1
* Name: ValueSet.small_1
Description: small deployment
Status: Active
CPUs: 1
Memory (MB): 2
Storage (GB): 1

What to do next

Add one or more value sets to the Size component profile by using the **Profiles** tab on a vSphere machine component. See Configure a Machine Blueprint and vSphere Machine Component Settings in vRealize Automation.

Integrating Third-Party Server Automation Tools

9

You can leverage your existing third-party server automation tools to deploy software on provisioned machines.

This chapter includes the following topics:

IaaS Integration for BMC BladeLogic

IaaS Integration for BMC BladeLogic provides information about integrating BMC BladeLogic Configuration Manager with VMware vRealize ™ Automation.

This documentation provides information on how you can enable deployment of BMC BladeLogic software jobs on machines provisioned by vRealize Automation.

Note Not all features and capabilities of vRealize Automation are available in all editions. For a comparison of feature sets in each edition, see https://www.vmware.com/products/vrealize-automation/.

Intended Audience

This information is intended for system administrators, tenant administrators, fabric administrators, and business group managers of vRealize Automation. This content is written for experienced Windows or Linux system administrators who are familiar with virtualization technology and the basic concepts described in *Foundations and Concepts*.

BMC BladeLogic Configuration Manager Overview

You can integrate BMC BladeLogic with vRealize Automation to enable deployment of BMC BladeLogic software jobs on machines provisioned by vRealize Automation. Custom properties can be used to specify whether these jobs can be selected by the requesting user on a per-machine basis or applied to all machines provisioned from a particular blueprint.

The following is a high-level overview of the requirements for integrating BMC BladeLogic Configuration Manager with vRealize Automation:

- A system administrator verifies that BMC BladeLogic Operations Manger 7.6.0.115 or BMC Server Automation Console 8.2 is installed on the same host as your external provisioning integration (EPI) agent.
- A system administrator sets the PowerShell execution policy to RemoteSigned. See Set the PowerShell Execution Policy to RemoteSigned.
- A system administrator installs at least one EPI agent. See Install an EPI Agent for BMC BladeLogic.
- A system administrator configures how software jobs are deployed. See Integrate BMC BladeLogic.
- A tenant administrator or a business group manager creates a blueprint that enables the deployment of software jobs. See Creating BMC BladeLogic Blueprints.

Set the PowerShell Execution Policy to RemoteSigned

You must set the PowerShell Execution Policy from Restricted to RemoteSigned or Unrestricted to allow local PowerShell scripts to be run.

For more information about the PowerShell Execution Policy, see the Microsoft PowerShell article about Execution Policies. If your PowerShell Execution Policy is managed at the group policy level, contact your IT support for about their restrictions on policy changes, and see the Microsoft PowerShell article about Group Policy Settings.

Prerequisites

- Verify that Microsoft PowerShell is installed on the installation host before agent installation. The version required depends on the operating system of the installation host. See Microsoft Help and Support.
- For more information about PowerShell Execution Policy, run `help about_signing` or `help Set-ExecutionPolicy` at the PowerShell command prompt.

Procedure

1. Using an administrator account, log in to the IaaS host machine where the agent is installed.
2. Select **Start > All Programs > Windows PowerShell version > Windows PowerShell**.
3. For Remote Signed, run `Set-ExecutionPolicy RemoteSigned`.
4. For Unrestricted, run `Set-ExecutionPolicy Unrestricted`.
5. Verify that the command did not produce any errors.
6. Type **`Exit`** at the PowerShell command prompt.

Install an EPI Agent for BMC BladeLogic

A system administrator must install at least one vRealize Automation EPI agent to manage interaction with BMC BladeLogic. The agent can be installed anywhere, but it must be able to communicate with vRealize Automation and BMC BladeLogic Configuration Manager.

Prerequisites

- Verify that BMC BladeLogic Operations Manger 7.6.0.115 or BMC Server Automation Console 8.2 is installed on the same host as your EPI agent.

  If the EPI agent is installed before BMC Operations Manager, the agent service must be restarted after BMC Operations Manager is installed.

- The agent must be installed on Windows Server 2008 SP1, Windows Server 2008 SP2 (32 or 64-bit), Windows Server 2008 R2 system, or Windows 2012 with .NET 4.5.
- The credentials under which the agent runs must have administrative access to all BMC BladeLogic hosts with which the agent interacts.
- Log in to the vRealize Automation console as a **system administrator**.

See *Installing vRealize Automation* for complete information about installing vRealize Automation agents.

Procedure

1. Select **Component Selection** on the Installation Type page.
2. Accept the root install location or click **Change** and select an installation path.

   Even in a distributed deployment, you might sometimes install more than one IaaS component on the same Windows server.

   If you install more than one IaaS component, always install them to the same path.

3. Click **Next**.
4. Log in with administrator privileges for the Windows services on the installation machine.

   The service must run on the same installation machine.

5. Click **Next**.
6. Select **EPIPowerShell** from the Agent type list.

7 Enter an identifier for this agent in the **Agent name** text box.

Maintain a record of the agent name, credentials, endpoint name, and platform instance for each agent. You need this information to configure endpoints and to add hosts in the future.

Important For high availability, you may add redundant agents and configure them identically. Otherwise, keep agents unique.

Option	Description
Redundant agent	Install redundant agents on different servers. Name and configure redundant agents identically.
Standalone agent	Assign a unique name to the agent.

8 Configure a connection to the IaaS Manager Service host.

Option	Description
With a load balancer	Enter the fully qualified domain name and port number of the load balancer for the Manager Service component, *mgr-svc-load-balancer.mycompany.com*:443. Do not enter IP addresses.
Without a load balancer	Enter the fully qualified domain name and port number of the machine where you installed the Manager Service component, *mgr-svc.mycompany.com*:443. Do not enter IP addresses.

The default port is 443.

9 Configure a connection to the IaaS Web server.

Option	Description
With a load balancer	Enter the fully qualified domain name and port number of the load balancer for the Web server component, *web-load-balancer.mycompany.com*:443. Do not enter IP addresses.
Without a load balancer	Enter the fully qualified domain name and port number of the machine where you installed the Web server component, *web.mycompany.com*:443. Do not enter IP addresses.

The default port is 443.

10 Click **Test** to verify connectivity to each host.

11 Select **BMC** in **EPI Type**.

12 Select the EPI type.

13 Enter the fully qualified domain name of the managed server in the **EPI Server** text box.

14 Click **Add**.

15 Click **Next**.

16 Click **Install** to begin the installation.

   After several minutes a success message appears.

17 Click **Next**.

18 Click **Finish**.

What to do next

Integrate BMC BladeLogic

Extend the Default Software Installation Timeout

When you install the software for the integration product, the software might take longer to install than the default 30-minute timeout. You can increase the default timeout to a value that allows the installation to finish.

Procedure

1. Navigate to the Manager Service installation directory. Typically, this is `%System-Drive%\Program Files x86\VMware\vCAC\Server`.
2. Create a backup of the `ManagerService.exe.config` file.
3. Open the `ManagerService.exe.config` file and locate the `workflowTimeoutConfigurationSection` element and increase the value of the `DefaultTimeout` attribute from 30 minutes to your desired limit.
4. Click **Save** and close the file.
5. Select **Start > Administrative Tools > Services**, and restart the vRealize Automation service.

Integrate BMC BladeLogic

If a system from which BMC BladeLogic Configuration Manager deploys software is available on the network and you have installed an EPI agent to interact with it, software can be deployed from it directly to newly provisioned machines. The requesting user can select which software to deploy or the blueprint can contain the specific jobs to be deployed on all machines provisioned from that blueprint.

Prerequisites

- Install an EPI Agent for BMC BladeLogic.
- Log in to the vRealize Automation EPI/BMC Agent host as a **system administrator**.
- As the **system administrator** under which the EPI agent is running, log in to the BladeLogic console to configure the authentication profile to be used and to accept any BladeLogic security certificates, and then close the console. This prerequisite is required only once.

Procedure

1. Select **Start > Administrative Tools > Services**, and stop the vRealize Automation EPI/BMC Agent service.

2. On the EPI agent installation host, which could be the same as the Manager Service host, change to the EPI agent installation directory, typically `%SystemDrive%\Program Files (x86)\VMware\vCAC Agents\agent_name`.

3. Edit every file in the `Scripts\nsh` folder in the EPI agent directory and under the parameter list section of each `.nsh` file, update the values for the following variables. The description of each variable appears above the variable definitions.

   ```
   USERNAME_USER=BLAdmin

   AUTH_TYPE=SRP

   PASSWORD_USER=password

   APP_SERVER_HOST=bladelogic.dynamicops.local

   ROLE_NAME=BLAdmins
   ```

4. Edit the agent configuration file, `VRMAgent.exe.config`, in the EPI agent installation directory and replace `CitrixProvisioningUnregister.ps1` with `DecomMachine.ps1`.

   a. Locate the following line.

   ```
   <DynamicOps.Vrm.Agent.EpiPowerShell
     registerScript="CitrixProvisioningRegister. ps1"
     unregisterScript="CitrixProvisioningUnregister.ps1"/>
   ```

   b. Change the line to match the following line.

   ```
   <DynamicOps.Vrm.Agent.EpiPowerShell
     registerScript="CitrixProvisioningRegister. ps1"
     unregisterScript="DecomMachine.ps1"/>
   ```

5. If you intend to provision by cloning with a static IP address assignment, you can enable BMC BladeLogic registration of provisioned machines by IP address rather than by machine name.

   a. Edit the files `InstallSoftware.ps1` and `DecomMachine.ps1` in the`Scripts` folder in the EPI agent directory and change the line `$byip=$false` to `$byip=$true`. edit the files `InstallSoftware.ps1` and `DecomMachine.ps1` in the `Scripts` folder in the EPI agent directory and change the line `$byip=$false` to `$byip=$true`.

   b. If you enable registration by IP address by making the above change, you must provision by using static IP address assignment, otherwise, BMC BladeLogic integration fails.

6. Select **Start > Administrative Tools > Services** to start the EPI/BMC agent service (vRealize Automation Agent – agentname service).

7. Place all the BMC BladeLogic jobs you want available to be selected by machine requestors or specified by blueprint architects under a single location within BMC BladeLogic Configuration Manager, for example, `/Utility`.

8. Prepare a reference machine and convert it to a template for cloning.

   a. Install a BMC BladeLogic agent that points to the server on which BMC BladeLogic Configuration Manager is running.

   b. Verify that you are able to connect to the agent on the guest and successfully execute jobs as expected after provisioning.

Results

Tenant administrators and business group managers can now integrate BMC BladeLogic into clone blueprints. See Add BMC BladeLogic Integration to a Blueprint.

Creating BMC BladeLogic Blueprints

BMC BladeLogic integration is invoked by adding custom properties for any BMC BladeLogic software jobs to be deployed on machines provisioned from a blueprint.

Obtain the following information so that tenant administrators and business group managers can include it in their blueprints:

- The name of the template.
- The name of the customization specification.
- The amount of total storage specified for the template.
- For vCenter Server integrations, the vCenter Server guest operating system version with which vCenter Server is to create the machine.

Add BMC BladeLogic Integration to a Blueprint

To create a blueprint that enables the deployment of BMC BladeLogic Configuration Manager software jobs on machines provisioned from it, a tenant administrator or a business group manager must create a blueprint for provisioning by cloning that includes BMC BladeLogic custom properties.

- Obtain the following information from your fabric administrator:
  - The name of the server that hosts BMC BladeLogic.
  - The name of the default authentication profile on the BMC BladeLogic server.
  - The BMC BladeLogic location of software jobs to be deployed. This must match the appropriate value of `Vrm.Software.Id`*NNNN*.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **business group manager**.

- Create a blueprint for cloning by using the template and customization specification provided to you by your fabric administrator. See *IaaS Configuration for Virtual Platforms*.

  Note A fabric administrator can create a build profile by using the property set BMCSoftWareProperties. This makes it easier for tenant administrators and business group managers to correctly include this information in their blueprints.

- For a list of all required and common BMC BladeLogic custom properties, see Custom Properties for BMC BladeLogic Configuration Manager Integration.

Procedure

1. Select **Design > Blueprints**.
2. Locate the clone blueprint that you want to integrate with BMC BladeLogic.
3. In the Actions column, click the down arrow and click **Edit**.
4. Click the **Properties** tab.
5. (Optional) Select one or more property groups.

   Property groups contain multiple custom properties.
6. (Optional) Add any custom properties to your machine component.

   a. Click **New Property**.

   b. Enter the custom property in the **Name** text box.

   c. (Optional) To encrypt the custom property in the database, select the **Encrypted** check box.

   d. Enter the value of the custom property in the **Value** text box.

   e. (Optional) To require the user to provide a value when they request a machine, select the **Prompt user** check box.

      If you choose to prompt users for a value, any value you provide for the custom property is presented to them as the default. If you do not provide a default, users cannot continue with the machine request until they provide a value for the custom property.

   f. Click the **Save** icon (✓).
7. Click **OK**.

Results

Your blueprint is saved.

What to do next

Publish your blueprint to make it available as a catalog item. See Publish a Blueprint.

Custom Properties for BMC BladeLogic Configuration Manager Integration

vRealize Automation includes custom properties that you can use to provide additional controls for BMC BladeLogic Configuration Manager integration.

Table 9-1. Custom Properties Required for BMC BladeLogic Configuration Manager Integrations

Custom Property	Description
`VirtualMachine.EPI.Type`	Specifies the type of external provisioning infrastructure.
`VirtualMachine.Admin.Owner`	Specifies the user name of the machine owner.
`BMC.Software.Install`	Set to True to enable BMC BladeLogic Configuration Manager integration.
`EPI.Server.Name`	Specifies the name of the external provisioning infrastructure server, for example, the name of the server hosting BMC BladeLogic. If at least one general BMC EPI agent was installed without specifying a BMC BladeLogic Configuration Manager host, this value directs the request to the desired server. If only dedicated BMC EPI agents for specific BMC BladeLogic Configuration Manager hosts were installed, this value must exactly match the server name configured for one of these agents.
`BMC.Service.Profile`	Specifies the name of the default authentication profile on the BMC BladeLogic server.
`BMC.Software.BatchLocation`	Specifies the location in BMC BladeLogic configuration where software jobs are deployed. This value must match the appropriate value of `Vrm.Software.IdNNNN`. For example, a valid value could be `/Application Deployment`.
`VMware.VirtualCenter.OperatingSystem`	Specifies the vCenter Server guest operating system version (`VirtualMachineGuestOsIdentifier`) with which vCenter Server creates the machine. This operating system version must match the operating system version to be installed on the provisioned machine. Administrators can create property groups using one of several property sets, for example, `VMware[OS_Version]Properties`, that are predefined to include the correct `VMware.VirtualCenter.OperatingSystem` values. This property is for virtual provisioning. For related information, see the enumeration type `VirtualMachineGuestOsIdentifier` in vSphere API/SDK Documentation. For a list of currently accepted values, see the vCenter Server documentation.

Custom Properties To Make BMC BladeLogic Configuration Manager Software Jobs Available

Configure BMC BladeLogic Configuration Manager jobs for vRealize Automation integrations. Make all software jobs available to machine requesters to select from, or specify a software job to apply to all machines provisioned from the blueprint.

Table 9-2. Custom Properties to Make Software Jobs Available

Custom Property	Description
`LoadSoftware`	Set to True to enable software install options.

`Vrm.Software.Id`*NNNN*	Specifies a software job or policy to be applied to all machines provisioned from the blueprint. Set the value to `job_type=job_path`, where `job_type` is the numeral that represents the BMC BladeLogic job type and `job_path` is the location of the job in BMC BladeLogic, for example `4=/Utility/putty`. *NNNN* is a number from 1000 to 1999. The first property must start with 1000 and increment in numerical order for each additional property.
	`1 – AuditJob`
	`2 – BatchJob`
	`3 – ComplianceJob`
	`4 – DeployJob`
	`5 – FileDeployJob`
	`6 – NSHScriptJob`
	`7 – PatchAnalysisJob`
	`8 – SnapshotJob`

Optional Custom Properties for BMC BladeLogic Configuration Manager Integrations

You can also use optional custom properties that are commonly used with BMC BladeLogic Configuration Manager blueprints.

Table 9-3. Optional Custom Properties for BMC BladeLogic Configuration Manager Integrations

Property	Definition
`BMC.AddServer.Delay`	Specifies the number of seconds to wait before adding the machine to BMC BladeLogic Configuration Manager. The default is 30.
`BMC.AddServer.Retry`	Specifies the number of seconds to wait before retrying if the first attempt to add the machine to BMC BladeLogic Configuration Manager is unsuccessful. The default is 100.

Publish a Blueprint

You can publish a blueprint for use in machine provisioning and optionally for reuse in another blueprint. To use the blueprint for requesting machine provisioning, you must entitle the blueprint after publishing it. Blueprints that are consumed as components in other blueprints do not required entitlement.

Prerequisites

- Log in to vRealize Automation as an **infrastructure architect**.
- Create a blueprint. See *Checklist for Creating vRealize Automation Blueprints*.

Procedure

1. Click the **Design** tab.
2. Click **Blueprints**.
3. Point to the blueprint to publish and click **Publish**.
4. Click **OK**.

Results

The blueprint is published as a catalog item but you must first entitle it to make it available to users in the service catalog.

What to do next

Add the blueprint to the catalog service and entitle users to request the catalog item for machine provisioning as defined in the blueprint.

IaaS Integration for HP Server Automation

IaaS Integration for HP Server Automation provides information about integrating HP Server Automation with VMware vRealize ™ Automation.

This documentation provides information on how you can use an HP Server Automation boot image or an HP Server Automation template to provision virtual machines by cloning.

Intended Audience

This information is intended for system administrators, tenant administrators, fabric administrators, and business group managers of vRealize Automation. This content is written for experienced Windows or Linux system administrators who are familiar with virtualization technology and the basic concepts described in *Foundations and Concepts*.

HP Server Automation Overview

You can you can provision virtual machines by using an HP Server Automation boot image or by provisioning by cloning and using an HP Server Automation template when you integrateHP Server Automation with vRealize Automation.

You can optionally identify the HP Server Automation policies to make available in vRealize Automation. Machine requestors can select from among these policies to install software on the requested machine, or you can specify HP Server Automation policies in the blueprint to be applied to every machine that is provisioned from that blueprint.

Integration Requirements Overview

The following is a high-level overview of the requirements for integrating HP Server Automation with vRealize Automation:

- A system administrator installs Microsoft PowerShell on the installation host prior to installing the agent.

  The required version of Microsoft PowerShell depends on the operating system of the installation host and might have been installed with that operating system. See Microsoft Help and Support.

- A system administrator installs the HP Server Automation snap-in on at least one host for vRealize Automation external provisioning integration (EPI) installation. See Install the HP Server Automation PowerShell Snap-In.
- A system administrator sets the PowerShell execution policy to RemoteSigned. See Set the PowerShell Execution Policy to RemoteSigned.
- A system administrator installs at least one EPI agent. See Install an EPI Agent for HP Server Automation.
- A system administrator sets up the selected integration method. See Integrating HP Server Automation.
- A system administrator enables software installation from HP Server Automation. See Enable vRealize Automation Software Installation from HP Server Automation.
- A tenant administrator or a business group manager creates a blueprint that enables the deployment of software jobs. See Creating Blueprints for HP Server Automation.
- A tenant administrator or a business group manager publishes the blueprint. See Publish a Blueprint.

Install the HP Server Automation PowerShell Snap-In

The HP Server Automation snap-in must be installed on at least one host for vRealize Automation external provisioning integration (EPI) installation prior to installing the EPI agent.

Prerequisites

- Obtain the HP Server Automation Snap-in software from the HP Server Automation installation media.
- Log in to the vRealize Automation console as a **system administrator**.

Procedure

1. Click **Start**, right-click **Command Prompt**, and click **Run as administrator**.
2. Change to the directory that contains the PowerShell snap-in.
3. Type `msiexec /i OPSWpowershell-37.0.0.5-0.msi`.
4. Complete the installation by accepting all defaults.

5 Select **Start > All Programs > Windows Power- Shell 1.0 > Windows PowerShell**.

6 Type `Add-PSSnapin 'OpswareSasPs'`.

7 Type `Exit`.

Set the PowerShell Execution Policy to RemoteSigned

You must set the PowerShell Execution Policy from Restricted to RemoteSigned or Unrestricted to allow local PowerShell scripts to run.

- For more information about PowerShell Execution Policy, type `help about_signing` or `help Set-ExecutionPolicy` at the PowerShell command prompt.

Prerequisites

- Log in as a Windows administrator.
- Install the HP Server Automation PowerShell Snap-In.

Procedure

1 Select **Start > All Programs > Windows PowerShell version > Windows PowerShell**.

2 Type `Set-ExecutionPolicy RemoteSigned` to set the policy to RemoteSigned.

3 Type `Set-ExecutionPolicy Unrestricted` to set the policy to Unrestricted.

4 Type `Get-ExecutionPolicy` to verify the current settings for the execution policy.

5 Type `Exit`.

Install an EPI Agent for HP Server Automation

A system administrator must install at least one vRealize Automation EPI agent to manage interaction with HP Server Automation. The agent can be installed anywhere, including the vRealize Automation server or the HP Server Automation server, as long as the agent can communicate with both servers.

Prerequisites

- Verify that the HP Server Automation PowerShell Snap-in is installed on the same host as your EPI agent. If the EPI agent is installed before the snap-in, then the agent service must be restarted after the snap-in is installed. See Install the HP Server Automation PowerShell Snap-In.
- The agent must be installed on Windows Server 2008 SP1, Windows Server 2008 SP2 (32 or 64-bit), Windows Server 2008 R2 system, or Windows 2012 with .NET 4.5.
- The credentials of the agent must have administrative access to all HP Server Automation hosts with which the agent will interact.
- Install the IaaS components, including the Manager Service and Website.

- See *Installing vRealize Automation* for complete information about installing vRealize Automation agents.
- Log in to the vRealize Automation console as a **system administrator**.

Procedure

1 Select **Custom Install** and **Proxy Agent** on the Installation Type page.

2 Accept the root install location or click **Change** and select an installation path.

Even in a distributed deployment, you might sometimes install more than one IaaS component on the same Windows server.

