Scenarios for Setting Up TLS Certificates for Horizon

VMware Horizon 2106

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

Scenarios for Setting Up TLS Certificates for Horizon

Scenarios for Setting Up TLS Certificates for Horizon provides examples of setting up TLS certificates for use by Horizon servers. The first scenario shows you how to obtain signed TLS certificates from a Certificate Authority and ensure that the certificates are in a format that can be used by Horizon servers. The second scenario shows you how to configure Horizon servers to off-load TLS connections to an intermediate server.

Intended Audience

This information is intended for anyone who wants to install Horizon and needs to obtain TLS certificates that are used by Horizon servers, or for anyone who uses intermediate servers to off-load TLS connections to Horizon. The information is written for experienced Windows or Linux system administrators who are familiar with virtual machine technology and datacenter operations.

Obtaining TLS Certificates from a Certificate Authority

1

VMware strongly recommends that you configure TLS certificates that are signed by a valid Certificate Authority (CA) for use by Horizon Connection Server instances.

Default TLS certificates are generated when you install Connection Server. Although you can use the default, self-signed certificates for testing purposes, replace them as soon as possible. The default certificates are not signed by a CA. Use of certificates that are not signed by a CA can allow untrusted parties to intercept traffic by masquerading as your server.

In a Horizon environment, replace the default certificate that is installed with vCenter Server with a certificate that is signed by a CA. You can use `openTLS` to perform this task for vCenter Server. For details, see "Replacing vCenter Server Certificates" on the VMware Technical Papers site at http://www.vmware.com/resources/techresources/.

This chapter includes the following topics:

Determining If This Scenario Applies to You

You configure certificates for Horizon by importing the certificates into the Windows local computer certificate store on the Horizon server host.

Before you can import a certificate, you must generate a Certificate Signing Request (CSR) and obtain a valid, signed certificate from a CA. If the CSR is not generated according to the example procedure described in this scenario, the resulting certificate and its private key must be available in a PKCS#12 (formerly called PFX) format file.

There are many ways to obtain TLS certificates from a CA. This scenario shows how to use the Microsoft `certreq` utility to generate a CSR and make a certificate available to a Horizon server. You can use another method if you are familiar with the required tools and they are installed on your server.

Use this scenario to solve the following problems:

- You do not have TLS certificates that are signed by a CA, and you do not know how to obtain them
- You have valid, signed TLS certificates, but they are not in PKCS#12 (PFX) format

If your organization provides you with TLS certificates that are signed by a CA, you can use these certificates. Your organization can use a valid internal CA or a third-party, commercial CA. If your certificates are not in PKCS#12 format, you must convert them. See Convert a Certificate File to PKCS#12 Format.

When you have a signed certificate in the proper format, you can import it into the Windows certificate store and configure a Horizon server to use it. See Set Up an Imported Certificate for a Horizon Server.

Selecting the Correct Certificate Type

You can use various types of TLS certificates with Horizon. Selecting the correct certificate type for your deployment is critical. Different certificate types vary in cost, depending on the number of servers on which they can be used.

Follow VMware security recommendations by using fully qualified domain names (FQDNs) for your certificates, no matter which type you select. Do not use a simple server name or IP address, even for communications within your internal domain.

Single Server Name Certificate

You can generate a certificate with a subject name for a specific server. For example: `dept.company.com`.

This type of certificate is useful if, for example, only one Connection Server instance needs a certificate.

When you submit a certificate signing request to a CA, you provide the server name that will be associated with the certificate. Be sure that the Horizon server can resolve the server name you provide so that it matches the name associated with the certificate.

Subject Alternative Names

A Subject Alternative Name (SAN) is an attribute that can be added to a certificate when it is being issued. You use this attribute to add subject names (URLs) to a certificate so that it can validate more than one server.

For example, a certificate might be issued for a server with the host name `dept.company.com`. You intend the certificate to be used by external users connecting to Horizon through Connection Server. Before the certificate is issued, you can add the SAN `dept-int.company.com` to the certificate to allow the certificate to be used on Connection Server instances behind a load balancer when tunneling is enabled.

Wildcard Certificate

A wildcard certificate is generated so that it can be used for multiple services. For example: `*.company.com`.

A wildcard is useful if many servers need a certificate. If other applications in your environment in addition to Horizon need TLS certificates, you can use a wildcard certificate for those servers, too. However, if you use a wildcard certificate that is shared with other services, the security of the VMware Horizon product also depends on the security of those other services.

Note You can use a wildcard certificate only on a single level of domain. For example, a wildcard certificate with the subject name `*.company.com` can be used for the subdomain `dept.company.com` but not `dept.it.company.com`.

Generating a Certificate Signing Request and Obtaining a Certificate with Microsoft Certreq

To make a certificate available to a Horizon server, you must create a configuration file, generate a certificate signing request (CSR) from the configuration file, and send the signing request to a CA. When the CA returns the certificate, you must import the signed certificate into the Windows local computer certificate store on the Horizon server host, where it joins the previously generated private key.