If you install more than one IaaS component, always install them to the same path.

3 Click **Next**.

4 Log in with administrator privileges for the Windows services on the installation machine.

The service must run on the same installation machine.

5 Click **Next**.

6 Select **EPIPowerShell** from the Agent type list.

7 Enter an identifier for this agent in the **Agent name** text box.

Maintain a record of the agent name, credentials, endpoint name, and platform instance for each agent. You need this information to configure endpoints and to add hosts in the future.

Important For high availability, you may add redundant agents and configure them identically. Otherwise, keep agents unique.

Option	Description
Redundant agent	Install redundant agents on different servers. Name and configure redundant agents identically.
Standalone agent	Assign a unique name to the agent.

8 Configure a connection to the IaaS Manager Service host.

Option	Description
With a load balancer	Enter the fully qualified domain name and port number of the load balancer for the Manager Service component, *mgr-svc-load-balancer.mycompany.com*:443. Do not enter IP addresses.
Without a load balancer	Enter the fully qualified domain name and port number of the machine where you installed the Manager Service component, *mgr-svc.mycompany.com*:443. Do not enter IP addresses.

The default port is 443.

9 Configure a connection to the IaaS Web server.

Option	Description
With a load balancer	Enter the fully qualified domain name and port number of the load balancer for the Web server component, *web-load-balancer.mycompany.com*:443. Do not enter IP addresses.
Without a load balancer	Enter the fully qualified domain name and port number of the machine where you installed the Web server component, *web.mycompany.com*:443. Do not enter IP addresses.

The default port is 443.

10 Click **Test** to verify connectivity to each host.

11 Click **Opsware** in **EPI Type**.

12 Type the fully qualified domain name of the managed server in the **EPI Server** text box.

Optionally, you can leave this blank to let the agent interact with multiple hosts.

The HP Server Automation server with which the agent interacts when provisioning a machine by using HP Server Automation depends on the value of the required custom property, `EPI.Server.Name`, in the blueprint.

Therefore, if you install a dedicated EPI agent by specifying an HP Server Automation server name during installation, only machines whose `EPI.Server.Name` property exactly matches the server name configured for the agent can be provisioned by that server.

If you install a general EPI agent by not specifying an HP Server Automation server name during installation, a machine can be provisioned by any server specified in the blueprint `EPI.Server.Name` property, that is assuming the agent can contact that server.

Note If no matching agent can be found, or there are no agents with unspecified server values, Opsware provisioning will wait until a suitable agent is found.

13 Click **Add**.

14 Click **Next**.

15 Click **Install** to begin the installation.

After several minutes a success message appears.

16 Click **Next**.

17 Click **Finish**.

What to do next

Determine the type of integration method to use. See Integrating HP Server Automation.

Extend the Default Software Installation Timeout

When you install the software for the integration product, the software might take longer to install than the default 30-minute timeout. You can increase the default timeout to a value that allows the installation to finish.

Procedure

1. Navigate to the Manager Service installation directory. Typically, this is `%System-Drive%\Program Files x86\VMware\vCAC\Server`.
2. Create a backup of the `ManagerService.exe.config` file.
3. Open the `ManagerService.exe.config` file and locate the `workflowTimeoutConfigurationSection` element and increase the value of the `DefaultTimeout` attribute from 30 minutes to your desired limit.
4. Click **Save** and close the file.
5. Select **Start > Administrative Tools > Services**, and restart the vRealize Automation service.

Integrating HP Server Automation

The steps required to integrate HP Server Automation with vRealize Automation depend on which provisioning method you want to use and whether you want to enable software installation from HP Server Automation.

When provisioning virtual machines, you can select from the following integration methods:

- Provision by using a system from which HP Server Automation deploys images that is available on the network.
- Provision by cloning by using a template that is prepared for HP Server Automation.

You can optionally identify the HP Server Automation policies to make available in vRealize Automation. Machine requestors can select from among these policies to install software on the requested machine, or you can specify HP Server Automation policies in the blueprint to be applied to every machine that is provisioned from that blueprint.

Enable Provisioning from HP Server Automation Boot Images

A system administrator can use an HP Server Automation boot image to enable vRealize Automation to provision machines by using that instance of HP Server Automation.

Prerequisites

- A system from which HP Server Automation deploys images is available on the network.
- An EPI agent installed. See Install an EPI Agent for HP Server Automation.
- Log in to the vRealize Automation console as a **system administrator**.

Procedure

1. On the EPI/Opsware Agent host, select **Start > Administrative Tools > Services**, and stop the vRealize Automation EPI/Opsware Agent.
2. On the EPI agent installation host, which may be the same as the Manager Service host, change to the EPI agent installation directory, typically `%SystemDrive%\Program Files (x86)\VMware\vCAC Agents\agent_name`.
3. Edit the agent configuration file, `VRMAgent.exe.config`, in the EPI agent installation directory.

   a. Locate the following line.

   ```
   <DynamicOps.Vrm.Agent.EpiPowerShell
   registerScript="CitrixProvisioningRegister.ps1"
   unregisterScript="CitrixProvisioningUnregister.ps1"/>
   ```

   b. Change the line to match the following line.

   ```
   <DynamicOps.Vrm.Agent.EpiPowerShell
   registerScript="CreateMachine.ps1"
   unregisterScript="DisposeVM.ps1"/>
   ```

4. Create an HP SA password file in the `Scripts` folder.

   The credentials you provide for this file must have administrator access to all instances of HP SA with which the agent will interact.

   a. Select **Start > All Programs > Windows Power- Shell 1.0 > Windows PowerShell**.

   b. Change to the `Scripts` directory.

   c. Type `\CreatePasswordFile.ps1 username`.

   d. Type the password when prompted.

   e. Type **`Exit`**.

5. On the vRealize Automation EPI/Opsware Agent host, select **Start > Administrative Tools > Services**, and then start or restart the vRealize Automation EPI/Opsware Agent service.

Preparing an HP Server Automation Template for Cloning

You can use an HP Server Automation template to integrate with vRealize Automation.

To create the HP Server Automation template, you must create a reference machine and add customization specifications to it.

For Windows, see Prepare a Reference Machine for Windows.

For Linux, see Prepare a Reference Machine for Linux.

Prepare a Reference Machine for Linux

You must prepare a reference machine and convert it to a template for cloning to add software installation by HP Server Automation to provisioning by cloning.

Procedure

1. Add the HP Server Automation agent installation package to the cloning template.
2. Copy the HP Server Automation agent installer to the reference machine.
3. Create a script to run the installer and install the HP Server Automation agent.
4. Copy the script to the reference machine.
5. Add the customization required to invoke the agent after provisioning, so that the agent is installed on each cloned machine.

Results

Note Do not install HP Server Automation on the reference machine. The agent must be installed by using the customization specification or postinstall script following cloning.

What to do next

- Optionally, identify the HP Server Automation policies to make available in vRealize Automation. See Enable vRealize Automation Software Installation from HP Server Automation
- Create a blueprint for the type of HP Server Automation integration you want to enable. See Creating Blueprints for HP Server Automation.

Prepare a Reference Machine for Windows

You must prepare a reference machine and convert it to a template for cloning before you can add software installation by HP Server Automation to provisioning by cloning.

Procedure

1. Add the HP Server Automation agent installation package to the cloning template.
2. Copy the HP Server Automation agent installer to the `C:\` directory of the reference machine.
3. Add the customization required to invoke the agent after provisioning, by adding following line to the `Run Once` section of the customization specification.

   ```
   C:\opswareagentinstaller --opsw_gw_addr opswareipaddress:3001 -s --force_sw_reg
   ```

   ```
   --force_full_hw_reg
   ```

   This customization also installs the agent on each cloned machine.
4. Replace *opswareagentinstaller* with the name of the HP Server Automation agent installer executable.

5 Replace *opswareipaddress* with the IP address of the server that is hosting the HP Server Automation instance that installs the software.

For example:

```
C:\ opsware-agent-37.0.0.2.61-win32-6.0.exe --opsw_gw_addr 10.20.100.52:3001 -s --force_sw_reg --
force_full_hw_reg
```

What to do next

- Optionally, identify the HP Server Automation policies to make available in vRealize Automation. See Enable vRealize Automation Software Installation from HP Server Automation.
- Create a blueprint for the type of HP Server Automation integration you want to enable. See Creating Blueprints for HP Server Automation.

Enable vRealize Automation Software Installation from HP Server Automation

A system administrator can optionally identify the HP Server Automation policies to make available in vRealize Automation. Machine requestors can select from among these policies to install software on the requested machine, or HP Server Automation policies can be specified in the blueprint to be applied to every machine that is provisioned from that blueprint.

Prerequisites

- An EPI agent installed. See Install an EPI Agent for HP Server Automation.
- Log in to the vRealize Automation console as a **system administrator**.

Procedure

1 Create a text file named `Software.txt` in the Web site directory under the vRealize Automation server install directory, typically `%SystemDrive%\Program Files (x86)\VMware\vCAC\Server\Website`.

Each line of the `Software.txt` file must be in the following format:

```
Software_policy_description=software_policy_name
```

2 Define the label and name of the software policy that a user sees when they request to install software from the HP Server Automation instance.

a Replace *Software_policy_description* with the label that identifies the software policy.

b Replace *software_policy_name* with the name of the policy.

For example, a `Software.txt` file, where you want to provide the user the ability to select HP Server Automation Windows ISM Tool, HP Server Automation Linux ISM Tool, or both, might contain the following information:

```
HP SA Windows ISM Tool=Windows_ISMtool
```

```
HP SA Linux ISM Tool=RedHatLinux_ISMtool
```

Creating Blueprints for HP Server Automation

The type of blueprint you create depends on how you want to enable HP Server Automation integration.

You need to create a blueprint that includes all of the information required for machine provisioning and the information required for HP Server Automation integration for either of the following integration methods:

- Provisioning by using a system from which HP Server Automation deploys images.
- Provisioning by cloning from a template that is prepared for HP Server Automation.

You can optionally identify the HP Server Automation policies to make available in vRealize Automation. Machine requestors can select from among these policies to install software on the requested machine, or you can specify HP Server Automation policies in the blueprint to be applied to every machine that is provisioned from that blueprint.

Create a Virtual Blueprint for Creating from an HP Server Automation Boot Image

A tenant administrator or business group manager creates a blueprint for using an HP Server Automation boot image to deploy HP Server Automation software jobs on machines provisioned from it.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **business group manager**.
- Obtain the following information from your fabric administrator:
  - The name of the HP Server Automation server to be used as the value for the `EPI.Server.Name` custom property.
  - The name of the HP Server Automation image to be used as the value for the `Opsware.BootImage.Name` custom property.

  - Optionally, information about the custom properties and values to be applied to all machines provisioned from the blueprint. See Custom Properties for HP Server Automation Integration.

    Note A fabric administrator can create a property group by using the property set `HPSABuildMachineProperties`, which allows HP Server Automation integration in provisioning by using a boot image, or `HPSASoftwareProperties`, which allows HP Server Automation integration in software deployment. These property groups makes it easier for tenant administrators and business group managers to include this information in their blueprints.

- For information on creating a virtual blueprint, see *IaaS Configuration for Virtual Platforms*.

Procedure

1. Select **Design > Blueprints**.
2. In the Actions column, click the down arrow and click **Edit**.
3. Click the **Properties** tab.
4. (Optional) Select one or more property groups.

   Property groups contain multiple custom properties.
5. (Optional) Add any custom properties to your machine component.
   a. Click **New Property**.
   b. Enter the custom property in the **Name** text box.
   c. (Optional) To encrypt the custom property in the database, select the **Encrypted** check box.
   d. Enter the value of the custom property in the **Value** text box.
   e. (Optional) To require the user to provide a value when they request a machine, select the **Prompt user** check box.

      If you choose to prompt users for a value, any value you provide for the custom property is presented to them as the default. If you do not provide a default, users cannot continue with the machine request until they provide a value for the custom property.
   f. Click the **Save** icon (✓).
6. Click the **Build Information** tab.
7. Select **Create** and the **ExternalProvisioningWorkflow** workflow.
8. Click **OK**.

Results

Your blueprint is saved.

What to do next

Publish your blueprint to make it available as a catalog item. See Publish a Blueprint.

Create a Blueprint for Cloning from an HP Server Automation Template

A tenant administrator or business group manager creates a blueprint that enables the deployment of HP Server Automation software jobs on machines provisioned from it.

Prerequisites

- Log in to vRealize Automation as a **tenant administrator** or **business group manager**.
- Obtain the following information from your fabric administrator:
  - An HP Server Automation template. See Preparing an HP Server Automation Template for Cloning.
  - The clone blueprint that you want to integrate with HP Server Automation.
  - Optionally, information about the custom properties and values to be applied to all machines provisioned from the blueprint. See Custom Properties for HP Server Automation Integration.

    Note A fabric administrator can create a property group by using the property set `HPSABuildMachineProperties`, which allows HP Server Automation integration in provisioning by using a boot image, or `HPSASoftwareProperties`, which allows HP Server Automation integration in software deployment. These property groups makes it easier for tenant administrators and business group managers to include this information in their blueprints.

  - If a policy is to be applied to all machines provisioned from the blueprint, you must include the custom property `Vrm.Software.Id`*NNNN* where *NNNN* is a number from 1000 to 1999, and the value is set to the name of the policy, for example Windows_ISMtool.
  - The name of the customization specification to be added to the blueprint. See Preparing an HP Server Automation Template for Cloning.
- For information on how to create a blueprint for cloning by using the template and customization specification provided to you by your fabric administrator, see *IaaS Configuration for Virtual Platforms*.

Procedure

1. Select **Design > Blueprints**.
2. Locate the clone blueprint that you want to integrate with HP Server Automation.
3. In the Actions column, click the down arrow and click **Edit**.
4. Click the **Properties** tab.

5. (Optional) Select one or more property groups.

   Property groups contain multiple custom properties.

6. (Optional) Add any custom properties to your machine component.

   a. Click **New Property**.

   b. Enter the custom property in the **Name** text box.

   c. (Optional) To encrypt the custom property in the database, select the **Encrypted** check box.

   d. Enter the value of the custom property in the **Value** text box.

   e. (Optional) To require the user to provide a value when they request a machine, select the **Prompt user** check box.

      If you choose to prompt users for a value, any value you provide for the custom property is presented to them as the default. If you do not provide a default, users cannot continue with the machine request until they provide a value for the custom property.

   f. Click the **Save** icon (✓).

7. Click **OK**.

Results

Your blueprint is saved.

What to do next

Publish your blueprint to make it available as a catalog item. See Publish a Blueprint.

Custom Properties for HP Server Automation Integration

vRealize Automation includes custom properties that you can use to provide additional controls for HP Server Automation integration. Some custom properties are required for HP Server Automation integration. Other custom properties are optional.

Required Custom Properties for HP Server Automation Integration

Certain custom properties are required for a blueprint to work with HP Server Automation.

Table 9-4. Required Custom Properties for HP Server Automation Integration

Property	Definition
`VMware.VirtualCenter.OperatingSystem`	Specifies the vCenter Server guest operating system version (`VirtualMachineGuestOsIdentifier`) with which vCenter Server creates the machine. This operating system version must match the operating system version to be installed on the provisioned machine. Administrators can create property groups using one of several property sets, for example, `VMware[OS_Version]Properties`, that are predefined to include the correct `VMware.VirtualCenter.OperatingSystem` values. This property is for virtual provisioning.
`VirtualMachine.EPI.Type`	Specifies the type of external provisioning infrastructure.
`EPI.Server.Name`	Specifies the name of the external provisioning infrastructure server, for example, the name of the server hosting BMC BladeLogic. If at least one general BMC EPI agent was installed without specifying a BMC BladeLogic Configuration Manager host, this value directs the request to the desired server.
`Opsware.Software.Install`	Set to True to allow HP Server Automation to install software.
`Opsware.Server.Name`	Specifies the fully qualified name of the HP Server Automation server.
`Opsware.Server.Username`	Specifies the user name provided when a password file in the agent directory was created, for example opswareadmin. This user name requires administrative access to the HP Server Automation instance.
`Opsware.BootImage.Name`	Specifies the boot image value as defined in HP Server Automation for the 32-bit WinPE image, for example winpe32. The property is not required when provisioning by cloning.
`Opsware.Customer.Name`	Specifies a customer name value as defined in HP Server Automation, for example MyCompanyName.
`Opsware.Facility.Name`	Specifies a facility name value as defined in HP Server Automation, for example Cambridge.
`Opsware.Machine.Password`	Specifies the default local administrator password for an operating system sequence WIM image such as `Opsware.OSSequence.Name` as defined in HP Server Automation, for example P@ssword1.
`Opsware.OSSequence.Name`	Specifies the operating system sequence name value as defined in HP Server Automation, for example Windows 2008 WIM.
`Opsware.Realm.Name`	Specifies the realm name value as defined in HP Server Automation, for example Production.
`Opsware.Register.Timeout`	Specifies the time, in seconds, to wait for creation of a provisioning job to complete.

Table 9-4. Required Custom Properties for HP Server Automation Integration (continued)

Property	Definition
`VirtualMachine.CDROM.Attach`	Set to False to provision the machine without a CD-ROM device. The default is True.
`Linux.ExternalScript.Name`	Specifies the name of an optional customization script, for example `config.sh`, that the Linux guest agent runs after the operating system is installed. This property is available for Linux machines cloned from templates on which the Linux agent is installed.
`Linux.ExternalScript.LocationType`	Specifies the location type of the customization script named in the `Linux.ExternalScript.Name` property. This can be either local or nfs.
`Linux.ExternalScript.Path`	Specifies the local path to the Linux customization script or the export path to the Linux customization on the NFS server. The value must begin with a forward slash and not include the file name, for example `/scripts/linux/config.sh`.

Optional Custom Properties for HP Server Automation Integration

Certain custom properties are optional for a blueprint to work with HP Server Automation.

Table 9-5. Optional Custom Properties for HP Server Automation Integration

Property	Definition
`Opsware.ProvFail.Notify`	(Optional) Specifies the notification email address for HP Server Automation to use in the event of provisioning failure, for example provisionfail@lab.local.
`Opsware.ProvFail.Notify`	(Optional) Specifies the HP Server Automation user to whom ownership is assigned if provisioning fails.
`Opsware.ProvSuccess.Notify`	(Optional) Specifies the notification email address for HP Server Automation to use if provisioning is successful.
`Opsware.ProvSuccess.Owner`	(Optional) Specifies the HP Server Automation user to whom ownership is assigned if provisioning is successful.

Custom Properties That Make HP Server Automation Software Jobs Available

Depending on how your fabric administrator configures HP Server Automation jobs for vRealize Automation integration, you might have a choice between making all software jobs available to machine requesters to select, or you can specify jobs to apply to all machines provisioned from your blueprint.

Table 9-6. Custom Properties to Make Software Jobs Available

Property	Definition
`LoadSoftware`	Set to True to enable software install options.
`Vrm.Software.Id`	(Optional) Specifies an HP Server Automation policy to be applied to all machines provisioned from the blueprint. *NNNN* is a number from 1000 to 1999. The first property must start with 1000 and increment in numerical order for each additional property.

Publish a Blueprint

You can publish a blueprint for use in machine provisioning and optionally for reuse in another blueprint. To use the blueprint for requesting machine provisioning, you must entitle the blueprint after publishing it. Blueprints that are consumed as components in other blueprints do not required entitlement.

Prerequisites

- Log in to vRealize Automation as an **infrastructure architect**.
- Create a blueprint. See *Checklist for Creating vRealize Automation Blueprints*.

Procedure

1. Click the **Design** tab.
2. Click **Blueprints**.
3. Point to the blueprint to publish and click **Publish**.
4. Click **OK**.

Results

The blueprint is published as a catalog item but you must first entitle it to make it available to users in the service catalog.

What to do next

Add the blueprint to the catalog service and entitle users to request the catalog item for machine provisioning as defined in the blueprint.

Maintaining and Customizing vRealize Automation Components and Options

10

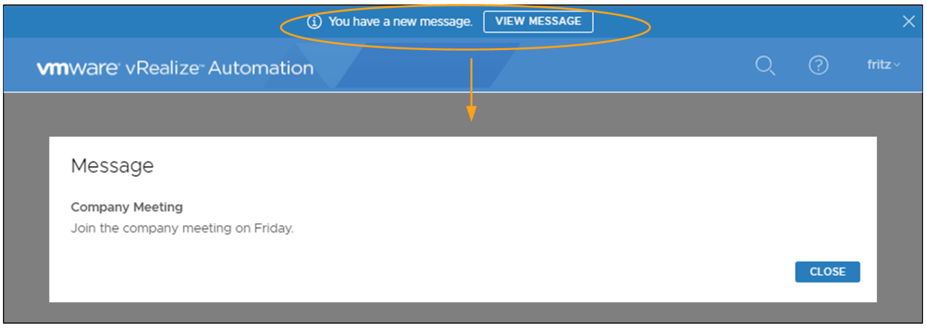
You can manage provisioned machines and other aspects of your vRealize Automation deployment.

This chapter includes the following topics:

Broadcast a Message to All Users

As the tenant administrator, you can broadcast a message to all users. The message notification appears at the top of browser page. Your users click the notification to see the message.

As a user, you can access the message from the banner, or from your user drop-down menu on the header.



Message

Company Meeting

Join the company meeting on Friday.

You use the message board to broadcast a text message or a Web page. Depending on the Web page, your users can navigate through the website in the message board.

The message board has the following limitations.

Table 10-1. Message Board Limitations

Option	Limitations
URL message limitations	■ The target URL must be included in the message board allowlist. See Create a Message Board URL Allowlist. ■ You can only publish content that is hosted on an https site. ■ You cannot use self-signed certificates. The option to accept the certificate does not appear in the message board. ■ The message board URL is embedded in an iframe. Some websites do not work in iframe and an error appears. One cause of the failure is the X-Frame-Options DENY or SAMEORIGIN in the header on the target website. If your target website is one that you control, you can set the X-Frame-Options header to `X-Frame-Options: ALLOW-FROM https://<vRealizeAutomationApplicanceURL>`. ■ Some websites have a redirect to a top-level page that might refresh entire vRealize Automation page. This type of website does not work in the message board. The refresh is suppressed and a Loading... message appears on the message board. ■ If you display an internal HTML page, the page cannot have the vRealize Automation host as the URL.
Custom message limitations	■ To maintain security, the Custom Message allows simple markup, but does not support HTML code. For example, you cannot use <href> to link to a website. You must use the URL message option.

Prerequisites

Log in to vRealize Automation as a **tenant administrator**.

Procedure

1. Click the **Administration** tab.
2. Select **Notifications > Message Board**
3. In the **Type** drop-down menu, select the message type.

Option	Description
None	Removes the message notification.
Custom Message	Enter a plain text message.
URL	Enter the page URL. The URL must be included in the message board allowlist. See Create a Message Board URL Allowlist. To log the user into a website, most commonly your internal website, based on their vRealize Automation user ID, select **Include user ID**. The URL that is passed to the website similar to `http://company.com/internal/message?userID=richard_dawson@company.com`. This method allows your website to use the `window.location.search` JavaScript property to provide the current user's ID to your website.

4. Click **OK**.

Results

The message is broadcast as a banner to all your tenant users.

To change or remove the message, you must be logged in as the tenant administrator. To change the message, repeat the same steps. To remove the message, select None as the Type and click **OK**.

Create a Message Board URL Allowlist

As the security administrator, you configure an allowed list of URLs that can be used in the message board. This allowlist ensures added security.

Prerequisites

Log in to vRealize Automation as a **security administrator**.

Procedure

1. Select **Administration > Message Board Whitelist**.
2. Click **New**.
3. Add a URL and click **OK**.

   The URL entries can include the following content:

   - IP address or FQDN of a site. For example, https://docs.vmware.com.
   - Includes https.

- Can include allowed ports. If a port is not specified, the allowed ports are 80 and 443.

4 Repeat for each additional entry.

Results

A tenant administrator cannot add a URL to the message board unless it is included in this list.

What to do next

Verify that you can add and broadcast a URL included in your allowlist using the message board. See Broadcast a Message to All Users.

Starting Up and Shutting Down vRealize Automation

A system administrator performs a controlled shutdown or startup of vRealize Automation to preserve system and data integrity.

You can also use a controlled shutdown and startup to resolve performance or product behavior issues that can result from an incorrect initial startup. Use the restart procedure when only some components of your deployment fail.

Start Up vRealize Automation

When you start vRealize Automation after it was powered off for any expected or unexpected reason, you must start components in a specified order.

If you are managing deployment components in vCenter Server, you can start their guest operating systems from there.

Prerequisites

Verify that the load balancers that your deployment uses are running.

Procedure

1. If you are using a legacy, standalone PostgreSQL database, start that server.
2. In any order, start standalone vRealize Automation MS SQL servers.
3. In a deployment that uses load balancers with health checks, deactivate all health checks except pings.
4. Start the primary vRealize Automation appliance.
5. In the primary vRealize Automation appliance management interface, look under the **Cluster** tab to check whether the system is in synchronous or asynchronous mode. A single-appliance deployment is always asynchronous.
   - If the deployment is synchronous, start the remaining vRealize Automation appliances.
   - If the deployment is asynchronous, go to the primary vRealize Automation appliance management interface, and wait until the licensing service is running and REGISTERED.

Afterward, start any remaining vRealize Automation appliances.

6 After all appliances have started, use their management interfaces to verify that services are running and REGISTERED.

   It might take 15 or more minutes for appliances to start.

7 Start all IaaS Web nodes, and wait 5 minutes.

8 Start the primary Manager Service node, and wait 2 to 5 minutes.

9 In a distributed deployment with multiple Manager Service nodes, start secondary Manager Service nodes, and wait 2 to 5 minutes.

   On secondary machines, do not start or run the Windows service unless you are configured for automatic Manager Service failover.

10 In any order, start the DEM Orchestrator, DEM Workers, and all vRealize Automation proxy agents.

   You do not need to wait for one startup to finish before starting another.

11 If you had to deactivate load balancer health checks, reactivate them.

12 Verify that started services are running and REGISTERED.

   a In a browser, log in to the primary vRealize Automation appliance management interface.

      https://vrealize-automation-appliance-FQDN:5480

   b Click the **Services** tab.

   c Monitor service startup progress by clicking **Refresh**.

Results

When all services are REGISTERED, the deployment is ready.

Restart vRealize Automation

Restarting vRealize Automation components might help resolve problems. You must restart components in a specified order.

If you are managing deployment components in vCenter Server, you can restart their guest operating systems from there.

If you can't perform a restart, try the instructions in Shut Down vRealize Automation and Start Up vRealize Automation instead.

Prerequisites

- Verify that all load balancers that your deployment uses are running.

Procedure

1 Verify that the vRealize Automation appliance database is set to asynchronous mode. If necessary, use the management interface to change it to asynchronous mode.

  You may return to synchronous mode after completing the whole procedure. See Managing the vRealize Automation Postgres Appliance Database for more information.

2 Restart the primary vRealize Automation appliance, and wait for startup to finish.

3 Use the primary vRealize Automation appliance management interface to verify that the licensing service is running and REGISTERED.

4 Restart the remaining vRealize Automation appliances at the same time.

5 Wait for the appliances to restart, and use their management interfaces to verify that services are running and REGISTERED.

  It might take 15 or more minutes for appliances to restart.

6 Restart the primary Web node, and wait for startup to finish.

7 If you are running a distributed deployment with multiple Web nodes, restart secondary Web nodes, and wait for startups to finish.

8 Restart Manager Service nodes, and wait for startups to finish.

  If you are running automatic Manager Service failover, and you want to keep the active and passive nodes the same, restart in the following order:

  a Stop the passive Manager Service nodes without restarting them.

  b Completely restart the active Manager Service node.

  c Start the passive Manager Service nodes.

9 In any order, restart the DEM Orchestrator, DEM Workers, and all vRealize Automation proxy agents. Wait for all startups to finish.

  You do not need to wait for one restart to finish before restarting another.

10 Verify that restarted services are running and REGISTERED.

  a In a browser, log in to the primary vRealize Automation appliance management interface.

    https://vrealize-automation-appliance-FQDN:5480

  b Click the **Services** tab.

  c Monitor service startup progress by clicking **Refresh**.

Results

When all services are REGISTERED, the deployment is ready.

Shut Down vRealize Automation

To preserve data integrity, you must shut down vRealize Automation in a specified order.

If you are managing deployment components in vCenter Server, you can shut down their guest operating systems from there.

Procedure

1. In any order, shut down the DEM Orchestrator, DEM Workers, and all vRealize Automation proxy agents. Wait for shutdown to finish.
2. Shut down Manager Service nodes, and wait for shutdown to finish.
3. In distributed deployments with multiple Web nodes, shut down secondary Web nodes, and wait for shutdown to finish.
4. Shut down the primary Web node, and wait for shutdown to finish.
5. In distributed deployments with multiple vRealize Automation appliances in synchronous mode, use the vRealize Automation appliance management interface to change to asynchronous mode.
6. In distributed deployments with multiple vRealize Automation appliances, shut down secondary appliances, and wait for shutdown to finish.
7. Shut down the primary vRealize Automation appliance, and wait for shutdown to finish.

   The primary vRealize Automation appliance is the one that contains the primary, or writeable, appliance database. Make note of which appliance is primary so that you can start back up in the correct order.
8. In any order, shut down any standalone vRealize Automation MS SQL servers, and wait for shutdown to finish.
9. If you are using a legacy, standalone PostgreSQL database, shut down that server.

Updating vRealize Automation Certificates

A system administrator can update or replace certificates for vRealize Automation components.

vRealize Automation contains three main components that use SSL certificates in order to facilitate secure communication with each other:

- vRealize Automation appliance
- IaaS website component
- IaaS manager service component

In addition, your deployment can have certificates for the vRealize Automation appliance management interface web site. Also, each IaaS machine runs a Management Agent that uses a certificate.

Note vRealize Automation uses several third party products, such as Rabbit MQ, to support a variety of functionality. Some of these products use their own self signed certificates that persist even if you replace primary vRealize Automation certificates with certificates supplied by a CA. Because of this situation, users cannot effectively control certificate use on specific ports, such as 5671 which is used by RabbitMQ for internal communication.

With one exception, changes to later components in this list do not affect earlier ones. The exception is that an updated certificate for IaaS components must be registered with vRealize Automation appliance.

Typically, self-signed certificates are generated and applied to these components during product installation. You might need to replace a certificate to switch from self-signed certificates to certificates provided by a certificate authority or when a certificate expires. When you replace a certificate for a vRealize Automation component, trust relationships for other vRealize Automation components are updated automatically.

For instance, in a distributed system with multiple instances of a vRealize Automation appliance, if you update a certificate for one vRealize Automation appliance all other related certificates are updated automatically.

Note vRealize Automation supports SHA2 certificates. The self-signed certificates generated by the system use SHA-256 With RSA Encryption. You might need to update to SHA2 certificates due to operating system or browser requirements.

The vRealize Automation appliance management interface provides options for updating or replacing certificates.

In a clustered deployment, you must initiate changes from the primary node interface.

- **Generate certificate** — Have vRealize Automation generate a self-signed certificate.
- **Import certificate** — Use your own certificate.
- **Provide certificate thumbprint** — Provide a certificate thumb print to use a certificate already in the certificate store on IaaS Windows servers.

  This option does not transmit the certificate from the vRealize Automation appliance to IaaS Windows servers. The option allows users to deploy existing certificates already on IaaS Windows servers without uploading the certificates in the vRealize Automation appliance management interface.
- **Keep Existing** — Continue to use the current certificate.

Certificates for the vRealize Automation appliance management interface web site do not have registration requirements.

Note If your certificate uses a passphrase for encryption, and you fail to enter it when replacing your certificate on the appliance, the certificate replacement fails, and the message `Unable to load private key` appears.

Virtual Machine Templates

After you change vRealize Automation appliance or IaaS Windows server certificates, you must update vRealize Automation guest and software agents on virtual machine templates so that the templates work again in vRealize Automation. If you don't update the agents, deployment requests involving software components fail with an error similar to the following example.

```
The following component requests failed: Linux. Request failed: Machine VM-001:
InstallSoftwareWorkflow. Install software work item timeout.
```

vRealize Orchestrator

After you change vRealize Automation certificates, you must update vRealize Orchestrator to trust the new certificates.

The vRealize Orchestrator component associated with your vRealize Automation deployment has its own certificates, but it must also trust the vRealize Automation certificates. By default, the vRealize Orchestrator component is embedded in vRealize Automation, although a few users elect to use an external vRealize Orchestrator. In either case, see the vRealize Orchestrator documentation for information about updating vRealize Orchestrator certificates.

If you run a multiple-node vRealize Orchestrator deployment behind a load balancer, all vRealize Orchestrator nodes must use the same certificate.

For More Information

For more about certificate troubleshooting, supportability, and trust requirements, see VMware Knowledge Base article 2106583.

Extracting Certificates and Private Keys

Certificates that you use with the virtual appliances must be in the PEM file format.

The examples in the following table use Gnu `openssl` commands to extract the certificate information you need to configure the virtual appliances.

Table 10-2. Sample Certificate Values and Commands (openssl)

Certificate Authority Provides	Command	Virtual Appliance Entries
RSA Private Key	openssl pkcs12 -in *path _to_.pfx certificate_file* -nocerts -out key.pem	**RSA Private Key**
PEM File	openssl pkcs12 -in *path _to_.pfx certificate_file* -clcerts -nokeys -out cert.pem	**Certificate Chain**
(Optional) Pass Phrase	n/a	**Pass Phrase**

Replace Certificates in the vRealize Automation Appliance

The system administrator can update or replace a self-signed certificate with a trusted one from a certificate authority. You can use Subject Alternative Name (SAN) certificates, wildcard certificates, or any other method of multi-use certification appropriate for your environment as long as you satisfy the trust requirements.

When you update or replace the vRealize Automation appliance certificate, trust with other related components is re-initiated automatically. See Updating vRealize Automation Certificates for more information about updating certificates.

Procedure

1 Log in to the vRealize Automation appliance management interface as root.

   https://*vrealize-automation-appliance-FQDN*:5480

2 Select **vRA > Certificates**.

3 Select the vRealize Automation component for which you are updating the certificate.

4 Select the appropriate action from the **Certificate Action** menu.

   If you are using a PEM-encoded certificate, for example for a distributed environment, select **Import**.

   Certificates that you import must be trusted and must also be applicable to all instances of vRealize Automation appliance and any load balancer through the use of Subject Alternative Name (SAN) certificates.

If you want to generate a CSR request for a new certificate that you can submit to a certificate authority, select **Generate Signing Request**. A CSR helps your CA create a certificate with the correct values for you to import.

Note If you use certificate chains, specify the certificates in the following order:

a Client/server certificate signed by the intermediate CA certificate

b One or more intermediate certificates

c A root CA certificate

Option	Action
Keep Existing	Leave the current SSL configuration. Select this option to cancel your changes.
Generate Certificate	a The value displayed in the **Common Name** text box is the Host Name as it appears on the upper part of the page. If any additional instances of the vRealize Automation appliance available, their FQDNs are included in the SAN attribute of the certificate. b Enter your organization name, such as your company name, in the **Organization** text box. c Enter your organizational unit, such as your department name or location, in the **Organizational Unit** text box. d Enter a two-letter ISO 3166 country code, such as `US`, in the **Country** text box.
Generate Signing Request	a Select **Generate Signing Request**. b Review the entries in the **Organization**, **Organization Unit**, **Country Code**, and **Common Name** text boxes. These entries are populated from the existing certificate. You can edit these entries if needed. c Click **Generate CSR** to generate a certificate signing request, and then click the **Download the generated CSR here** link to open a dialog that enables you to save the CSR to a location where you can send it to a certificate authority. d When you receive the prepared certificate, click **Import** and follow instructions for importing a certificate into vRealize Automation.
Import	a Copy the certificate values from BEGIN PRIVATE KEY to END PRIVATE KEY, including the header and footer, and paste them in the **RSA Private Key** text box. b Copy the certificate values from BEGIN CERTIFICATE to END CERTIFICATE, including the header and footer, and paste them in the **Certificate Chain** text box. For multiple certificate values, include a BEGIN CERTIFICATE header and END CERTIFICATE footer for each certificate. **Note** In the case of chained certificates, additional attributes may be available. c (Optional) If your certificate uses a pass phrase to encrypt the certificate key, copy the pass phrase and paste it in the **Passphrase** text box.

5 Click **Save Settings**.

   A vRealize Automation appliance certificate update requires vRealize Automation services to gracefully restart. The restart might take anywhere from 15 minutes to an hour depending on the number of vRealize Automation appliances in your environment.

   After the restart, the certificate details for all applicable instances of the vRealize Automation appliance appear on the page.

6 If required by your network or load balancer, copy the imported or newly created certificate to the virtual appliance load balancer.

   You might need to enable root SSH access in order to export the certificate.

   a If not already logged in, log in to the vRealize Automation appliance Management Console as root.

   b Click the **Admin** tab.

   c Click the **Admin** sub menu.

   d Select the **SSH service enabled** check box.

      Deselect the check box to deactivate SSH when finished.

   e Select the **Administrator SSH login** check box.

      Deselect the check box to deactivate SSH when finished.

   f Click **Save Settings**.

7 Confirm that you can log in to vRealize Automation console.

   a Open a browser and navigate to https://*vcac-hostname.domain.name*/vcac/.

      If you are using a load balancer, the host name must be the fully qualified domain name of the load balancer.

   b If prompted, continue past the certificate warnings.

   c Log in with `administrator@vsphere.local` and the password you specified when configuring Directories Management.

      The console opens to the **Tenants** page on the **Administration** tab. A single tenant named `vsphere.local` appears in the list.

8 If you are using a load balancer, configure and enable any applicable health checks.

Results

The certificate is updated.

Replace the Infrastructure as a Service Certificate

The system administrator can replace an expired certificate or a self-signed certificate with one from a certificate authority to ensure security in a distributed deployment environment.

You can use a Subject Alternative Name (SAN) certificate on multiple machines. Certificates used for the IaaS components (Website and Manager Service) must be issued with SAN values including FQDNs of all Windows hosts on which the corresponding component is installed and with the Load Balancer FQDN for the same component.

Procedure

1 Log in to the vRealize Automation appliance management interface as root.

   https://*vrealize-automation-appliance-FQDN*:5480

2 Select **vRA > Certificates**.

3 Click **IaaS Web** on the **Component Type** menu.

4 Go to the **IaaS Web Certificate** pane.

5 Select the certificate replacement option from the **Certificate Action** menu.

   If you are using a PEM-encoded certificate, for example for a distributed environment, select **Import**.

   Certificates that you import must be trusted and must also be applicable to all instances of vRealize Automation appliance and any load balancer through the use of Subject Alternative Name (SAN) certificates.

   Note If you use certificate chains, specify the certificates in the following order:

   a Client/server certificate signed by the intermediate CA certificate

   b One or more intermediate certificates

   c A root CA certificate

Option	Description
Keep Existing	Leave the current SSL configuration. Choose this option to cancel your changes.
Generate Certificate	a The value displayed in the **Common Name** text box is the Host Name as it appears on the upper part of the page. If any additional instances of the vRealize Automation appliance available, their FQDNs are included in the SAN attribute of the certificate. b Enter your organization name, such as your company name, in the **Organization** text box. c Enter your organizational unit, such as your department name or location, in the **Organizational Unit** text box. d Enter a two-letter ISO 3166 country code, such as `US`, in the **Country** text box.

Option	Description
Import	a Copy the certificate values from BEGIN PRIVATE KEY to END PRIVATE KEY, including the header and footer, and paste them in the **RSA Private Key** text box. b Copy the certificate values from BEGIN CERTIFICATE to END CERTIFICATE, including the header and footer, and paste them in the **Certificate Chain** text box. For multiple certificate values, include a BEGIN CERTIFICATE header and END CERTIFICATE footer for each certificate. **Note** In the case of chained certificates, additional attributes may be available. c (Optional) If your certificate uses a pass phrase to encrypt the certificate key, copy the pass phrase and paste it in the **Passphrase** text box.
Provide Certificate Thumbprint	Use this option if you want to provide a certificate thumbprint to use a certificate that is already deployed in the certificate store on the IaaS servers. Using this option will not transmit the certificate from the virtual appliance to the IaaS servers. It enables users to deploy existing certificates on IaaS servers without uploading them in the management interface.

6 Click **Save Settings**.

An IaaS Windows server certificate update requires vRealize Automation services to gracefully restart. The restart might take anywhere from 15 minutes to an hour depending on the number of vRealize Automation appliances in your environment.

After the restart, the certificate details appear on the page.

Replace the IaaS Manager Service Certificate

A system administrator can replace an expired certificate or a self-signed certificate with one from a certificate authority to ensure security in a distributed deployment environment.

You can use a Subject Alternative Name (SAN) certificate on multiple machines. Certificates used for the IaaS components (Website and Manager Service) must be issued with SAN values including FQDNs of all Windows hosts on which the corresponding component is installed and with the Load Balancer FQDN for the same component.

The IaaS Manager Service and the IaaS Web Service share a single certificate.

Procedure

1 Open a Web browser to the vRealize Automation appliance management interface URL.

2 Log in with user name `root` and the password you specified when deploying the vRealize Automation appliance.

3 Select **vRA > Certificates**.

4 Click **Manager Service** from the **Component Type** menu.

5 Select the certificate type from the **Certificate Action** menu.

If you are using a PEM-encoded certificate, for example for a distributed environment, select **Import**.

Certificates that you import must be trusted and must also be applicable to all instances of vRealize Automation appliance and any load balancer through the use of Subject Alternative Name (SAN) certificates.

Note If you use certificate chains, specify the certificates in the following order:

a Client/server certificate signed by the intermediate CA certificate

b One or more intermediate certificates

c A root CA certificate

Option	Description
Keep Existing	Leave the current SSL configuration. Choose this option to cancel your changes.
Generate Certificate	a The value displayed in the **Common Name** text box is the Host Name as it appears on the upper part of the page. If any additional instances of the vRealize Automation appliance available, their FQDNs are included in the SAN attribute of the certificate. b Enter your organization name, such as your company name, in the **Organization** text box. c Enter your organizational unit, such as your department name or location, in the **Organizational Unit** text box. d Enter a two-letter ISO 3166 country code, such as `US`, in the **Country** text box.
Import	a Copy the certificate values from BEGIN PRIVATE KEY to END PRIVATE KEY, including the header and footer, and paste them in the **RSA Private Key** text box. b Copy the certificate values from BEGIN CERTIFICATE to END CERTIFICATE, including the header and footer, and paste them in the **Certificate Chain** text box. For multiple certificate values, include a BEGIN CERTIFICATE header and END CERTIFICATE footer for each certificate. **Note** In the case of chained certificates, additional attributes may be available. c (Optional) If your certificate uses a pass phrase to encrypt the certificate key, copy the pass phrase and paste it in the **Passphrase** text box.
Provide Certificate Thumbprint	Use this option if you want to provide a certificate thumbprint to use a certificate that is already deployed in the certificate store on the IaaS servers. Using this option will not transmit the certificate from the virtual appliance to the IaaS servers. It enables users to deploy existing certificates on IaaS servers without uploading them in the management interface.

6 Click **Save Settings**.

After a few minutes, the certificate details appear on the page.

7 If required by your network or load balancer, copy the imported or newly created certificate to the load balancer.

8 Open a browser and navigate to `https://`*`managerServiceAdddress`*`/vmpsProvision/` from a server that this running a DEM worker or agent.

   If you are using a load balancer, the host name must be the fully qualified domain name of the load balancer.

9 If prompted, continue past the certificate warnings.

10 Validate that the new certificate is provided and is trusted.

11 If you are using a load balancer, configure and enable any applicable health checks.

Update Embedded vRealize Orchestrator to Trust vRealize Automation Certificates

If you update or change vRealize Automation appliance or IaaS certificates, you must update vRealize Orchestrator to trust the new or updated certificates.