A CSR can be generated in several ways, depending on how the certificate itself will be generated.

Procedure

1 Create a CSR Configuration File

The Microsoft `certreq` utility uses a configuration file to generate a CSR. You must create a configuration file before you can generate the request. Create the file and generate the CSR on the Windows Server computer that hosts the Horizon server that will use the certificate.

2 Generate a CSR and Request a Signed Certificate from a CA

Using the completed configuration file, you can generate a CSR by running the `certreq` utility. You send the request to a third-party CA, which returns a signed certificate.

3 Verify That the CSR and Its Private Key Are Stored in the Windows Certificate Store

If you use the `certreq` utility to generate a CSR, the utility also generates an associated private key. The utility stores the CSR and private key in the Windows local computer certificate store on the computer on which you generated the CSR. You can confirm that the CSR and private key are properly stored by using the Microsoft Management Console (MMC) Certificate snap-in.

4 Import a Signed Certificate by Using Certreq

When you have a signed certificate from a CA, you can import the certificate into the Windows local computer certificate store on the Horizon server host.

5 Set Up an Imported Certificate for a Horizon Server

After you import a server certificate into the Windows local computer certificate store, you must take additional steps to allow a Horizon server to use the certificate.

Create a CSR Configuration File

The Microsoft `certreq` utility uses a configuration file to generate a CSR. You must create a configuration file before you can generate the request. Create the file and generate the CSR on the Windows Server computer that hosts the Horizon server that will use the certificate.

Prerequisites

Gather the information that you need to fill out the configuration file. You must know the FQDN of the Horizon server and the organizational unit, organization, city, state, and country to complete the Subject name.

Procedure

1 Open a text editor and paste the following text, including the beginning and ending tags, into the file.

```
;----------------- request.inf -----------------

[Version]

Signature="$Windows NT$"

[NewRequest]

Subject = "CN=View_Server_FQDN, OU=Organizational_Unit, O=Organization, L=City, S=State,
C=Country"
; Replace View_Server_FQDN with the FQDN of the Horizon server.
; Replace the remaining Subject attributes.
KeySpec = 1
KeyLength = 2048
; KeyLength is usually chosen from 2048, 3072, or 4096. A KeyLength
; of 1024 is also supported, but it is not recommended.
HashAlgorithm = SHA256
; Algorithms earlier than SHA-2 are insufficiently secure and are not recommended.
Exportable = TRUE
MachineKeySet = TRUE
SMIME = False
PrivateKeyArchive = FALSE
UserProtected = FALSE
UseExistingKeySet = FALSE
ProviderName = "Microsoft RSA SChannel Cryptographic Provider"
ProviderType = 12
RequestType = PKCS10
KeyUsage = 0xa0

[EnhancedKeyUsageExtension]
```

```
OID=1.3.6.1.5.5.7.3.1 ; this is for Server Authentication

;-----------------------------------------------
```

If an extra CR/LF character is added to the `Subject =` line when you copy and paste the text, delete the CR/LF character.

2 Update the `Subject` attributes with appropriate values for your Horizon server and deployment.

For example: `CN=dept.company.com`

To comply with VMware security recommendations, use the fully qualified domain name (FQDN) that client devices use to connect to the host. Do not use a simple server name or IP address, even for communications within your internal domain.

Some CAs do not allow you to use abbreviations for the state attribute.

3 (Optional) Update the `Keylength` attribute.

The default value, 2048, is adequate unless you specifically need a different `KeyLength` size. Many CAs require a minimum value of 2048. Larger key sizes are more secure but have a greater impact on performance.

A `KeyLength` of 1024 is also supported, although the National Institute of Standards and Technology (NIST) recommends against keys of this size, as computers continue to become more powerful and can potentially crack stronger encryption.

Important Do not generate a `KeyLength` value under 1024. Horizon Client for Windows will not validate a certificate on a Horizon server that was generated with a `KeyLength` under 1024, and the Horizon Client devices will fail to connect to Horizon. Certificate validations that are performed by Connection Server will also fail, resulting in the affected Horizon servers showing as red in the Horizon Console dashboard.

4 Save the file as `request.inf`.

What to do next

Generate a CSR from the configuration file.

Generate a CSR and Request a Signed Certificate from a CA

Using the completed configuration file, you can generate a CSR by running the `certreq` utility. You send the request to a third-party CA, which returns a signed certificate.

Prerequisites

- Verify that you completed a CSR configuration file. See Create a CSR Configuration File.
- Perform the `certreq` operation described in this procedure on the computer where the CSR configuration file is located.

Procedure

1 Open a command prompt by right-clicking on **Command Prompt** in the **Start** menu and selecting **Run as administrator**.