This procedure applies to all vRealize Automation deployments that use an embedded vRealize Orchestrator instance. If you use an external vRealize Orchestrator instance, see Update External vRealize Orchestrator to Trust vRealize Automation Certificates.

Note This procedure resets tenant and group authentication back to the default settings. If you have customized your authentication configuration, note your changes so that you can re-configure authentication after completing the procedure.

See the vRealize Orchestrator documentation for information about updating and replacing vRealize Orchestrator certificates.

In a clustered configuration, you must complete this procedure on the primary vRealize Automation appliance node and then perform a `join-cluster` against the primary from each replica vRealize Automation appliance node.

Note In a cluster, stop the `vco-configurator` service on all replica nodes until the procedure is completed to avoid unwanted automatic control center synchronization.

If you replace or update vRealize Automation certificates without completing this procedure, the vRealize Orchestrator Control Center may be inaccessible, and errors may appear in the `vco-server` and `vco-configurator` log files.

Problems with updating certificates can also occur if vRealize Orchestrator is configured to authenticate against a different tenant and group than vRealize Automation. For information, see VMware Knowledge Base article Exception Untrusted certificate chain after replacing vRA certificate (2147612).

The trust command syntaxes shown herein are representative rather than definitive. While they are appropriate for most typical deployments, there may be situations in which you need to experiment with variations on the commands.

- If you specify `--certificate` you must provide the path to a valid certificate file in PEM format.
- If you specify `--uri`, you must provide the uri from which the command can fetch a trusted certificate.
- If you specify the `--registry-certificate` option, you indicate that the requested certificate should be treated as the certificate for the component registry and the trusted certificate is added to the truststore under a specific alias used by the component registry certificate.

You can also manage certificates by using SSL Trust Manager workflows in vRealize Orchestrator. For information, see the *Manage Orchestrator Certificates* topic in the vRealize Orchestrator documentation.

Procedure

1 Stop the vRealize Orchestrator server and Control Center services.

```
service vco-server stop
service vco-configurator stop
```

2 Reset the vRealize Orchestrator authentication provider by running the following command.

```
/var/lib/vco/tools/configuration-cli/bin/vro-configure.sh reset-authentication
ls -l /etc/vco/app-server/
mv /etc/vco/app-server/vco-registration-id /etc/vco/app-server/vco-registration-id.old
vcac-vami vco-service-reconfigure
```

3 Check the trusted certificate for the vRealize Orchestrator trust store using the command line interface utility located at `/var/lib/vco/tools/configuration-cli/bin` with the following command.

```
/var/lib/vco/tools/configuration-cli/bin/vro-configure.sh list-trust
```

- Check for the certificate with the following alias: vco.cafe.component-registry.ssl.certificate. This should be the vRealize Automation certificate that the vRealize Orchestrator instance uses as an authentication provider.
- This certificate must match the newly configured vRealize Automation certificate. If it does not match, it can be changed as follows:

  1 Copy your vRealize Automation signed appliance certificate PEM file to the `/tmp` folder on the appliance.

  2 Run the following command adding the appropriate certificate path.

```
./vro-configure.sh trust --certificate path-to-the-certificate-file-in-PEM-format--
registry-certificate
```

See the following example command.

```
/var/lib/vco/tools/configuration-cli/bin/vro-configure.sh trust --certificate /var/tmp/
test.pem --registry-certificate
```

4. You may need to run the following commands to trust the certificate.

```
/var/lib/vco/tools/configuration-cli/bin/vro-configure.sh trust --uri https://vra.domain.com

/var/lib/vco/tools/configuration-cli/bin/vro-configure.sh trust --registry-certificate --uri
https://vra.domain.com
```

5. Ensure that the vRealize Automation certificate is now injected into the vRealize Orchestrator trust store using the following command.

```
/var/lib/vco/tools/configuration-cli/bin/vro-configure.sh list-trust
```

6. Start the vRealize Orchestrator server and control center services.

```
service vco-server start
service vco-configurator start
```

What to do next

You can validate that trust has been updated on a clustered system.

1. Log in to the virtual appliance management interface as root.
2. Select the Services page.
3. Ensure that there are no duplicate vco services listed.

   If you see any duplication of the vco services listed, click **Unregister** to remove the services that do not have a state of `Registered`.
4. Ensure that `vco-configurator` is started on all virtual appliance nodes.
5. Log in to the vRealize Orchestrator control center and navigate to the Validate Configuration page to validate the configuration.
6. Navigate to the Authentication Provider page, and verify that the auth settings are correct.

   You can also test the login credentials on this page.

Update External vRealize Orchestrator to Trust vRealize Automation Certificates

If you update or change vRealize Automation appliance or IaaS certificates, you must update vRealize Orchestrator to trust the new or updated certificates.

This procedure applies to vRealize Automation deployments that use an external vRealize Orchestrator instance.

Note This procedure resets tenant and group authentication back to the default settings. If you have customized your authentication configuration, note your changes so that you can re-configure authentication after completing the procedure.

See the vRealize Orchestrator documentation for information about updating and replacing vRealize Orchestrator certificates.

If you replace or update vRealize Automation certificates without completing this procedure, the vRealize Orchestrator Control Center may be inaccessible, and errors may appear in the vco-server and vco-configurator log files.

Problems with updating certificates can also occur if vRealize Orchestrator is configured to authenticate against a different tenant and group than vRealize Automation. See Knowledge Base article 2147612.

Procedure

1 Stop the vRealize Orchestrator server and Control Center services.

```
service vco-configurator stop
```

2 Reset the vRealize Orchestrator authentication provider.

```
/var/lib/vco/tools/configuration-cli/bin/vro-configure.sh reset-authentication
```

3 Start the vRealize Orchestrator Control Center service.

```
service vco-configurator start
```

4 Log in to the Control Center using virtual appliance management interface root credentials.

5 Unregister and re-register the authentication provider.

Updating the vRealize Automation Appliance Management Site Certificate

The system administrator can replace the SSL certificate of the management site service when it expires or to replace a self-signed certificate with one issued by a certificate authority. You secure the management site service on port 5480.

The vRealize Automation appliance uses lighttpd to run its own management site. When you replace a management site certificate, you must also configure all Management Agents to recognize the new certificate.

If you are running a distributed deployment, you can update management agents automatically or manually. If you are running a minimal deployment, you must update the management agent manually.

See Manually Update Management Agent Certificate Recognition for more information.

Procedure

1 Find the Management Agent Identifier

   You use the Management Agent identifier when you create and register a new management site server certificate.

2 Replace the vRealize Automation Appliance Management Site Certificate

   If the SSL certificate of the management site service expires, or you started with a self-signed certificate and site policies require a different one, you can replace the certificate.

3 Update Management Agent Certificate Recognition

   After replacing a vRealize Automation appliance management site certificate, you must update all management agents to recognize the new certificate and to reestablish trusted communications between the virtual appliance management site and management agents on IaaS hosts.

Find the Management Agent Identifier

You use the Management Agent identifier when you create and register a new management site server certificate.

Procedure

1 Open the Management Agent configuration file located at `<`*`vra-installation-dir`*`>\Management Agent\VMware.IaaS.Management.Agent.exe.config`.

2 Record the value from the id attribute of the agentConfiguration element.

```
<agentConfiguration id="0E22046B-9D71-4A2B-BB5D-70817F901B27">
```

Replace the vRealize Automation Appliance Management Site Certificate

If the SSL certificate of the management site service expires, or you started with a self-signed certificate and site policies require a different one, you can replace the certificate.

You are allowed to reuse the certificate used by the vRealize Automation service on port 443, or use a different one. If you are requesting a new CA-issued certificate to update an existing certificate, a best practice is to reuse the Common Name from the existing certificate.

Note The vRealize Automation appliance uses lighttpd to run its own management site. You secure the management site service on port 5480.

Prerequisites

- The certificate must be in PEM format.
- The certificate must include both of the following, in order, together in one file:

   a RSA private key

   b Certificate chain

- The private key cannot be encrypted.
- The default location and file name is `/opt/vmware/etc/lighttpd/server.pem`.

See Extracting Certificates and Private Keys for more information about exporting a certificate and private key from a Java keystore to a PEM file.

Procedure

1. Log in by using the appliance console or SSH.
2. Back up your current certificate file.

   ```
   cp /opt/vmware/etc/lighttpd/server.pem /opt/vmware/etc/lighttpd/server.pem-bak
   ```

3. Copy the new certificate to your appliance by replacing the content of the file `/opt/vmware/etc/lighttpd/server.pem` with the new certificate information.
4. Run the following command to restart the lighttpd server.

   ```
   service vami-lighttp restart
   ```

5. Run the following command to restart the haproxy service.

   ```
   service haproxy restart
   ```

6. Log in to the management console and validate that the certificate is replaced. You might need to restart your browser.

What to do next

Update all management agents to recognize the new certificate.

For distributed deployments, you can update management agents manually or automatically. For minimal installations, you must update agents manually.

- For information about automatic update, see Automatically Update Management Agents in a Distributed Environment to Recognize a vRealize Automation Appliance Management Site Certificate .
- For information about manual update, see Manually Update Management Agent Certificate Recognition .

Update Management Agent Certificate Recognition

After replacing a vRealize Automation appliance management site certificate, you must update all management agents to recognize the new certificate and to reestablish trusted communications between the virtual appliance management site and management agents on IaaS hosts.

Each IaaS host runs a management agent and each management agent must be updated. Minimal deployments must be updated manually, while distributed deployments can be updated manually or by using an automated process.

- Manually Update Management Agent Certificate Recognition

  After replacing a vRealize Automation appliance management site certificate, you must update Management Agents manually to recognize the new certificate to reestablish trusted communications between the virtual appliance management site and Management Agents on IaaS hosts.

- Automatically Update Management Agents in a Distributed Environment to Recognize a vRealize Automation Appliance Management Site Certificate

  After the management site certificate is updated in a high-availability deployment, the management agent configuration must also be updated to recognize the new certificate and reestablish trusted communication.

Manually Update Management Agent Certificate Recognition

After replacing a vRealize Automation appliance management site certificate, you must update Management Agents manually to recognize the new certificate to reestablish trusted communications between the virtual appliance management site and Management Agents on IaaS hosts.

Perform these steps for each Management Agent in your deployment after you replace a certificate for the vRealize Automation appliance management site.

For distributed deployments, you can update Management Agents manually or automatically. For information about automatic update, see Automatically Update Management Agents in a Distributed Environment to Recognize a vRealize Automation Appliance Management Site Certificate .

Prerequisites

Obtain the SHA1 thumbprints of the new vRealize Automation appliance management site certificate.

Procedure

1. Stop the VMware vCloud Automation Center Management Agent service.

2. Navigate to the Management Agent configuration file located at `[`*`vcac_installation_folder`*`]\Management Agent \VMware.IaaS.Management.Agent.exe.Config`, typically `C:\Program Files (x86)\VMware \vCAC\Management Agent\VMware.IaaS.Management.Agent.exe.Config`.

3 Open the file for editing and locate the endpoint configuration setting for the old management site certificate. which you can identify by the endpoint address.

For example:

```
<agentConfiguration id="C816CFBC-4830-4FD2-8951-C17429CEA291" pollingInterval="00:03:00">
  <managementEndpoints>
    <endpoint address="https://vra-va.local:5480"
thumbprint="D1542471C30A9CE694A512C5F0F19E45E6FA32E6" />
  </managementEndpoints>
</agentConfiguration>
```

4 Change the thumbprint to the SHA1 thumbprint of the new certificate.

For example:

```
<agentConfiguration id="C816CFBC-4830-4FD2-8951-C17429CEA291" pollingInterval="00:03:00">
  <managementEndpoints>
    <endpoint address="https://vra-va.local:5480"
thumbprint="8598B073359BAE7597F04D988AD2F083259F1201" />
  </managementEndpoints>
</agentConfiguration>
```

5 Start the VMware vCloud Automation Center Management Agent service.

6 Login to the virtual appliance management site and select the **Cluster** tab.

7 Check the Distributed Deployment Information table to verify that the IaaS server has contacted the virtual appliance recently, which confirms that the update is successful.

Automatically Update Management Agents in a Distributed Environment to Recognize a vRealize Automation Appliance Management Site Certificate

After the management site certificate is updated in a high-availability deployment, the management agent configuration must also be updated to recognize the new certificate and reestablish trusted communication.

You can update vRealize Automation appliance management site certificate information for distributed systems manually or automatically. For information about manually updating management agents, see Manually Update Management Agent Certificate Recognition .

Use this procedure to update the certificate information automatically.

Procedure

1 When Management Agents are running, replace the certificate on a single vRealize Automation appliance management site in your deployment.

2 Wait fifteen minutes for the management agent to synchronize with the new vRealize Automation appliance management site certificate.

3 Replace certificates on other vRealize Automation appliance management sites in your deployment.

   Management agents are automatically updated with the new certificate information.

Replace a Management Agent Certificate

The system administrator can replace the Management Agent certificate when it expires or replace a self-signed certificate with one issued by a certificate authority.

Each IaaS host runs its own Management Agent. Repeat this procedure on each IaaS node whose Management Agent you want to update.

Prerequisites

- Copy the Management Agent identifier in the Node ID column before you remove the record. You use this identifier when you create the new Management Agent certificate and when you register it.
- When you request a new certificate, ensure that the Common Name (CN) attribute in the certificate subject field for the new certificate is typed in the following format:

  ```
  VMware Management Agent 00000000-0000-0000-0000-000000000000
  ```

  Use the string VMware Management Agent, followed by a single space and the GUID for the Management Agent in the numerical format shown.

Procedure

1 Stop the Management Agent service from your Windows Services snap-in.

   a From your Windows machine, click **Start**.

   b In the Windows Start Search box, enter **`services.msc`** and press Enter.

   c Right-click **VMware vCloud Automation Center Management Agent** service and click **Stop** to stop the service.

2 Remove the current certificate from the machine. For information about managing certificates on Windows Server 2008 R2, see the Microsoft Knowledge Base article at http://technet.microsoft.com/en-us/library/cc772354.aspx or the Microsoft wiki article at http://social.technet.microsoft.com/wiki/contents/articles/2167.how-to-use-the-certificates-console.aspx.

   a Open the Microsoft Management Console by entering the command **`mmc.exe`**.

   b Press Ctrl + M to add a new snap-in to the console or select the option from the File drop-down menu.

   c Select **Certificates** and click **Add**.

   d Select **Computer account** and click **Next**.

   e Select **Local computer: (the computer this console is running on)**.

f Click **OK**.

g Expand **Certificates (Local Computer)** on the left side of the console.

h Expand **Personal** and select the Certificates folder.

i Select the current Management Agent certificate and click **Delete**.

j Click **Yes** to confirm the delete action.

3 Import the newly generated certificate into the local `computer.personal` store, or do not import anything if you want the system to auto-generate a new self-signed certificate.

4 Register the Management Agent certificate with the vRealize Automation appliance management site.

a Open a command prompt as an administrator and navigate to the Cafe directory on the machine on which the Management Agent is installed at `<`*`vra-installation-dir`*`>\Management Agent\Tools\Cafe`, typically `C:\Program Files (x86)\VMware\vCAC\Management Agent\Tools\Cafe`.

b Enter the `Vcac-Config.exe RegisterNode` command with options to register the Management Agent identifier and certificate in one step. Include the Management Agent identifier you recorded earlier as the value for the `-nd` option.

Table 10-3. Required Options and Arguments for Vcac-Config.exe RegisterNode

Option	Argument	Notes
`-vamih`	"*vra-va-hostname.domain.name*:5480"	The URL of the management site host, including a port specification.
`-cu`	"root"	The user name, which must be the root user.
`-cp`	"*password*"	Password for the root user as a quoted string.
`-hn`	"*machine-hostname.domain.name*"	The machine name of the Management Agent host, including domain information. This value must match the hostname that the current node is registered with in the vRealize Automation appliance. Can be seen with option 1 specified above for the node ID or in the VAMI - Distributed Deployment Information table. If it is not the same value, the following error is returned when the command is run: Failure: Cannot add duplicate node id 00000000-0000-0000-0000-000000000000.
`-nd`	"*00000000-0000-0000-0000-000000000000*"	Management Agent identifier.
`-tp`	"*0000000000000000000000000000000000000000*	Thumbprint of the SSL certificate of the management site host, as defined in the *-vamih* parameter.

The following example shows the command format:

```
Vcac-Config.exe RegisterNode -v -vamih "vra-va-hostname.domain.name:5480"
-cu "root" -cp "password" -hn "machine-hostname.domain.name"
-nd "00000000-0000-0000-0000-000000000000"
-tp "0000000000000000000000000000000000000000"
```

5 Restart the Management Agent.

Example: Command to Register a Management Agent Certificate

```
Vcac-Config.exe RegisterNode -v -vamih "vra-va.eng.mycompany:5480" -cu "root" -cp
"secret" -hn "iaas.eng.mycompany" -nd "C816CFBX-4830-4FD2-8951-C17429CEA291" -tp
"70928851D5B72B206E4B1CF9F6ED953EE1103DED"
```

Change the Polling Method for Certificates

If there are commas in the OU section of the IaaS certificate, you might encounter STOMP WebSocket errors in the Manager Service log files. In addition, virtual machine provisioning might fail. You can remove the commas, or change the polling method from WebSocket to HTTP.

To change the polling method, take the following steps.

Procedure

1 Open the following file in a text editor.

   `C:\\:Program FIles (x86)\VMware\vCAC\Server\Manager Service.exe.config`.

2 Add the following lines inside the `<appSettings>` section.

```
<add key="Extensibility.Client.RetrievalMethod" value="Polling"/>
<add key="Extensibility.Client.PollingInterval" value="2000"/>
<add key="Extensibility.Client.PollingMaxEvents" value="128"/>
```

3 Save and close `Manager Service.exe.config`.

4 Restart the Manager Service.

Results

For more information about the Manager Service, see Infrastructure as a Service.

Managing the vRealize Automation Postgres Appliance Database

vRealize Automation requires the appliance database for system operation. You can manage the appliance database through the vRealize Automation Appliance Virtual Appliance Management Interface.

Note This information applies only to deployments that use an embedded appliance database. It does not apply to deployments that use an external Postgres database.

You can configure the database as a single node or with multiple nodes to facilitate high availability through failover. The vRealize Automation installer includes a database node on each vRealize Automation appliance installation. So if you install three instances of a vRealize Automation appliance, you have three database nodes. Automatic failover is implemented on applicable deployments. The appliance database requires no maintenance unless a machine configuration changes or, if you use a clustered configuration, you promote a different node for the primary.

Note The database clustered configuration is set up automatically when you join a virtual appliance to the cluster using the Join cluster operation. The database cluster is not directly dependent upon the virtual appliance cluster. For instance, a virtual machine joined to a cluster can operate normally even if the embedded appliance database is not started or has failed.

For high availability, vRealize Automation uses the PostgresSQLprimary-replica model to support data replication. This means that all of the database nodes work in a cluster with one leading node, known as the primary, and several replicating nodes, known as replicas. The primary node handles all database requests and the replica nodes stream and replay transactions from the primary locally.

A clustered configuration contains one primary node and one or more replica nodes. The primary node is the vRealize Automation appliance node with the primary database that supports system functionality. Replica nodes contain copies of the database that can be pulled into service if the primary node fails.

Several high availability appliance database options exist. Selecting the replication mode is the most important database configuration option. The replication mode determines how your vRealize Automation deployment maintains data integrity and, for high availability configurations, how it fails over if the primary or primary node fail. There are two available replication modes: synchronous and asynchronous.

Both replication modes support database failover, though each has advantages and disadvantages. To support high availability database failover, asynchronous mode requires two nodes, whereas synchronous mode requires three nodes. Synchronous mode also invokes automatic failover.

Replication Mode	Advantages	Disadvantages
Synchronous	■ Minimizes chance of data loss. ■ Invokes automatic faiiover.	■ Might affect system performance. ■ Requires three nodes.
Asynchronous	■ Requires only two nodes. ■ Affects system performance less than synchronous mode.	Not as robust as synchronous mode in preventing data loss.

vRealize Automation supports both modes, but operates in asynchronous mode by default and provides high availability only if there are at least two appliance database nodes. The **Cluster** tab on the Virtual Appliance Management Interface enables you to switch synchronization modes and to add database nodes as needed.

When operating in synchronous mode, vRealize Automation invokes automatic failover.

If you begin with one node in a non-high-availability configuration, you can add nodes later as required to enhance high availability. If you have the appropriate hardware and require maximum protection against data loss, consider configuring your deployment to operate in synchronous mode.

Appliance Database Failover

In a high availability configuration, the primary constantly streams transactions to the replica servers. If the primary fails, the active and working replica is ready to proceed with read-only requests. When the new primary is promoted, either manually or automatically, all of the upcoming requests are moved to it.

Configure the Appliance Database

You can use the Virtual Appliance Management Interface Database page to monitor or update the configuration of the appliance database. You can also use it to change the primary node designation and the synchronization mode used by the database.

The appliance database is installed and configured during vRealize Automation system installation and configuration, but you can monitor and change the configuration from the **Database** tab on the Virtual Appliance Management Interface.

The **Connection Status** text box indicates whether the database is connected to the vRealize Automation system and is functioning correctly.

If your appliance database uses multiple nodes to support failover, the table at the bottom of the page displays the nodes, and their status and indicates which node is the primary. The **Replication mode** text box shows the currently configured operation mode for the system, either synchronous or asynchronous. Use this page to update appliance database configuration.

The Sync State* column in the database nodes table shows the synchronization method for the cluster. This column works with the Status column to show the state of cluster nodes. Potential status differs depending on whether the cluster uses asynchronous or synchronous replication.

Table 10-4. Sync State for Appliance Database Replication Modes

Mode	Sync State Message
Synchronous replication	Primary node - no status Replica node - sync Other nodes - potential
Asynchronous replication	Primary node - no status Other nodes - potential

The Valid column indicates whether replicas are synchronized with the primary node. The primary node is always valid.

The Priority column shows the position of replica nodes in relation to the primary node. The primary node has no priority value. When promoting a replica to become the primary, select the node with the lowest priority value.

When operating in synchronous mode, vRealize Automation invokes automatic failover. In the event of primary node failure the next available replica node will automatically become the new primary. The failover operation requires 10 to 30 seconds on a typical vRealize Automation deployment.

Prerequisites

- Install and configure vRealize Automation according to appropriate instructions in Installing vRealize Automation.
- Log in to vRealize Automation Appliance Management as **root** using the password you entered when you deployed the vRealize Automation appliance.
- Configure an appropriate embedded Postgres appliance database cluster as part of your vRealize Automation deployment.

Procedure

1 On the Virtual Appliance Management Interface, select **vRA Settings > Database**.

2 If your database uses multiple nodes, review the table at the bottom of the page and ensure that the system is operating appropriately.

   - Ensure that all nodes are listed.
   - Ensure that the appropriate node is the designated primary node.

   Note Do not click **Sync Mode** to change the synchronization mode of the database unless you are certain that your data is secure. Changing the sync mode without preparation may cause data loss.

3 To promote one of the nodes to be the primary, click **Promote** in the appropriate column.

4 Click **Save Settings** to save your configuration if you have made any changes.

Three Node Appliance Database Automatic Failover Scenarios

There are several appliance database high availability failover scenarios, and vRealize Automation behavior varies depending on appliance database configuration and the number of nodes that fail.

Single Node Failure Scenarios

If one of the three nodes fails, vRealize Automation will initiate an auto failover. No additional auto failover operations can occur until all three nodes are restored.

The following table describes behavior and actions related to a primary node failure in a high availability deployment.

Table 10-5. The Primary Node Fails

Expected Behavior	■ The configured sync replica node becomes the primary and automatically picks up appliance database functionality. ■ The potential sync replica becomes the sync standby node. ■ The vRealize Automation deployment functions in read only mode until the automatic failover completes.
Further Action	■ When the former primary is recovered, it will be reset as replica automatically by the failover agent repair logic. No manual action is required. ■ If the former primary cannot be recovered, manually set the appliance database to asynchronous mode.

The following table describes behavior and actions related to a sync replica node failure in a high availability deployment.

Table 10-6. The Sync Replica Fails

Expected Behavior	■ The vRealize Automation deployment experiences no downtime. There will be a delay of a couple of seconds for database requests until the potential replica becomes the new sync replica. The appliance database performs this action automatically.
Further Action	■ When the former synch replica comes online, it will become a potential replica automatically. No manual action is required. ■ If the former sync replica cannot be repaired, manually set the appliance database to asynchronous mode.

The following table describes behavior and actions related to a primary node failure in a high availability deployment.

Table 10-7. The Potential Replica Fails

Expected Behavior	No deployment downtime.
Further Action	■ When the former potential replica comes online, it becomes a potential replica automatically. No manual action is required. ■ If the former potential replica cannot be repaired, set the appliance database to asynchronous mode.

Two Node Failure Scenarios

If two out of the three nodes fail simultaneously, vRealize Automation switches to read only mode until a manual repair is performed.

The following table describes behavior and actions related to a primary node and potential replica node failure in a high availability deployment.

Table 10-8. The Primary Node and Potential Replica Fail

Expected Behavior	- The sync replica is not promoted to primary automatically. vRealize Automation functions in read only mode as it is able to process read-only transactions until a manual promotion is performed.
Further Action	- Manual promotion is required. Set the appliance database to asynchronous mode. - When the primary and potential replica are recovered, manually set them to synchronize against the new primary. At that point, you can switch vRealize Automation back to synchronous mode. - When two out of three nodes are down simultaneously, vRealize Automation will switch to read-only mode until you effect a manual repair. If only one database node is available, switch your deployment to asynchronous mode.

The following table describes behavior and actions related to Sync and Potential node failure in a high availability deployment.

Table 10-9. The Sync and Potential Replicas Fail

Expected Behavior	- vRealize Automation functions in read only mode as it is able to process read-only transactions until a manual repair is performed.
Further Action	- Manual promotion is required. Set the appliance database to asynchronous mode. - When the sync and potential replicas are recovered, they should be manually reset to synchronize against the primary. At this point, you can switch vRealize Automation back to synchronous mode. - When two out of three nodes are down simultaneously, vRealize Automation will switch to read-only mode until you effect a manual repair. If only one database node is available, switch your deployment to asynchronous mode.

Links Failures Among Nodes

If a link failure occurs among nodes on a distributed deployment, the automatic failover agent attempts to repair the configuration.

The following table describes behavior and actions related to a link failure between two sites in a high availability deployment with the specified configuration when all nodes remain up and online.

Site A: Primary and potential replica

Site B: Sync replica

Table 10-10. Link Failure Between Two Sites when all Nodes Remain Up and Online

Expected Behavior	No downtime for the vRealize Automation deployment. The potential replica automatically becomes the sync replica.
Further Action	No manual action is required.

The following table describes behavior and actions related to a link failure between two sites in a high availability deployment with the specified configuration when all nodes remain up and online.

Site A: Primary

Site B: Sync and potential replica

Table 10-11. Link Failure Between Two Sites when all Nodes Remain Up and Online - Alternate Configuration

Expected Behavior	Sync replica becomes the primary and automatically picks up appliance database functionality. Automatic failover agent promotes the potential replica to become the new sync replica. vRealize Automation deployment operates in read only mode until this promotion completes.
Further Action	No manual action is required. When the link is recovered, the automatic failover agent resets the former primary as replica.

Scenario: Perform Manual vRealize Automation Appliance Database Failover

When there is a problem with the vRealize Automation appliance Postgres database, you manually fail over to a replica vRealize Automation appliance node in the cluster.

Follow these steps when the Postgres database on the primary vRealize Automation appliance node fails or stops running.

Note Once a node goes into a unhealthy state, do not attempt to use its virtual appliance management interface for any operations including failover.

Prerequisites

- Configure a cluster of vRealize Automation appliance nodes. Each node hosts a copy of the embedded Postgres appliance database.

Procedure

1 Remove the primary node IP address from the external load balancer.

2 Log in to the vRealize Automation appliance management interface as root.

https://*vrealize-automation-appliance-FQDN*:5480

3 Select **Cluster**.

4 From the list of database nodes, locate the replica node with the lowest priority.

Replica nodes appear in ascending priority order.

5 Click **Promote** and wait for the operation to finish.

When finished, the replica node is listed as the new primary node.

6 Correct issues with the former primary node and add it back to the cluster:

a Isolate the former primary node.

Disconnect the node from its current network, the one that is routing to the remaining vRealize Automation appliance nodes. Select another NIC for management, or manage it directly from the virtual machine management console.

b Recover the former primary node.

Power the node on or otherwise correct the issue. For example, you might reset the virtual machine if it is unresponsive.

c From a console session as root, stop the vpostgres service.

```
service vpostgres stop
```

d Add the former primary node back to its original network, the one that is routing to the other vRealize Automation appliance nodes.

e From a console session as root, restart the haproxy service.

```
service haproxy restart
```

f Log in to the new vRealize Automation appliance primary node management interface as root.

g Select **Cluster**.

h Locate the former primary node, and click **Reset**.

i After a successful reset, restart the former primary node.

j With the former primary powered on, verify that the following services are running.

```
haproxy
horizon-workspace
rabbitmq-server
vami-lighttp
vcac-server
vco-server
```

k Re-add the former primary node to the external load balancer.

Note If a primary node that was demoted to replica is still listed as primary, you might need to manually re-join it to the cluster to correct the problem.

Scenario: Perform a Maintenance Database Failover

As a vRealize Automation system administrator, you must perform an appliance database maintenance failover operation.

This scenario assumes that the current primary node is up and running normally. There are two database failover maintenance steps: maintenance of the primary and maintenance of a replica node. When a primary node has been replaced so that it becomes a replica, you should perform maintenance on it so that it is suitable to become the primary again should the need arise.

Note Do not stop or restart the HAProxy service on the applicable host machine while performing a maintenance failover.

Prerequisites

- vRealize Automation is installed and configured according to appropriate instructions in Installing vRealize Automation.
- Log in to vRealize Automation Appliance Management as **root** using the password you entered when you deployed the vRealize Automation appliance.
- Install and configure an appropriate embedded Postgres appliance database cluster.
- If your database uses synchronous replication mode, ensure that there are three active nodes in the cluster.

Procedure

1. Remove the primary node IP address from the external load balancer.
2. Isolate the primary node.

   Disconnect the node from its current network. This should be the network that is routing to the remaining vRealize Automation appliance nodes.
3. Select another NIC for management, or manage it directly from the Virtual Appliance Management Interface.
4. Select **Cluster** on the Virtual Appliance Management Interface.
5. Select the replica node with the lowest priority for promotion to the primary, and click **Promote**.

   Replica nodes appear in ascending priority order.

   The old primary is demoted to replica status, and the new primary is promoted.
6. Perform the appropriate replica maintenance.
7. When the maintenance is complete, ensure that the virtual appliance is running with network connectivity and that its HAProxy service is running.

   a. Log in to the vRealize Automation management console as **root**.

   b. Ensure that the replica node can be pinged, resolved by name, and has a recent status in the Virtual Appliance Management Interface **Cluster** tab.

8 Click **Reset** for the replica node.

   This operation resets the database so that it is configured to replicate to the current primary and re-synchronizes the replica node with the latest haproxy configuration from the primary node.

9 Following successful reset, return the replica virtual appliance node IP address to the external virtual appliance load balancer IP address pool.

10 Ensure that the replica node appears healthy on the database table and that it can be pinged and resolved by name.

What to do next

Correct issues with the former primary node and add it back to the cluster.

Manually Recover Appliance Database from Catastrophic Failure

If the appliance database fails, and no database nodes are up and running or all replica nodes are out of sync when the primary fails, use the following procedure to attempt to recover the database.

This procedure applies to situations in which no database nodes are operational across a cluster that is running in asynchronous mode. In this scenario, you typically see errors similar to the following on the Virtual Appliance Management Interface page when trying to load or refresh the page:

```
Error initializing the database service: Could not open JDBC Connection for transaction; nested
exception is org.postgresql.util.PSQLException: The connection attempt failed.
```

Procedure

1 Try to recover the database using the Virtual Appliance Management Interface from one of the database nodes.

   a If possible, open the Virtual Appliance Management Interface **Cluster** page of the node with the most recent state. Typically, this node is the one that was the primary node before the database failed.

   b If the Virtual Appliance Management Interface for the primary node fails to open, try to open the Interface for other replica nodes.

   c If you can find a database node with a working Virtual Appliance Management Interface, try to recover it by performing a manual failover.

      See Scenario: Perform Manual vRealize Automation Appliance Database Failover.

2 If the procedure in step 1 fails, start a shell session and try to determine the node with the most recent state. Start a shell session to all the available cluster nodes and try to start their databases by running the following shell command: `service vpostgres start`

3. Use the following procedure for each node that has a running local database to determine the node with the most recent state.

   a. Run the following command to determine the node with the most recent state. If the command returns `f`, then it is the node with most recent state and you can proceed to step 4.

      ```
      su - postgres
      psql vcac
      vcac=# select pg_is_in_recovery();
       pg_is_in_recovery
      ```

      - If this command returns an `f`, then this node has the most recent state.
      - If the node returns a `t`, run the following command on the node:

      ```
      SELECT pg_last_xlog_receive_location() as receive_loc, pg_last_xlog_replay_location() as
      replay_loc, extract(epoch from pg_last_xact_replay_timestamp()) as replay_timestamp;
      ```

      This command should return a result similar to the following.

      ```
      vcac=# SELECT pg_last_xlog_receive_location() as receive_loc, pg_last_xlog_replay_location()
      as replay_loc, extract(epoch from pg_last_xact_replay_timestamp()) as replay_timestamp;
       receive_loc | replay_loc | replay_timestamp
      -------------+------------+------------------
       0/20000000 | 0/203228A0 | 1491577215.68858
      (1 row)
      ```

4. Compare the results for each node to determine which one has the most recent state.

   Select the node with greatest value under the `receive_loc` column. If equal, select the greatest from the `replay_loc` column and then, if again equal, select the node with greatest value of `replay_timestamp`.

5. Run the following command on the node with the most recent state: `vcac-vami psql-promote-master -force`

6. Open the `/etc/haproxy/conf.d/10-psql.cfg` file in a text editor and update the following line.

   ```
   server masterserver sc-rdops-vm06-dhcp-170-156.eng.vmware.com:5432 check on-marked-up shutdown-
   backup-sessions
   ```

   To read as follows with the current node FQDN:

   ```
   server masterserver current-node-fqdn:5432 check on-marked-up shutdown-backup-sessions
   ```

7. Save the file.

8. Run the `service haproxy restart` command.

9 Open the Virtual Appliance Management Interface **Cluster** page for the most recent node.

This node should appear as the primary node with the other nodes as invalid replicas. In addition, the **Reset** button for the replicas is enabled.

10 Click **Reset** and for each replica in succession until the cluster state is repaired.

Backup and Recovery for vRealize Automation Installations

To minimize system downtime and data loss in the event of failures, administrators back up the entire vRealize Automation installation on a regular basis. If your system fails, you can recover by restoring the last known working backup and reinstalling some components.

To back up and restore vRealize Automation, see the following topics in the vRealize Suite documentation:

- vRealize Automation Preparations for Backing Up
- vRealize Automation System Recovery

The Customer Experience Improvement Program

This product participates in VMware's Customer Experience Improvement Program (CEIP). The CEIP provides VMware with information that enables VMware to improve its products and services, to fix problems, and to advise you on how best to deploy and use our products. You can choose to join or leave the CEIP for vRealize Automation at any time.

Details regarding the data collected through CEIP and the purposes for which it is used by VMware are set forth at the Trust & Assurance Center at http://www.vmware.com/trustvmware/ceip.html.

Join or Leave the Customer Experience Improvement Program for vRealize Automation

You can join or leave the Customer Experience Improvement Program (CEIP) for vRealize Automation at any time.

vRealize Automation gives you the opportunity to join the Customer Experience Improvement Program (CEIP) when you initially install and configure the product. After installation, you can join or leave the CEIP by following these steps.

Procedure

1 Log in as root to the vRealize Automation appliance management interface.

https://*vrealize-automation-appliance-FQDN*:5480

2 Click the **Telemetry** tab.

3 Check or uncheck the **Join the VMware Customer Experience Improvement Program** option.

When checked, the option activates the Program and sends data to https://vmware.com.

4 Click **Save Settings**.

Configure Data Collection Time

You can set the day and time when the Customer Experience Improvement Program (CEIP) sends data to VMware.

Procedure

1 Log in to a console session on the vRealize Automation appliance as root.

2 Open the following file in a text editor.

```
/etc/telemetry/telemetry-collector-vami.properties
```

3 Edit the properties for day of week (dow) and hour of day (hod).

Property	Description
`frequency.dow=<day-of-week>`	Day when data collection occurs.
`frequency.hod=<hour-of-day>`	Local time of day when data collection occurs. Possible values are 0–23.

4 Save and close `telemetry-collector-vami.properties`.

5 Apply the settings by entering the following command.

```
vcac-config telemetry-config-update --update-info
```

Changes are applied to all nodes in your deployment.

Adjusting System Settings

As a system administrator, you adjust logging and customize IaaS email templates. You can also manage settings that appear as defaults for each tenant, such as email servers to handle notifications. Tenant administrators can choose to override these defaults if their tenant requires different settings.

Modify the All Services Icon in the Service Catalog

You can modify the default icon in the service catalog to display a custom image. When you modify the icon, it changes for all tenants. You cannot configure tenant-specific icons for the catalog.

Commands are provided for Linux or Mac and Windows so that you can run the cURL commands on any of those operating systems.

Prerequisites

- Convert the image to a base64 encoded string.
- cURL must be installed on the machine where you run the commands.

- You must have the credentials for a vRealize Automation user with the system administrator role.

Procedure

1 Set the VCAC variable in the terminal session for the cURL commands.

Operating System	Command
Linux/Mac	`export VCAC=<VA URL>`
Windows	`set VCAC=<VA URL>`

2 Retrieve the authentication token for the system administrator user.

Operating System	Command
Linux/Mac	`curl https://$VCAC/identity/api/tokens --insecure -H "Accept: application/json" -H 'Content-Type: application/json' --data '{"username":"<Catalog Administrator User>","password":"<password>","tenant":"vsphere.local"}'`
Windows	`curl https://%VCAC%/identity/api/tokens --insecure -H "Accept:application/json" -H "Content-Type:application/json" --data "{\"username\":\"<Catalog Administrator User>\",\"password\":\"<password>\",\"tenant\":\"vsphere.local\"}"`

An authentication token is generated.

3 Set the authentication token variable by replacing <Auth Token> with the token string you generated in the previous step.

Operating System	Command
Linux/Mac	`export AUTH="Bearer <Auth Token>"`
Windows	`set AUTH=Bearer <Auth Token>`

4 Add the base64 encoded string for the image.

Operating System	Command
Linux/Mac	`curl https://$VCAC/catalog-service/api/icons --insecure -H "Accept: application/json" -H 'Content-Type: application/json' -H "Authorization: $AUTH" --data '{"id":"cafe_default_icon_genericAllServices","fileName":"<filename>","contentType":"image/png","image":"<IMAGE DATA as base64 string>"}'`
Windows	`curl https://%VCAC%/catalog-service/api/icons --insecure -H "Accept: application/json" -H "Content-Type: application/json" -H "Authorization: %AUTH%" --data "{\"id\":\"cafe_default_icon_genericAllServices\",\"fileName\":\"<filename>\",\"contentType\":\"image/png\",\"image\":\"<IMAGE DATA as base64 string>\"}"`

Results

The new services icon appears in the service catalog after approximately five minutes.

If you want to revert to the default icon, you can run the following command after you follow steps 1-3..

Operating System	Command
Linux/Mac	`curl https://$VCAC/catalog-service/api/icons/cafe_default_icon_genericAllServices --insecure -H "Authorization: $AUTH" --request DELETE`
Windows	`curl https://%VCAC%/catalog-service/api/icons/cafe_default_icon_genericAllServices --insecure -H "Authorization: %AUTH%" --request DELETE`

Customize Data Rollover Settings

You can configure vRealize Automation data rollover settings to control how your system retains, archives, and deletes legacy data.

Use the data rollover feature to enable rollover, set the maximum number of days for vRealize Automation to retain data in the IaaS SQL Server database before archiving or deleting it, and other data rollover controls.

By default, the data rollover feature is deactivated.

Configure data rollover settings on the vRealize Automation **Global Settings** page. When activated, this feature queries and removes data from the following SQL Server database tables:

- UserLog
- Audit
- CategoryLog
- VirtualMachineHistory
- VirtualMachineHistoryProp
- AuditLogItems
- AuditLogItemsProperties
- TrackingLogItems
- WorkflowHistoryInstances
- WorkflowHistoryResults

If you set `DataRolloverIsArchiveEnabled` to True, archive versions of the tables are created in the dbo schema. For example, the archive version of `UserLog` would be `UserLogArchive` and the archive version of `VirtualMachineHistory` would be `VirtualMachineHistoryArchive`.

When enabled, the data rollover feature runs once a day at a predetermined time of 3 AM according to the vRealize Automation appliance time zone configuration. Using the `DataRollover MaximumAgeInDays` setting, you can set the maximum number of days that you want to retain the data. Note that this process generally runs quickly within a few minutes to an hour. However, when this feature is first turned on the process may have a lot of data to archive/delete to catch up and thus could take much longer to complete. This process is designed to run until done. It performs its work in small and quick batch sized transactional chunks of work as to not cause concurrency issues. Note that this process can be gracefully stopped as described below.

Note You can stop the DataRollover process by changing the `DataRollover Status` setting of Running to Disabled or Enabled. This causes the currently running process to quit gracefully. No work is lost. All data archived or deleted up to the point of stopping the process is saved.

If `DataRollover IsArchiveEnabled` is set to True, data older than that specified in the `DataRollover MaximumAgeInDays` setting is moved to the archive tables. If `DataRollover IsArchiveEnabled` is set to False, data is permanently deleted and no data archiving occurs. Deleted data is not recoverable.

Procedure

1. Log in to the vRealize Automation console as a **system administrator**.
2. Select **Infrastructure > Administration > Global Settings**.
3. On the **Global Settings** page, locate the **Data Rollover** section of the table and review and configure settings.

Setting	Description
`DataRollover BatchSize`	This is defaulted to 2000 and probably does not need to be changed. However, if there seem to be some performance impacts, then a smaller BatchSize may help. A larger BatchSize may get the job done faster, but will put more pressure on concurrent processing. Valid range is 100 to 20000.
`DataRollover IsArchiveEnabled`	Specifies whether to move rollover data to archive tables after the maximum number of days is reached. By default this value is set to True. If you set this value to False, all data older than that specified in the `DataRollover MaximumAgeInDays` setting is permanently deleted.
`DataRollover MaximumAgeInDays`	Specifies the maximum number of days that the system retains data in the database before moving it to archive or permanently deleting it. By default this value is set to 90 days.
`DataRollover Status`	Specifies whether to enable data rollover. By default this value is set to Disabled. To enable data rollover, set the value to Enabled.

Setting	Description
`DataRollover VirtualMachineHistory BatchSize`	Specifies batch size in the `VirtualMachineHistory` table in the range of 1 - 5 records. The default is 1.
`DataRollover UpdateStatistics`	The UpdateStatistics is off by default, but is highly recommended to be turned on (set to 1) as updated statistics is good for query performance. This causes the [dbo].[usp_DataRollover] stored procedure to perform update statistics command on the tables after the archival process has run.

4 Click the **Edit** icon (✎) in the first table column to edit a setting.

The **Value** area for the applicable setting becomes editable.

5 Click the **Save** icon (✓) in the first table column to save your changes.

Adjusting Settings in the Manager Service Configuration File

You can use the manager service configuration file (`managerService.exe.config`) to adjust common settings for machine deployments.

The `managerService.exe.config` file is typically located in the `%System–Drive%\Program Files x86\VMware\vCAC\Server` directory. You should always make a copy of the file before editing it.

You can use the following `managerService.exe.config` file settings to control various aspects of machine deployments. Default values are shown.

- `<add key="ProcessLeaseWorkflowTimerCallbackIntervalMilliseconds" value="3600000"/>`
- `<add key="BulkRequestWorkflowTimerCallbackMilliseconds" value="10000"/>`
- `<add key="MachineRequestTimerCallbackMilliseconds" value="10000"/>`
- `<add key="MachineWorkflowCreationTimerCallbackMilliseconds" value="10000"/>`
- `<add key="RepositoryConnectionMaxRetryCount" value="100"/>`
- `<add key="MachineCatalogRegistrationRetryTimerCallbackMilliseconds" value="120000"/>`
- `<add key="MachineCatalogUnregistrationRetryTimerCallbackMilliseconds" value="120000"/>`
- `<add key="MachineCatalogUpdateMaxRetryCount" value="15"/>`

Setting Resource-Intensive Concurrency Limits

To conserve resources, vRealize Automation limits the number of concurrently running instances of machine provisioning and data collection. You can change the limits.

Configuring Concurrent Machine Provisioning

Multiple concurrent requests for machine provisioning can impact the performance of vRealize Automation. You can make some changes to limits placed on proxy agents and workflow activities to alter performance.

Depending on the needs of machine owners at your site, the vRealize Automation server may receive multiple concurrent requests for machine provisioning. This can happen under the following circumstances:

- A single user submits a request for multiple machines
- Many users request machines at the same time
- One or more group managers approve multiple pending machine requests in close succession

The time required for vRealize Automation to provision a machine generally increases with larger numbers of concurrent requests. The increase in provisioning time depends on three important factors:

- The effect on performance of concurrent resource-intensive vRealize Automation workflow activities, including the SetupOS activity (for machines created within the virtualization platform, as in WIM-based provisioning) and the Clone activity (for machines cloned within the virtualization platform).
- The configured vRealize Automation limit on the number of resource-intensive (typically lengthy) provisioning activities that can be executed concurrently. By default this is eight. Concurrent activities beyond the configured limit are queued.
- Any limit within the virtualization platform or cloud service account on the number of vRealize Automation work items (resource-intensive or not) that can be executed concurrently. For example, the default limit in vCenter Server is four, with work items beyond this limit being queued.

By default, vRealize Automation limits concurrent virtual provisioning activities for hypervisors that use proxy agents to eight per endpoint. This ensures that the virtualization platform managed by a particular agent never receives enough resource-intensive work items to prevent execution of other items. Plan to carefully test the effects of changing the limit before making any changes. Determining the best limit for your site may require that you investigate work item execution within the virtualization platform as well as workflow activity execution within vRealize Automation.

If you do increase the configured vRealize Automation per-agent limit, you may have to make additional configuration adjustments in vRealize Automation, as follows:

- The default execution timeout intervals for the SetupOS and Clone workflow activities are two hours for each. If the time required to execute one of these activities exceeds this limit, the activity is cancelled and provisioning fails. To prevent this failure, increase one or both of these execution timeout intervals.
- The default delivery timeout intervals for the SetupOS and Clone workflow activities are 20 hours for each. Once one of these activities is initiated, if the machine resulting from the activity has not been provisioned within 20 hours, the activity is cancelled and provisioning fails. Therefore, if you have increased the limit to the point at which this sometimes occurs, you will want to increase one or both of these delivery timeout intervals.

Configuring Concurrent Data Collections

By default, vRealize Automation limits concurrent data collection activities. If you change this limit, you can avoid unnecessary timeouts by changing the default execution timeout intervals for the different types of data collection.

vRealize Automation regularly collects data from known virtualization compute resources through its proxy agents and from cloud service accounts and physical machines through the endpoints that represent them. Depending on the number of virtualization compute resources, agents, and endpoints in your site, concurrent data collection operations may occur frequently.

Data collection running time depends on the number of objects on endpoints including virtual machines, datastores, templates, and compute resources. Depending on many conditions, a single data collection can require a significant amount of time. As with machine provisioning, concurrency increases the time required to complete data collection.

By default, concurrent data collection activities are limited to two per agent, with those over the limit being queued. This ensures that each data collection completes relatively quickly and that concurrent data collection activities are unlikely to affect IaaS performance.

Depending on the resources and circumstances at your site, however, it may be possible to raise the configured limit while maintaining fast enough performance to take advantage of concurrency in proxy data collection. Although raising the limit can increase the time required for a single data collection, this might be outweighed by the ability to collect more information from more compute resources and machines at one time.

If you do increase the configured per-agent limit, you might have to adjust the default execution timeout intervals for the different types of data collection that use a proxy agent—inventory, performance, state, and WMI. If the time required to execute one of these activities exceeds the configured timeout intervals, the activity is canceled and restarted. To prevent cancellation of the activity, increase one or more of these execution timeout intervals.

Adjust Concurrency Limits and Timeout Intervals

You can change the per-agent limits on concurrent provisioning, data collection activities, and the default timeout intervals.

When typing a time value for these variables, use the format hh:mm:ss (hh=hours, mm=minutes, and ss=seconds).

Prerequisites

Log in as an administrator to the server hosting the IaaS Manager Service. For distributed installations, this is the server on which the Manager Service was installed.

Procedure

1 Open the `ManagerService.exe.config` file in an editor. The file is located in the vRealize Automation server install directory, typically `%SystemDrive%\Program Files x86\VMware\vCAC\Server`.

2 Locate the section called `workflowTimeoutConfigurationSection`.

3 Update the following variables, as required.

Parameter	Description
MaxOutstandingResourceIntensiveWorkItems	Concurrent provisioning limit (default is 8)
CloneExecutionTimeout	Virtual provisioning execution timeout interval
SetupOSExecutionTimeout	Virtual provisioning execution timeout interval
CloneTimeout	Virtual provisioning clone delivery timeout interval
SetupOSTimeout	Virtual provisioning setup OS delivery timeout interval
CloudInitializeProvisioning	Cloud provisioning initialization timeout interval
MaxOutstandingDataCollectionWorkItems	Concurrent data collection limit
InventoryTimeout	Inventory data collection execution timeout interval
PerformanceTimeout	Performance data collection execution timeout interval
StateTimeout	State data collection execution timeout interval

4 Save and close the file.

5 Select **Start > Administrative Tools > Services**.

6 Stop and then restart the vRealize Automation service.

7 (Optional) If vRealize Automation is running in High Availability mode, any changes made to the `ManagerService.exe.config` file after installation must be made on both the primary and failover servers.

Adjust Execution Frequency of Machine Callbacks

You can change the frequency of several callback procedures, including the frequency that the vRealize Automation callback procedure is run for changed machine leases.

vRealize Automation uses a configured time interval to run different callback procedures on the Model Manager service, such as *ProcessLeaseWorkflowTimerCallbackIntervalMiliSeconds* which searches for machines whose leases have changed. You can change these time intervals to check more or less frequently.

When entering a time value for these variables, enter a value in milliseconds. For example, 10000 milliseconds = 10 seconds and 3600000 milliseconds = 60 minutes = 1 hour.

Prerequisites

Log in as an administrator to the server hosting the IaaS Manager Service. For distributed installations, this is the server on which the Manager Service was installed.

Procedure

1 Open the `ManagerService.exe.config` file in an editor. The file is located in the vRealize Automation server install directory, typically `%SystemDrive%\Program Files x86\VMware\vCAC\Server`.

2 Update the following variables, as desired.

Parameter	Description
RepositoryWorkflowTimerCallbackMiliSeconds	Checks the repository service, or Model Manager Web Service, for activity. Default value is 10000.
ProcessLeaseWorkflowTimerCallbackIntervalMiliSeconds	Checks for expired machine leases. Default value is 3600000.
BulkRequestWorkflowTimerCallbackMiliSeconds	Checks for bulk requests. Default value is 10000.
MachineRequestTimerCallbackMiliSeconds	Checks for machine requests. Default value is 10000.
MachineWorkflowCreationTimerCallbackMiliSeconds	Checks for new machines. Default value is 10000.

3 Save and close the file.

4 Select **Start > Administrative Tools > Services**.

5 Stop and then restart the vCloud Automation Center service.

6 (Optional) If vRealize Automation is running in High Availability mode, any changes made to the `ManagerService.exe.config` file after installation must be made on both the primary and failover servers.

Adjust IaaS Log Settings

You can adjust vRealize Automation to log only the information you want to see in the Manager Service log.

If vRealize Automation is running in high availability mode, and you make changes to the `ManagerService.exe.config` file after installation, you must make the changes on the primary and the failover vRealize Automation servers.

Procedure

1 Log in to the vRealize Automation server by using credentials with administrative access.

2 Edit the `ManagerService.exe.config` file in `%SystemDrive%\Program Files x86\VMware\vCAC\Server`, or in the vRealize Automation server install directory, if it is in a different location.

3 Edit the `RepositoryLogSeverity` and `RepositoryLogCategory` keys to configure what types of events get written to your log files.

Option	Description
RepositoryLogSeverity	Specify a severity level to ignore events below that severity. ■ *Error* logs only recoverable errors and higher ■ *Warning* logs noncritical warnings and higher ■ *Information* logs all informative messages and higher ■ *Verbose* logs a debugging trace and can impair performance For example, `<add key="RepositoryLogSeverity" value="Warning" />`.
RepositoryLogCategory	Specify a category to log all events for that category regardless of severity. For example, `<add key="RepositoryLogCategory" value="MissingMachines,UnregisteredMachines,AcceptMachineRequest,RejectMachineRequest" />` logs all events for missing or unregistered machines, and every accepted or rejected machine request.

4 Save and close the file.

5 Select **Start > Administrative Tools > Services** and restart the vCloud Automation Center service.

Results

You can see how your changes effect logging by viewing the Manager Service log file located in `%SystemDrive%\Program Files (x86)\VMware\vCAC\Server\Logs` on the machine where the Manager Service is installed, or in the vRealize Automation server install directory, if you installed it in a different location.

Monitoring vRealize Automation

Depending on your role, you can monitor workflows or services, view event or audit logs, or collect logs for all the hosts in a distributed deployment.

Monitoring Workflows and Viewing Logs

Depending on your role, you can monitor workflows and view activity logs.

Table 10-12. Monitoring and Log Display Options

Objective	Role	Menu Sequence and Description
Display information about actions that have occurred, such as the action type, date and time of the action, and so on.	IaaS administrator	Display default log information or control display content using column and filter options. Select **Infrastructure > Monitoring > Audit Log**. The audit log provides details about the status of managed virtual machines and activities performed on these machines during reconfiguration. The log includes information about machine provisioning, NSX, reclamation, and reconfigure actions.
View the status of scheduled and available Distributed Execution Manager and other workflows.	IaaS administrator	Display workflow status and optionally open a specific workflow to display its details. Select **Infrastructure > Monitoring > DEM Status**.
View and optionally export log data.	IaaS administrator	Display default log information or control display content using column and filter options. Select **Infrastructure > Monitoring > Log**.
View the status and history of executed Distributed Execution Manager and other workflows.	IaaS administrator	Display workflow history and optionally open a specific workflow to display its execution details. Select **Infrastructure > Monitoring > Workflow History**.
Display a list of events, including event type, time, user ID, and so on, and optionally display an event details page.	System administrator	View a list of events and their associated attributes, such as run time, event description, tenant name, target type and ID, and other characteristics. Select **Administration > Events > Event Logs**.
Monitor the status of your requests and view request details.	Tenant administrator or business group manager	Display the status of requests that you are responsible for or own. Click **Requests**.
View information about recent events.	IaaS administrator or Tenant administrator	Display recent events for the currently logged in user. Select **Infrastructure > Recent Events**

Monitoring Event Logs and Services

You can monitor vRealize Automation event logs and services to determine their current and historic states.

The default retention period for the event logs is 90 days. You can change the period from the `/etc/vcac/vcac.properties` file.

For information about clearing logs, see Customize Data Rollover Settings.

vRealize Automation Services

A system administrator can view the status of vRealize Automation services from the Event Log on the system administrator console.

Subsets of services are required to run individual product components. For example, identity services and UI core services must be running before you can configure a tenant.

The following tables tell you which services are associated with areas of vRealize Automation functionality.

Table 10-13. Identity Service Group

Service	Description
management-service	Identity Service Group
sts-service	Single Sign-on Appliance
authorization	Authorization Service
authentication	Authentication
eventlog-service	Event log service
licensing-service	Licensing service

Table 10-14. UI Core services

Service	Description
shel-ui-app	Shell Service
branding-service	Branding Service
plugin-service	Extensibility (Plug-in) Service
portal-service	Portal Service

All the following services are required to run the IaaS component.

Table 10-15. Service Catalog Group (Governance Services)

Service	Description
notification-service	Notification service
workitem-service	Work Item service
approval-service	Approval Service
catalog-service	Service Catalog

Table 10-16. IaaS Services Group

Service	Description
iaas-proxy-provider	IaaS Proxy
iaas-server	IaaS Windows machine

Table 10-17. XaaS

Service	Description
vco	vRealize Orchestrator
advanced-designer-service	XaaS blueprints and resource actions

Using vRealize Automation Audit Logging

vRealize Automation offers audit logging to support collection and retention of important system events.

Currently, vRealize Automation supports audit logging as an extension of event logging. This functionality provides basic auditing information, and retention settings are configurable only using the appropriate vRealize Automation REST API event broker service calls. Audit logging is currently available to tenant administrators and system administrators who can log on to tenants. It provides search and filter capabilities for events.

By default, vRealize Automation supports audit logging for workflow subscription, endpoint, and fabric group create, update, and delete events. vRealize Automation also supports audit logging customization for a variety of IaaS events as well.

vRealize Automation audit logging is deactivated by default. You can switch it on or off by toggling the **Enabled** check box in the Audit Log Integration section on the **vRA > Logs** page of the virtual appliance management interface.

Audit log information appears on the standard Event Logs page. As a tenant admin, select **Administration > Event Logs** to view this page. Audit events are identified in the event log table with the designation Audit in the Event Type field. Each entry shows an Event Description for each event as well as the Tenant, Time, User, and related Service Name.

Enabling audit logging for any other IaaS events requires a custom configuration file and running the appropriate commands on your IaaS host machine. Contact VMware Professional Services for assistance.

You can configure vRealize Automation to export events to an external syslog server, specifically VMware Log Insight.

Configure vRealize Automation for Log Insight Audit Logging

You can configure vRealize Automation to export audit events to VMware Log Insight to facilitate viewing audit events.

Audit logging is deactivated by default and you must enable it to generate and view audit logging events.

If used, SSL is configured on the vRealize Automation appliance where the Log Insight agent resides, and it concerns the connection to the Log Insight Syslog server. To use SSL, you must configure the appropriate certificates and connectivity between vRealize Automation and the Log Insight server installed on your deployment.

Prerequisites

vRealize Automation uses the Log Insight Agent that is installed by default on a vRealize Automation deployment to read log entries for viewing in Log Insight.

Procedure

1. Log in to the Virtual Appliance Management Interface as a system administrator.
2. Select **vRA > Logs**.
3. Verify that the **Enabled** check box for audit logging is selected under the Audit Log Integration heading.
4. Enter the **Host** machine name for the Log Insight server under the Log Insight Agent Configuration heading.

   a. Enter the **Host** machine name for the Log Insight agent.

   b. Enter the **Port** to be used for communication with the Log Insight agent.

   c. Select the appropriate communication protocol.

   d. Use the **SSL Enabled** check box to indicate whether SSL will be used for communication between the Log Insight agent and server.

   If you choose not to use SSL, you can ignore the remainder of the settings on the page. If SSL is used, you must configure these settings.
5. Make the appropriate selections in the SSL Trusted Root Certificates section if you are using SSL.

   By default, the vRealize Automation appliance uses a self-signed certificate. If you want to use a Trusted Root certificate, you must import it.

   a. Select the appropriate check box to indicate whether you want to use a new certificate or an existing certificate.

   See the notes on the Virtual Appliance Management Interface Configure vRealize Automation Logging page for more information.
6. Click **Save Settings**.
7. Make the appropriate selections in the SSL Server Certificates section.
8. Use the Agent Behavior Configuration section to configure how the agent works with log files.

Results

vRealize Automation audit log events are visible from the Log Insight interface.

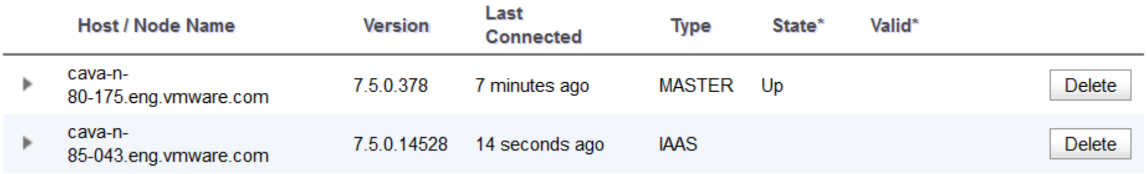
Viewing Host Information for Clusters in Distributed Deployments

You can collect logs for all nodes that are clustered in a distributed deployment from the vRealize Automation appliance management console.

You can also view information for each host in your deployment. The **Cluster** tab on the vRealize Automation management console includes a Distributed Deployment Information table that displays the following information:

- A list of all nodes in your deployment
- The host name for the node. The host name is given as a fully qualified domain name.
- The time since the host last replied to the management console. Nodes for IaaS components report availability every three minutes and nodes for virtual appliances report every nine minutes.
- The vRealize Automation component type. Identifies whether the node is a virtual appliance or an IaaS server.

Figure 10-1. Distributed Deployment Information table



	Host / Node Name	Version	Last Connected	Type	State*	Valid*	
▸	cava-n-80-175.eng.vmware.com	7.5.0.378	7 minutes ago	MASTER	Up		Delete
▸	cava-n-85-043.eng.vmware.com	7.5.0.14528	14 seconds ago	IAAS			Delete

You can use this table to monitor activity in your deployment. For example, if the Last Connected column indicates a host has not connected recently, that can be an indication of a problem with the host server.

Log Collection

You can create a zip file that contains log files for all hosts in your deployment using the Create Support Bundle button on the **vRA > Logs** page. For more information, see Collect Logs for Clusters and Distributed Deployments.

Removing Nodes from the Table

When you remove a host from your deployment, remove the corresponding node from the Distributed Deployment Information table to optimize log collection times. Click the **Delete** button to remove a node from the table.

Collect Logs for Clusters and Distributed Deployments

To support troubleshooting and record keeping activities, you can create a zip file that includes all log files for servers in your deployment.

The Distributed Deployment Information table on the Cluster tab of the Virtual Appliance Management Interface lists the nodes for which log files are collected. You can also delete nodes from this table.

For related information about vRealize Automation appliance deployment configuration, see Deploy the vRealize Automation Appliance and Configure the vRealize Automation Appliance.

Procedure

1. Log in to the Virtual Appliance Management Interface as a system administrator.
2. Click **vRA > Logs**.
3. Click **Create Support Bundle**.

   Log files for each node are collected and copied to a zip file.

Remove a Node from the Distributed Deployment Information Table

Delete a node when you want to remove it from your deployment cluster or when you are replacing a Management Agent certificate.

The Distributed Deployment Information table on the Cluster tab of the Virtual Appliance Management Interface lists the nodes for the applicable cluster. You can click the **Delete** button for any node on the table to remove that node from the cluster, or you can use the following procedure.

Procedure

1. Log in to the vRealize Automation appliance by using the user name `root` and the password you specified when you deployed the appliance.
2. Click the **Cluster** tab.

   The Distributed Deployment Information table displays a list of nodes for the distributed deployment.
3. Locate the node ID for the node to be deleted by opening a command prompt and running the following command:

   ```
   /usr/sbin/vcac-config cluster-config-node --action list
   ```

4. Locate the node ID, for example `cafe.node.46686239.17144`, in the JSON output.
5. Open a command prompt and type a command of the following form, using the node ID that you obtained in the previous step.

   ```
   /usr/sbin/vcac-config cluster-config-node
   --action delete --id node-UID
   ```

   For example, enter the following command for the example node ID `cafe.node.46686239.17144`:

   ```
   /usr/sbin/vcac-config cluster-config-node --action delete --id cafe.node.46686239.17144
   ```

6. Click **Refresh**.

   The node no longer appears in the display.

Monitoring vRealize Automation Health

The vRealize Automation Health Service assesses the functional health of a vRealize Automation environment.

IaaS administrators configure the Health Service to run test suites that determine if the components are registered and the necessary resources are available. This table shows the test suites provided by the Health Service and some example tests in each suite.

Health Service Test Suites	Example Tests
System tests for vRealize Automation	■ SSO/Identity VA Connection Test ■ vRealize Automation License Check - Is License Expired? ■ vRealize Automation Virtual Appliance Root Password Check - Is Password Expiring Soon?
Tenant tests for vRealize Automation	■ Check vSphere Reservation Storage Paths ■ Check Reservation Policy to Reservation Assignments ■ Check the Portal Service Status
Tests for vRealize Orchestrator	■ Check number of active vRO nodes ■ Check the utilization of the java memory heap in the vRO nodes ■ Check the status of the vro-server service in the vRO nodes

After you run a test suite on a virtual machine, the Health Service reports the number of tests that passed or failed. For each failed test, the Health Service provides these links:

Link	Content
Cause	Explanation of why the test failed.
Remediation	Information that you can use to fix the problem.

You can configure the Health Service to run tests on a schedule or only on demand.

You can also use Python to create custom tests. See the *vRealize Automation Health Service Extensibility Guide*.

Tenant administrators with a Health Consumer role can view test results for their tenancy but cannot configure or run a test.

Configure System Tests for vRealize Automation

An **IaaS administrator** configures the Health Service to run system tests on a selected vRealize Automation virtual appliance. These tests determine if components, such as the vRealize Automation license, are registered and necessary resources, such as memory, are available on the virtual appliance. When you configure the system tests, the Health page displays the tests as a test card.

To configure the Health Service to run system tests for vRealize Automation, complete this procedure.

Prerequisites

Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1 Select **Administration > Health**.

2 Click **New Configuration**.

3 On the Configuration Details page, provide the requested information.

Option	Description
Name	Your title for this configuration. This title appears on the test card.
Description	A description of the test suite.
Product	Select vRealize Automation.
Schedule	Select how often the test suite runs.

4 Click **Next**.

5 On the Select Test Suites page, select **System Tests for vRealize Automation**.

6 Click **Next**.

7 On the Configure Parameters page, provide the requested information.

Table 10-18. vRealize Automation Virtual Appliance

Option	Description
Public Web Server Address	■ For a minimal deployment, the base URL for the vRealize Automation appliance host. For example, https://*va-host.domain*/. ■ For a high-availability deployment, the base URL for the vRealize Automation load balancer. For example, https://*load-balancer-host.domain*/.
SSH Console Address	Fully qualified domain name of the vRealize Automation appliance. For example,*va-host.domain*.
SSH Console User	root
SSH Console Password	The root password.

Table 10-19. vRealize Automation System Tenant

Option	Description
System Tenant Administrator	administrator
System Tenant Password	The administrator password.

Table 10-20. vRealize Automation Disk Space Monitoring

Option	Description
Warning Threshold Percent	Acceptable percent of virtual appliance disk space that is used before the warning test fails.
Critical Threshold Percent	Acceptable percent of virtual appliance disk space that is used before the critical test fails.

8 Click **Next**.

9 On the Summary page, review the information.

10 Click **Finish**.

Tests run according to the selected schedule.

What to do next

View the vRealize Automation Health Service Test Suite Results

Configure Tenant Tests For vRealize Automation

An **IaaS administrator** configures the Health Service to run tenant tests on a selected vRealize Automation virtual appliance. These tests determine if tenant-related components, such as software-service, are registered and necessary resources, such as vSphere virtual machines, are available on the virtual appliance. When you configure the tenant tests, the Health page displays the tests as a test card.

To configure the Health Service to run tenant tests for vRealize Automation, complete this procedure.

Prerequisites

Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1 Select **Administration > Health**.

2 Click **New Configuration**.

3 On the Configuration Details page, provide the requested information.

Option	Description
Name	Your title for this configuration. This title appears on the test card.
Description	A description of the tests.
Product	Select vRealize Automation.
Schedule	Select how often these tests run.

4 Click **Next**.

5 On the Select Test Suites page, select **Tenant Tests for vRealize Automation**.

6 Click **Next**.

7 On the Configure Parameters page, provide the requested information.

Table 10-21. vRealize Automation Virtual Appliance

Option	Description
vRealize Automation Web Address	■ For a minimal deployment, the base URL for the vRealize Automation appliance host. For example, https://*va-host.domain*/. ■ For a high-availability deployment, the base URL for the vRealize Automation load balancer. For example, https://*load-balancer-host.domain*/.
SSH Console Address	Fully qualified domain name of the SSH host. For example, *ssh-host.domain*.
SSH Console User	root
SSH Console Password	Password for root.
Max Service Response time (ms)	Maximum amount of time in milliseconds the system waits for a response.

Table 10-22. vRealize Automation Tenant

Option	Description
Tenant Under Test	qe
Fabric Administrator Username	Fabric administrator user name. **Note** This fabric administrator must also have a tenant administrator and an IaaS administrator role in order for all of the tests to run.
Fabric Administrator Password	Password for fabric administrator.

Table 10-23. vRealize Automation System Tenant

Option	Description
System Tenant Administrator	administrator
System Tenant Password	Password for administrator.

Table 10-24. vRealize Automation Disk Space Monitoring

Option	Description
Critical Threshold Percent	Acceptable percent of virtual appliance disk space that is used before the critical test fails.

8 Click **Next**.

9 On the Summary page, review the information.

10 Click **Finish**.

Tests run according to the selected schedule.

What to do next

View the vRealize Automation Health Service Test Suite Results

Configure Tests For vRealize Orchestrator

An **IaaS administrator** configures the health service to run tests for vRealize Orchestrator on the vRealize Orchestrator host. These tests confirm that components, such as the vro-server service, are registered and necessary resources, such as sufficient Java memory heap, are available are available on the host machine. When you configure the vRealize Orchestrator tests, the Health page displays the tests as a test card.

Prerequisites

Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1 Select **Administration > Health**.

2 Click **New Configuration**.

3 On the Configuration Details page, provide the requested information.

Option	Description
Name	Your title for this configuration. This title appears on the test card.
Description	A description of the tests.

Option	Description
Product	Select vRealize Orchestrator.
Schedule	Select how often the tests run.

4 Click **Next**.

5 On the Select Test Suites page, select **Tests for vRealize Orchestrator**.

6 Click **Next**.

7 On the Configure Parameters page, provide the requested information.

Table 10-25. vRealize Orchestrator Host/Load Balancer

Option	Description
Client Address	■ For a minimal deployment, the fully qualified domain name of the vRealize Orchestrator host. For example, *vro-host.domain*. ■ For a high-availability deployment, the base URL for the vRealize Orchestrator load balancer, https://*load-balancer-host.domain*/.
Client Username	administrator
Client Password	The administrator password.
SSH Console Username	root
SSH Console Password	The root password.
Heap Utilization Threshold	Acceptable percent of heap space that is used before the warning test fails.

Table 10-26. vRealize Orchestrator Instances Behind Load Balancer

Option	Description
SSH Console Address	IP address or URL of the vRealize Orchestrator instance behind the load balancer.
SSH Console Username	User name with access to this instance.
SSH Console Password	The user name password.

- Click **ADD** to add another vRealize Orchestrator instance to the list.
- Click **REMOVE** to remove a selected vRealize Orchestrator instance from the list of instances behind the load balancer.

8 Click **Next**.

9 On the Summary page, review the information.

10 Click **Finish**.

Tests run according to the selected schedule.

What to do next

View the vRealize Automation Health Service Test Suite Results

Custom Test Suite

You can use Python to create a custom test suite for vRealize Automation Health Service.

Creating a custom test suite allows you to extend the tests supplied for the health service by adding a test suite to determine the health of additional vRealize Automation components. For information about creating a custom test suite, see the *vRealize Automation Health Service Extensibility Guide*.

Add a Custom Test Suite

An **IaaS administrator** must add a custom test suite to vRealize Automation health service before you run the test suite.

To add a custom test suite for a vRealize Automation asset, complete this procedure.

Prerequisites

- Create a Python wheel for the custom test suite files. For information, see the *vRealize Automation Health Service Extensibility Guide*.
- Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1 Click **Administration > Health**.

2 In the upper right, click the gear icon and select **Extensibility**.

3 Click **New Asset**.

4 In the Add Asset dialog box, provide the requested information.

Option	Description
Asset Title	The name and version number of the test suite you are running, for example, Infoblox 1.0.
Asset Description	A description of the tests contained in the Python wheel.
Asset Version	Test suite version number.
Asset File	Click **Choose File** and select your custom test suite file.

5 Click **Add**.

A new row is added to the asset table with the status UPLOADED. When the status changes to INSTALLED, your test suite is ready to use. If the install process fails, you see a popup that provides a reason.

Note If the page does not update, click the refresh icon.

What to do next

Run a Custom Test Suite.

Run a Custom Test Suite

An **IaaS administrator** configures the health service to run a custom test suite in the vRealize Automation environment. When you configure the custom test suite, the Health page displays the test suite as a test card.

To configure the health service to run a custom test suite for vRealize Automation, complete this procedure.

Prerequisites

- Add a Custom Test Suite.
- Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1 Select **Administration > Health**.

2 Click **New Configuration**.

3 On the Configuration Details page, provide the requested information.

Option	Description
Name	Your title for this configuration. This title appears on the test card.
Description	A description of the test suite.
Product	Select the product you want to test from the **Product** drop-down menu.
Schedule	Select how often you want to run this test suite.

4 Click **Next**.

5 On the Select Test Suites page, select the custom test suite, and click **Next**.

6 On the Configure Parameters page, enter the requested information, and click **Next**.

7 On the Summary page, review the information, and click **Finish**.

The custom test suite runs according to the selected schedule.

What to do next

View the vRealize Automation Health Service Test Suite Results

View the vRealize Automation Health Service Test Suite Results

You can view the health service test results after you run the tests.

The Health page displays each configured test suite as a test card. When a test suite runs, the result appears in the middle of the test card.

The test cards that you see on the Health page are filtered according to your privilege.

- IaaS administrators can see all test cards.
- Tenant administrators with the Health Consumer role can see only the test card for their tenancy.

Prerequisites

- The configured test suite has run on schedule.
- Log in to the vRealize Automation console as an **IaaS administrator** or as a **tenant administrator**.

Procedure

1 Select **Administration > Health**.

2 If a test is not scheduled to run, click **Run** on the test card.

3 Click the center of a test card after the tests are finished.

   A page appears that displays the status of each test. To see why a test failed, click **Cause**. To open a topic that explains how to fix the problem, click the **Remediation** link if one is available.

Troubleshooting the Health Service

The Health Service troubleshooting topics provide solutions to problems you might experience when you use the Health Service.

Service Status Test Fails

You can fix a failed service test by changing the test schedule setting.

Problem

If a service status test fails and you click **Cause**, you see this message: `Cannot establish SSH connection ; Exception message:[Auth fail]`.

Cause

When the test suite is scheduled to run every 15 minutes, the system login locks the root user account.

Solution

- Change the test schedule to **None**, wait 15 minutes, and run the test suite again.

After Upgrade the Health Page in the Appliance Console Is Empty

After you upgrade vRealize Automation, the Health page in the appliance console is empty.

Problem

The health service does not start after upgrade.

Solution

- On each vRealize Automation virtual appliance, open a command prompt as **`root`** and run these commands.

  a To configure the health service to start automatically, run this command.

  ```
  chkconfig vrhb-service on
  ```

  b To start the health service on this virtual appliance, run this command.

  ```
  service vrhb-service start
  ```

Monitoring vRealize Automation Environment Resources Using SNMP

As a system administrator familiar with SNMP, you want to use the vRealize AutomationvRealize Automation REST API for vSNMP to facilitate how you monitor your vRealize Automation nodes. Using vSNMP, you can use SNMP to act as an encrypted early warning system when vRealize Automation is about to run out of CPU, RAM, disk space so that you avoid slowdowns.

You can manually monitor the SNMP OIDs, or you can actively monitor resources by setting SNMP traps.

For example, if vSNMP sends you an event, such as "High CPU usage detected," you might start gathering information about the processes consuming CPU and determine which one is using excessive resources. You might then correlate the CPU, memory, and other usage to troubleshoot additional problems.

Using the vRealize Automation vSNMP, you can expose the entire Linux tree for monitoring and retrieving data using the REST API, or by using the vSNMPD daemon that is running on your vRealize Automation instances.

vRealize Automation SNMP does not have a general use interface. You must use the REST API or the daemon commands.

For more information, see "Using SNMP to Monitor vRealize Automation" in the vRealize Automation Programming Guide. To locate the Programming Guide, see vRealize Automation API Documentation, and select the version link.

Monitoring and Managing Resources

Different vRealize Automation roles monitor resource usage and manage infrastructure in different ways.

Choosing a Resource Monitoring Scenario

Fabric administrators, tenant administrators, and business group managers have different concerns when it comes to resource monitoring. Because of this, vRealize Automation allows you to monitor different facets of resource usage.

For example, a fabric administrator is concerned with monitoring the resource consumption of reservations and compute resources, whereas a tenant administrator is concerned with the resource usage of the provisioning groups within a tenant. Depending on your role and the specific resource usage you want to monitor, vRealize Automation allows you different ways to track resource consumption.

Table 10-27. Choose a Resource Monitoring Scenario

Resource Monitoring Scenario	Privileges Required	Location
Monitor the amount of physical storage and memory on your compute resources that is currently being consumed and determine what amount remains free. You can also monitor the number of reserved and allocated machines provisioned on each compute resource.	**Fabric Administrator** (monitor resource usage on compute resources in your fabric group)	**Infrastructure > Compute Resources > Compute Resources**
Monitor machines that are currently provisioned and under vRealize Automation management.	**Fabric Administrator**	**Infrastructure > Machines > Managed Machines**
Monitor the amount of storage, memory, and machine quota of your reservation that is currently allocated and determine the capacity that remains available to the reservation.	**Fabric Administrator** (monitor resource usage for reservations on your compute resources and physical machines)	**Infrastructure > Reservations > Reservations**
Monitor the amount of storage, memory, and the machine quota that your business groups are currently consuming and determine the capacity that remains on reserve for them.	■ **Tenant Administrator** (monitor resource usage for all groups in your tenant) ■ **Business Group Manager** (monitor resource usage for groups that you manage)	**Administration > Users & Groups > Business Groups**

Resource Usage Terminology

vRealize Automation uses explicit terminology to distinguish between resources that are available, resources that have been set aside for specific usages, and resources that are actively being consumed by provisioned machines.

The Resource Usage Terminology table explains the terminology vRealize Automation uses to display resource usage.

Table 10-28. Resource Usage Terminology

Term	Description
Physical	Indicates the actual memory or storage capacity of a compute resource.
Reserved	Indicates the machine quota, memory, and storage capacity set aside for a reservation. For example, if a compute resource has a physical capacity of 600 GB and there are three reservations on it for 100 GB each, then the reserved storage of the compute resource is 300 GB and the storage reserved is 50 percent.
Managed	Indicates that the machine is provisioned and currently under vRealize Automation management.
Allocated	Indicates the machine quota, memory, or storage resources actively being consumed by provisioned machines. For example, consider a reservation with a machine quota of 10. If there are 15 provisioned machines on it, but only 6 of them are currently powered on, the machine quota is 60 percent allocated.
Used	The **Used** column value always equals the **Allocated** column value.
Free	Indicates the unused physical capacity on a storage path.

Connecting to a Cloud Machine

The first time you connect to a cloud machine you must log in as Administrator.

You can then add the credentials under which you log in to the vRealize Automation console as a user on the machine, and log in under your vRealize Automation credentials from that point on.

Important If you are using Amazon Web Services, RDP, or SSH must be enabled on the Amazon machine instance and the machines must be in a security group in which the correct ports are open.

Collect User Credentials for an Amazon Machine

To log in to an Amazon machine as an administrator, you must discover the machine's administrator password.

The administrator password is available on the Machine Information Details page. If the Amazon machine image from which the machine was provisioned is not configured to generate the administrator password on every boot, you will need to find the password using an alternate technique. For information about otherwise obtaining the administrator password, search on *Connect to Your Amazon EC2 Instance* topics in Amazon documentation.

If needed, you can create the necessary vRealize Automation user credentials. The user credentials are then valid for subsequent logins to that machine.

Prerequisites

- The Amazon machine has already been provisioned.
- Log in to the vRealize Automation as a machine owner, **business group manager**, or **support user**.
- RDP or SSH is active on the Amazon machine image that will be used for provisioning
- The machines are in a security group in which the correct ports are open.

Procedure

1. Navigate to the **Items** page and filter on the groups you manage or a specific group.
2. Select the Amazon machine in the list of machines.

   You can click **View Details** on the **Actions** drop-down menu to display details such as machine type.
3. Select **Edit** in the **Actions** drop-down menu.
4. Click **Show Administrator Password** to obtain the administrator password of the machine.

   Alternatively, you can obtain the password using an external Amazon procedure.
5. Click **Connect Using RDP** from the **Actions** drop-down menu.
6. Click **User another account** when prompted for the login credentials.
7. Type `LOCAL\Administrator` when prompted for the user name.
8. Type the administrator password when prompted.
9. Click **OK**.

   You are now logged in to the machine as an administrator.
10. Add your vRealize Automation credentials as appropriate. For example, on a Windows server machine, open the server manager and select **Configuration > Local Users and Groups** and add your credentials, using a `DOMAIN\username` format, to the **Remote Desktop Users** group.

    Your vRealize Automation user name and password are now valid credentials for subsequent login to this machine.
11. Log out of the Amazon machine.
12. Click **Connect Using RDP** from the **Actions** drop-down menu.
13. When prompted to log in, type your vRealize Automation user name and password credentials to log in to the machine.

Results

Machine owners can now log in to the machine using their vRealize Automation credentials.

Collect User Credentials for a vCloud Machine

To log in to an vCloud Air or vCloud Director machine as an administrator, you must discover the machine's administrator password.

The administrator password is available on the Machine Information Details page. If the machine image from which the machine was provisioned is not configured to generate the administrator password on every boot, you can find the password using an alternate technique. For information about otherwise obtaining the administrator password, see vCloud Air or vCloud Director documentation.

If needed, you can create the necessary vRealize Automation user credentials. The user credentials are then valid for subsequent logins to that machine.

Prerequisites

- The vCloud Air or vCloud Director machine has already been provisioned.
- Log in to the vRealize Automation as a machine owner, **business group manager**, or **support user**.
- RDP or SSH is active on the vCloud Air or vCloud Director machine image that will be used for provisioning
- The machines are in a security group in which the correct ports are open.

Procedure

1. Navigate to the **Items** page and filter on the groups you manage or a specific group.
2. Select the vCloud Air or vCloud Director machine in the list of machines.

   You can click **View Details** on the **Actions** drop-down menu to display details such as machine type.
3. Select **Edit** in the **Actions** drop-down menu.
4. Click **Show Administrator Password** to obtain the administrator password of the machine.

   Alternatively, you can obtain the password using an external vCloud Air or vCloud Director procedure.
5. Click **Connect Using RDP** from the **Actions** drop-down menu.
6. Click **User another account** when prompted for the login credentials.
7. Type `LOCAL\Administrator` when prompted for the user name.
8. Type the administrator password when prompted.
9. Click **OK**.

   You are now logged in to the machine as an administrator.

10 Add your vRealize Automation credentials as appropriate. For example, on a Windows server machine, open the server manager and select **Configuration > Local Users and Groups** and add your credentials, using a `DOMAIN\username` format, to the **Remote Desktop Users** group.

   Your vRealize Automation user name and password are now valid credentials for subsequent login to this machine.

11 Log out of the vCloud Air or vCloud Director machine.

12 Click **Connect Using RDP** from the **Actions** drop-down menu.

13 When prompted to log in, type your vRealize Automation user name and password credentials to log in to the machine.

Results

Machine owners can now log in to the machine using their vRealize Automation credentials.

Reducing Reservation Usage by Attrition

Fabric administrators can reduce the number of machines on a particular reservation over the long term while keeping the reservation and the existing machines provisioned on it active.

You can reduce the reserved machine quota, memory, and storage of a virtual reservation below the amount currently allocated. This allows management of existing machines to continue without change while preventing provisioning of new machines until allocation falls below the new reserved amount.

Note Because virtual machines that are powered off are not included in allocated memory and machine quota totals, reducing the memory or machine allocation of a reservation might prevent machines that are currently powered off from being powered back on.

For example, consider a business group with a reservation that contains 20 provisioned machines that are set to expire over the next 90 days. If you want to reduce this reservation by attrition to no more than 15 machines, you can edit the reservation to reduce the quota from 20 machines to 15. No further machines can be provisioned on the reservation until the number of machines on the reservation is naturally reduced by the upcoming expirations.

Decommissioning a Storage Path

If you are decommissioning a storage path and moving machines to a new one, a fabric administrator must deactivate the storage path in vRealize Automation.

The following is a high-level overview of the sequence of steps required to decommission a storage path:

1 A fabric administrator deactivates the storage path on all reservations that use it. See Deactivate a Storage Path.

2 Move the machines to a new storage path outside of vRealize Automation.

3 Wait for vRealize Automation to automatically run inventory data collection or initiate inventory data collection manually. See Configure Compute Resource Data Collection.

Deactivate a Storage Path

Fabric administrators can deactivate storage paths on reservations when storage paths are decommissioned.

Note For each reservation where you deactivate a storage path, verify that there is sufficient space remaining on other enabled storage paths.

Prerequisites

Log in to vRealize Automation as a **fabric administrator**.

Procedure

1 Select **Infrastructure > Reservations > Reservations**.

2 Point to the reservation on which the storage path you are decommissioning is used and click **Edit**.

3 Click the **Resouces** tab.

4 Locate the storage path you are decommissioning.

5 Click the **Edit** icon (✏).

6 Select the check box in the Disabled column to deactivate this storage path.

7 Click the **Save** icon (✅).

8 Click **OK**.

9 Repeat this procedure for all reservations that use the storage path you are decommissioning.

Data Collection

vRealize Automation collects data from infrastructure source endpoints and their compute resources.

Data collection occurs at regular intervals. Each type of data collection has a default interval that you can override or modify. Each type of data collection also has a default timeout interval that you can override or modify.

IaaS administrators can manually initiate data collection for infrastructure source endpoints and fabric administrators can manually initiate data collection for compute resources.

Table 10-29. Data Collection Types

Data Collection Type	Description
Infrastructure Source Endpoint Data Collection	Updates information about virtualization hosts, templates, and ISO images for virtualization environments. Updates virtual datacenters and templates for vCloud Director. Updates Amazon regions and machines provisioned on Amazon regions. Endpoint data collection runs every 4 hours.
Inventory Data Collection	Updates the record of the virtual machines whose resource use is tied to a specific compute resource, including detailed information about the networks, storage, and virtual machines. This record also includes information about unmanaged virtual machines, which are machines provisioned outside of vRealize Automation. Inventory data collection runs every 24 hours. The default timeout interval for inventory data collection is 2 hours.
State Data Collection	Updates the record of the power state of each machine discovered through inventory data collection. State data collection also records missing machines that vRealize Automation manages but cannot be detected on the virtualization compute resource or cloud endpoint. State data collection runs every 15 minutes. The default timeout interval for state data collection is 1 hour.
Performance Data Collection (vSphere compute resources only)	Updates the record of the average CPU, storage, memory, and network usage for each virtual machine discovered through inventory data collection. Performance data collection runs every 24 hours. The default timeout interval for performance data collection is 2 hours.
Network and security inventory data collection (vSphere compute resources only)	Updates the record of network and security data related to vCloud Networking and Security and NSX, particularly information about security groups and load balancing, for each machine following inventory data collection.
WMI data collection (Windows compute resources only)	Updates the record of the management data for each Windows machine. A WMI agent must be installed, typically on the Manager Service host, and enabled to collect data from Windows machines.

Start Endpoint Data Collection Manually

Endpoint data collection runs automatically every 4 hours, but IaaS administrators can manually start endpoint data collection at any time for endpoints that do not require proxy agents.

The **Data Collection** page provides information on the status and age of data collections and allows you to manually start a new endpoint data collection.

Prerequisites

Log in to vRealize Automation as an **IaaS administrator**.

Procedure

1. Select **Infrastructure > Endpoints > Endpoints**.
2. Click in the row of the endpoint that you want to data collect.
3. Select an available data collection action.

Configure Compute Resource Data Collection

You can activate or deactivate data collection, configure the frequency of data collection, or manually request data collection.

The **Data Collection** page provides information on the status and age of data collections. It also allows you to configure data collection for your compute resources.

Prerequisites

Log in to vRealize Automation as a **fabric administrator**.

Procedure

1. Select **Infrastructure > Compute Resources > Compute Resources**.
2. Point to the compute resource for which to configure data collection and click **Data Collection**.
3. Configure **Compute Resource** data collection specifications.
   - Select **On** to activate data collection.
   - Select **Off** to deactivate data collection.
4. Configure **Inventory** data collection.
   - Select **On** to activate data collection.
   - Select **Off** to deactivate data collection.
   - Enter a number in the **Frequency** text box to configure the time interval (in hours) between inventory data collections.
   - Click **Request Now** to manually start data collection.
5. Configure **State** data collection.
   - Select **On** to activate data collection.
   - Select **Off** to deactivate data collection.
   - Enter a number in the **Frequency** text box to configure the time interval (in minutes) between state data collections.
   - Click **Request Now** to manually start data collection.

6 Configure **Performance** data collection.

   This is available only for vSphere integrations.

   - Select **On** to activate data collection.
   - Select **Off** to deactivate data collection.
   - Enter a number in the **Frequency** text box to configure the time interval (in hours) between performance data collections.
   - Click **Request Now** to manually start data collection.

7 Configure **Snapshot Inventory** data collection.

   This is option is available for compute resources managed by vRealize Business for Cloud.

   - Select **On** to activate data collection.
   - Select **Off** to deactivate data collection.
   - Enter a number in the **Frequency** text box to configure the time interval (in hours) between snapshot data collections.
   - Click **Request Now** to manually start data collection.

8 Click **OK**.

Update Cost Data for All Compute Resources

Fabric administrators can manually update cost information for all compute resources managed by vRealize Business for Cloud.

Prerequisites

Log in to vRealize Automation as a **fabric administrator**.

Procedure

1 Select **Infrastructure > Compute Resources > Compute Resources**.

2 Click **Update Cost**.

3 Click **Request Now**.

Results

When the cost update is complete, the status changes to successful.

Understanding vSwap Allocation Checking for vCenter Server Endpoints

You can use vSwap to determine swap space availability for the maximum size swap file on a target machine. The vSwap check occurs when you create or reconfigure a virtual machine from vRealize Automation. vSwap allocation checking is only available for vCenter Server endpoints.

vRealize Automation storage allocation checks if there is sufficient space available on the datastore to accommodate virtual machine disks during a create or reconfigure request. However, when the machine is powered on, if enough space is not available to create swap files on the vCenter Server endpoint, the machine fails to power on. When the power on operation fails, any customizations that depend on the machine also fail. The machine may also be disposed of. Depending on the size of the request, feedback that the machine is not powering on or not provisioning is not immediately obvious.

You can use the vSwap allocation check to help overcome these limitations by checking swap space availability for the maximum size swap file as part of the vRealize Automation create and reconfigure process for vCenter Server endpoints. To enable the vSwap allocation check, set the custom property `VirtualMachine.Storage.ReserveMemory` to True in the machine component or overall blueprint.

Consider the following behaviors for vSwap allocation checks:

- The swap file is located on the datastore that contains the virtual machine. Alternate vCenter Server configurations for locating swap files on a dedicated or different datastore are not supported.
- Swap size is considered when creating or reconfiguring a virtual machine . The maximum swap size is the size of the virtual machine's memory.
- Reserved values for vRealize Automation storage reservations in a host must not exceed the physical capacity of the compute resource.
- When creating a reservation, the sum of the reserved values must not exceed the available storage space.
- Resource pool or host level or virtual machine level memory reservations on vSphere are not collected from the vSphere endpoint and not considered during the calculations on vRealize Automation.
- vSwap does not validate the swap space that is available during power on operations for existing machines.
- You must re-run data collection to capture any changes made to the vSphere endpoint relative to vSwap.

Removing Datacenter Locations

To remove a datacenter location from a user menu, a system administrator must remove the location information from the locations file and a fabric administrator must remove location information from the compute resource.

For example, if you add London to the locations file, associate ten compute resources with that location, and then remove London from the file, the compute resources are still associated with the location London and London is still included in the location drop-down list on the Confirm Machine Request page. To remove the location from the drop-down list, a fabric administrator must edit the compute resource and reset the Location to blank for all compute resources that are associated with the location.

The following is a high-level overview of the sequence of steps required to remove a datacenter location:

1 A system administrator removes the datacenter location information from the locations file.

2 A fabric administrator removes all the compute resource associations to the location by editing the locations of each associated compute resource.

Monitoring Containers

You can monitor the status of a container that you create in Containers for vRealize Automation.

After you create your containers based on a template, you can monitor their state. By clicking **Details** on a container, you can monitor the network bandwidth, CPU and memory usage, logs, and properties of that container.

Bulk Import, Update, or Migrate Virtual Machines

You can use the Bulk Imports feature to import, update, or migrate virtual machines to vRealize Automation. Bulk Imports streamlines the management of multiple machines in multiple environments.

Bulk Imports creates a CSV file that contains defining virtual machine data such as reservation, storage path, blueprint, owner, and any custom properties. You use the CSV file to import virtual machines to your vRealize Automation environment. Bulk Imports supports the following administrative tasks:

- Import one or more unmanaged virtual machines so that they can be managed in a vRealize Automation environment.
- Make a global change to a virtual machine property, such as a storage path.
- Migrate a virtual machine from one vRealize Automation environment to another.

Note Only vCloud Director and vSphere are supported for bulk import. Setting the filter to another endpoint type does not generate data in the CSV file.

You can run the Bulk Imports feature commands using either the vRealize Automation console or the CloudUtil command-line interface. For more information about using the CloudUtil command-line interface, see the *Life Cycle Extensibility* documentation.

Note Bulk machine importing does not bypass normal provisioning steps. Any existing external workflows that are triggered by the Event Broker during provisioning are run for imported machines. You can temporarily deactivate workflows for imported machines by performing one of the following:

- Deactivate all Event Broker subscriptions. If you are deactivating subscriptions, you must schedule a service outage for your vRealize Automation cluster because extensiblity will not be applied to any normal machine provisioned during this time.
- Add a condition to event subscriptions to not trigger when a machine is imported. To add this condition, navigate to Event Subscriptions, select the subscription to deactivate, and add a custom property `VirtualMachine.Imported.ConvergedBlueprint` does not equal `<Id of the import blueprint>`. This condition does not effect normally provisioned machines and instead is only applied to imported machines.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator** and as a **business group manager**.
- If you are importing virtual machines that use static IP addresses, prepare a properly configured address pool.

Import a Virtual Machine to a vRealize Automation Environment

You can import an unmanaged virtual machine to a vRealize Automation environment.

An unmanaged virtual machine exists in a hypervisor but is not managed in a vRealize Automation environment and cannot be viewed in the console. After you import an unmanaged virtual machine, the virtual machine is managed using the vRealize Automation management interface. Depending on your privileges, you can see the virtual machine on the **Managed Machines** tab or the **Deployments** tab.

The bulk import option does not support deployments that are provisioned from a blueprint that contains an NSX network and security component or a software component.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator** and as a **business group manager**.
- If you are importing virtual machines that use static IP addresses, prepare a properly configured address pool. For more information, see Using Network Profiles to Control IP Address Ranges .
- If you use bulk import to import a virtual machine with a static IP address that is allocated to another virtual machine, the import fails.

Procedure

1. Temporarily deactivate all Event Broker Subscriptions.

   Note When disabling subscriptions you must schedule a service outage for your vRealize Automation cluster. During this process, extensibility is not applied to any normally provisioned machine. Failure to deactivate subscriptions can result in data loss and permanent deletion of machines from the backing infrastructure.

2. Generate a virtual machine CSV data file.

   a. Select **Infrastructure > Administration > Bulk Imports**.

   b. Click **Generate CSV File**.

   c. Select **Unmanaged** from the **Machines** drop-down menu.

   d. Select the **Business group** default value from the drop-down menu.

   e. Enter the **Owner** default value.

   f. Select the **Blueprint** default value from the drop-down menu.

      The blueprint must be published and added to an entitlement for the import to be successful.

   g. Select the **Component machine** default value from the drop-down menu.

      If you select a value for **Business group** and **Blueprint**, you might see the following results in the CSV data file:

      - `Host Reservation (Name or ID) = INVALID_RESERVATION`
      - `Host To Storage (Name or ID) = INVALID_HOST_RESERVATION_TO_STORAGE`

      These messages appear if you do not have a reservation in the selected business group for the host virtual machine that also hosts the unmanaged virtual machine. If you have a reservation in that business group for the unmanaged virtual machine host, the Host Reservation and Host to Storage values fill in properly.

   h. Select one of the available resource types from the **Resource** drop-down menu.

Menu Item	Description
Endpoint	Information required to access a virtualization host.
Compute Resource	Information required to access a group of virtual machines performing a similar function.

   i. Select the name of the virtual machine resource from the **Name** drop-down menu.

   j. Click **OK**.

3 Edit your virtual machine CSV data file.

a Open the CSV file, and edit the data categories to match existing categories in the target vRealize Automation environment.

To import virtual machines contained in a CSV data file, each virtual machine must be associated with the following items:

- Reservation
- Storage location
- Blueprint
- Virtual machine component
- Owner that exists in the target deployment

All the values for each virtual machine must be present in the target vRealize Automation environment for the import to succeed. You can change the values for reservation, storage location, blueprint, and owner, or add a static IP address value to individual virtual machines by editing the CSV file.

Heading	Comment
# Import—Yes or No	Change to No to prevent a particular virtual machine from being imported.
Virtual Machine Name	Do not change.
Virtual Machine ID	Do not change.
Host Reservation (Name or ID)	Enter the name or ID of a reservation in the target vRealize Automation environment.
Host To Storage (Name or ID)	Enter the name or ID of a storage location in the target vRealize Automation environment.
Deployment Name	Enter a new name for the deployment, for example, the virtual machine name, you are creating in the target vRealize Automation environment. **Note** Each virtual machine must be imported to its own deployment. You cannot import a single virtual machine to an existing deployment. You cannot import multiple virtual machines to a single deployment.
Blueprint ID	Enter the ID of the blueprint in the target vRealize Automation environment that you use to import the virtual machine. **Note** Enter only the blueprint ID, not the blueprint name. You must select a blueprint that contains only a single virtual machine component. The blueprint must be published and added to an entitlement. For imported virtual machines, do not associate a blueprint that includes component profiles. Existing settings in imported virtual machines, such as memory or storage size, might be outside of profile limits. When that happens, validation for any future blueprint-based reconfiguration of the virtual machines fails.

Heading	Comment
Component Machine ID	Enter the name of a virtual machine component that is contained in the blueprint you selected. You cannot import a virtual machine into a blueprint that has more than one component.
Owner Name	Enter a user in the target vRealize Automation environment who is entitled to the blueprint.

If you import a virtual machine with one or more custom properties, you identify each custom property using three comma separated values appended to the line with the values for that machine. Use this format for each custom property.

```
,Custom.Property.Name, Value, FLAGS
```

FLAGS are three characters that describe how the property is treated by vRealize Automation. In their order of use, the flags are:

1 H or N = Hidden or Not Hidden

2 E or O = Encrypted or Not Encrypted

3 R or P = Runtime or Not Runtime

For example, you can append a custom property to configure a static IP address for a machine. Using the following format, this custom property allocates an available static IP address from a network profile.

```
,VirtualMachine.Network#.Address, w.x.y.z, HOP
```

You change the variables with the appropriate information for your virtual machine.

- Change *#* to the number of the network interface being configured with this static IP address. For example, `VirtualMachine.Network0.Address`.
- Change *w.x.y.z* to be the static IP address for the virtual machine. For example, `11.27.42.57`.

The HOP flag string—Hidden, Not encrypted, Not Runtime—sets the visibility of the property. Because this particular property is used only by bulk import, it is removed from the virtual machine after a successful import.

In order for this custom property to work, the IP address must be available in a properly configured address pool. If the address cannot be found or is already in use, the import succeeds without the static IP address definition, and an error is logged.

b Save the CSV file.

4 Use the vRealize Automation management interface to import your virtual machine to a vRealize Automation environment.

a Select **Infrastructure > Administration > Bulk Imports**.

b Click **New**.

c Enter a unique name for this task in the **Name** text box, for example, unmanaged import 10.

d Enter the CSV filename in the **CSV file** text box by browsing to the CSV filename.

e Select import options.

Option	Description
Start time	Schedule a future start date. The chosen start time is the local server time and not the local time of the user workstation.
Now	Begin the import process immediately.
Delay (seconds)	If you are importing many virtual machines, select the number of seconds to delay each virtual machine registration. Selecting this menu item slows the import process. Leave blank to select no delay.
Batch size	If you are importing many virtual machines, select the total number of virtual machines to register at a given time. Selecting this menu item slows the import process. Leave blank to select no limit.
Ignore managed machines	Leave unselected.
Skip user validation	Selecting this menu item sets the virtual machine owner to the value listed in the Owner column of the CSV data file without verifying that the user exists. Selecting this menu item can decrease the import time.
Test import	Test the import process without importing the virtual machines so you can test your CSV file for errors.

f Click **OK**.

The progress of the operation appears on the Bulk Imports page.

Update a Virtual Machine in a vRealize Automation Environment

You can make a change to a virtual machine property, such as a storage path, to update one or more managed virtual machines in a vRealize Automation environment.

A managed virtual machine is a machine that is managed in a vRealize Automation environment and can be viewed in the console.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator** and as a **business group manager**.

Procedure

1 Generate a virtual machine CSV data file.

a Select **Infrastructure > Administration > Bulk Imports**.

b Click **Generate CSV File**.

c Select **Managed** from the **Machines** drop-down menu.

d Select one of the available resource types from the **Resource** drop-down menu.

Option	Description
Endpoint	Information required to access a virtualization host.
Compute Resource	Information required to access a group of virtual machines performing a similar function.

e Select the name of the virtual machine resource from the **Name** drop-down menu.

f (Optional) Select **Include custom properties** if you want to migrate the virtual machine custom properties.

g Click **OK**.

2 Edit your virtual machine CSV data file.

   a Open the CSV file with a text editor and edit the data categories that you want to change globally.

   Note The deployment name can not match the current deployment name otherwise validation fails.

   To update virtual machines contained in a CSV data file, each machine must be associated with the following items:

   - Reservation
   - Storage location
   - Blueprint
   - Machine component
   - Owner that exists in the target deployment

   All of the values for each machine must be present in the target vRealize Automation environment for the update to succeed. You can change the values for reservation, storage location, blueprint, and owner, or add a static IP address value to individual machines by editing the CSV file.

   b If you are changing a virtual machine static IP address, append a command in the following form to the CSV file.

   ```
   ,VirtualMachine.Network#.Address, w.x.y.z, HOP
   ```

   Configure the command with the appropriate information for your virtual machine.

   - Change the *#* to the number of the network interface being configured with this static IP address. For example, `VirtualMachineNetwork0.Address`.
   - Change *w.x.y.z* to be the static IP address for the virtual machine. For example, `11.27.42.57`.
   - The *HOP* string, Hidden, Not encrypted, Not runtime, sets the visibility of the property. This default property is removed from the virtual machine after a successful import.

   For a successful update, the IP address must be available in a properly configured address pool. If the address cannot be found or is already in use, the update succeeds without the static IP address definition, and an error is logged.

   c Save the CSV file and close your text editor.

3 Use the vRealize Automation management interface to update one or more virtual machines in a vRealize Automation environment.

   a Select **Infrastructure > Administration > Bulk Imports**.

   b Click **New**.

c Enter a unique name for this task in the **Name** text box, for example, managed global update 10.

d Enter the CSV file name in the **CSV file** text box by browsing to the CSV file name.

e Select import options.

Option	Description
Start time	Schedule a future start date. The specified start time is the local server time and not the local time of the user workstation.
Now	Begin the import process immediately.
Delay (seconds)	If you are updating a large number of virtual machines, select the number of seconds to delay each virtual machine update. Selecting this option slows the update process. Leave blank to specify no delay.
Batch size	If you are updating a large number of virtual machines, select the total number of machines to update at a given time. Selecting this option slows the update process. Leave blank to specify no limit.
Ignore managed machines	Leave unselected.
Skip user validation	Selecting this option sets the machine owner to the value listed in the Owner column of the CSV data file without verifying that the user exists. Selecting this option can decrease the update time.
Test import	Leave unselected.

f Click **OK**.

The progress of the operation appears on the Bulk Imports page.

Migrate a Virtual Machine to a Different vRealize Automation Environment

You can migrate one or more managed virtual machines in a VMware vRealize ™ Automation environment to a different vRealize Automation environment.

A managed virtual machine is a virtual machine that is managed in a vRealize Automation environment and can be viewed in the console.

Prerequisites

- Log in to vRealize Automation as a **fabric administrator** and as a **business group manager**.
- If you are importing virtual machines that use static IP addresses, prepare a properly configured address pool.

Procedure

1 Generate a virtual machine CSV data file.

a Select **Infrastructure > Administration > Bulk Imports**.

b Click **Generate CSV File**.

c Select **Managed** from the **Machines** drop-down menu.

d Select one of the available resource types from the **Resource** drop-down menu.

Option	Description
Endpoint	Information required to access a virtualization host.
Compute Resource	Information required to access a group of virtual machines performing a similar function.

e Select the name of the virtual machine resource from the **Name** drop-down menu.

f (Optional) Select **Include custom properties**.

You include custom properties when you import a virtual machine into a new deployment with the same properties.

g Click **OK**.

2 Edit your virtual machine CSV data file.

Whether you must edit the CSV data file depends on the similarity of the source and target environments. If the configuration values in the source environment do not match the values in the target environment, you must edit the CSV data file so that the values match before you begin migration.

a Open the CSV file, and edit the data categories to match existing categories in the target vRealize Automation environment.

To migrate virtual machines contained in a CSV data file, each virtual machine must be associated with a reservation, storage location, blueprint, machine component, and owner that exists in the target vRealize Automation environment. All the values for each virtual machine must be present in the target vRealize Automation environment for migration to succeed. You can change the values for reservation, storage location, blueprint, and owner, or add a static IP address value to individual virtual machines by editing the CSV file.

Heading	Comment	Example
# Import--Yes or No	Change to No to prevent a particular virtual machine from being imported.	Yes
Virtual Machine Name	Do not change.	MyMachine
Virtual Machine ID	Do not change.	a6e05812-0b06-4d4e-a84a-fed242340426a
Host Reservation (Name or ID)	Enter the name or ID of a reservation in the target vRealize Automation environment.	DevReservation
Host To Storage (Name or ID)	Enter the name or ID of a storage location in the target vRealize Automation environment.	ce-san-1:custom-nfs-2
Deployment Name	Enter a new name for the deployment you are creating in the target vRealize Automation environment. Each virtual machine must be migrated to its own deployment. You cannot import a single virtual machine to an existing deployment. You cannot import multiple virtual machines to a single environment.	ImportedDeployment0001
Converged Blueprint ID	Enter the ID of the blueprint in the target vRealize Automation environment that you use to import the virtual machine. Make sure that you enter only the blueprint ID. Do not enter the blueprint name. You must select a blueprint that contains only a single virtual machine component. The blueprint must be published and added to an entitlement.	ImportBlueprint

Heading	Comment	Example
Component Blueprint ID	Enter the name of a virtual machine component that is contained in the blueprint you selected. You cannot import a virtual machine into a blueprint that has more than one component.	ImportedMachine
Owner Name	Enter a user in the target vRealize Automation environment.	user@tenant

Example of a complete, properly formatted CSV line: Yes, MyMachine, a6e05812-0b06-4d4e-a84a-fed242340426, DevReservation, ce-san-1:custom-nfs-2, Imported Deployment 0001, ImportBlueprint, ImportedMachine, user@tenant

b If you are migrating a virtual machine with a static IP address, append a command in the following form to the CSV file.

```
,VirtualMachine.Network#.Address, w.x.y.z, HOP
```

Configure the command with the appropriate information for your virtual machine.

- Change the # to the number of the network interface being configured with this static IP address. For example, `VirtualMachineNetwork0.Address`.
- Change *w.x.y.z* to be the static IP address for the virtual machine. For example, `11.27.42.57`.
- The *HOP* string, Hidden, Not encrypted, Not runtime, sets the visibility of the property. This default property is removed from the virtual machine after a successful import.

For a successful migration, the IP address must be available in a properly configured address pool. If the address cannot be found or is already in use, the migration succeeds without the static IP address definition, and an error is logged.

c Save the CSV file.

3 Use the vRealize Automation management interface to migrate your virtual machine to a vRealize Automation environment.

a Select **Infrastructure > Administration > Bulk Imports**.

b Click **New**.

c Enter a unique name for this task in the **Name** text box, for example, managed migration 10.

d Enter the CSV filename in the **CSV file** text box by browsing to the CSV filename.

e Select import options.

Option	Description
Start time	Schedule a future start date. The chosen start time is the local server time and not the local time of the user workstation.
Now	Begin the migration process immediately.
Delay (seconds)	If you are migrating many virtual machines, select the number of seconds to delay each virtual machine registration. Selecting this option slows the migration process. Leave blank to select no delay.
Batch size	If you are migrating many virtual machines, select the total number of virtual machines to register at a given time. Selecting this option slows the migration process. Leave blank to select no limit.
Ignore managed machines	Leave unselected.
Skip user validation	Selecting this option sets the virtual machine' owner to the value listed in the Owner column of the CSV data file without verifying that the user exists. Selecting this option can decrease the migration time.
Test import	Test the migration process without migrating the virtual machines so you can test your CSV file for errors.

f Click **OK**.

The progress of the operation appears on the Bulk Imports page.