2 Navigate to the directory where you saved the `request.inf` file.

For example: **`cd c:\certificates`**

3 Generate the CSR file.

For example: **`certreq –new request.inf certreq.txt`**

4 Use the contents of the CSR file to submit a certificate request to the CA in accordance with the CA's enrollment process.

a When you submit the request to a CA, the CA prompts you to select the type of server on which you will install the certificate. Since Horizon uses the Microsoft Certificates MMC to manage certificates, select a certificate for a server type of Microsoft, Microsoft IIS 7, or something similar. The CA should produce a certificate in the format needed to work with Horizon.

b If you request a single server name certificate, use a name that Horizon Client devices can resolve into an IP address for this Horizon server. The name that computers use to connect to the Horizon server should match the name associated with the certificate.

Note The CA might require that you copy and paste the contents of the CSR file (such as `certreq.txt`) into a Web form. Using a text editor, you can copy the contents of the CSR file. Be sure to include the beginning and ending tags. For example:

```
-----BEGIN NEW CERTIFICATE REQUEST-----
MIID2jCCAsICAQAwazEWMBQGA1UEBhMNVW5pdGVkIFN0YXRlczELMAkGA1UECAwC
Q0ExEjAQBgNVBAcMCVBhbG8gQWx0bzEKMAgGA1UECgwBTzELMAkGA1UECwwCT1Ux
FzAVBgNVBAMMDm15LmNvbXBhbnkuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8A
. . .
. . .
L9nPYX76jeu5rwQfXLivSCea6nZiIOZYw8Dbn8dgwAqpJdzBbrwuM1TuSnx6bAK8
S52Tv0GxW58jUTtxFV+Roz8TE8wZDFB51jx+FmLs
-----END NEW CERTIFICATE REQUEST-----
```

After conducting some checks on your company, the CA creates a server certificate based on the information in the CSR, signs it with its private key, and sends you the certificate.

The CA also sends you a root CA certificate and, if applicable, an intermediate CA certificate.

5 Rename the certificate text file to `cert.cer`.

Make sure that the file is located on the Horizon server on which the certificate request was generated.

6 Rename the root CA and intermediate CA certificate files to `root.cer` and `intermediate.cer`.

Make sure that the files are located on the Horizon server on which the certificate request was generated.

Note These certificates do not have to be in PKCS#12 (PFX) format when you use the `certreq` utility to import the certificates into the Windows local computer certificate store. PKCS#12 (PFX) format is required when you use the Certificate Import wizard to import certificates into the Windows certificate store.

What to do next

Verify that the CSR file and its private key were stored in the Windows local computer certificate store.

Verify That the CSR and Its Private Key Are Stored in the Windows Certificate Store

If you use the `certreq` utility to generate a CSR, the utility also generates an associated private key. The utility stores the CSR and private key in the Windows local computer certificate store on the computer on which you generated the CSR. You can confirm that the CSR and private key are properly stored by using the Microsoft Management Console (MMC) Certificate snap-in.

The private key must later be joined with the signed certificate to enable the certificate to be properly imported and used by a Horizon server.

Prerequisites

- Verify that you generated a CSR by using the `certreq` utility and requested a signed certificate from a CA. See Generate a CSR and Request a Signed Certificate from a CA.
- Familiarize yourself with the procedure for adding a Certificate snap-in to the Microsoft Management Console (MMC). See "Add the Certificate Snap-in to MMC" in the chapter, "Configuring TLS Certificates for Horizon Servers," in the *Horizon Installation* document.

Procedure

1 On the Windows Server computer, add the Certificate snap-in to MMC.

2 In the MMC window on the Windows Server computer, expand the **Certificates (Local Computer)** node and select the **Certificate Enrollment Request** folder.

3 Expand the **Certificate Enrollment Request** folder and select the **Certificates** folder.

4 Verify that the certificate entry is displayed in the **Certificates** folder.

The **Issued To** and **Issued By** fields must show the domain name that you entered in the **`subject:CN`** field of the `request.inf` file that was used to generate the CSR.

5 Verify that the certificate contains a private key by taking one of the following steps:

- Verify that a yellow key appears on the certificate icon.

- Double-click the certificate and verify that the following statement appears in the Certificate Information dialog box: `You have a private key that corresponds to this certificate.`.

What to do next

Import the certificate into the Windows local computer certificate store.

Import a Signed Certificate by Using Certreq

When you have a signed certificate from a CA, you can import the certificate into the Windows local computer certificate store on the Horizon server host.

If you used the `certreq` utility to generate a CSR, the certificate private key is local to the server on which you generated the CSR. To work correctly, the certificate must be combined with the private key. Use the `certreq` command shown in this procedure to ensure that the certificate and private key are properly combined and imported into the Windows certificate store.

If you use another method to obtain a signed certificate from a CA, you can use the Certificate Import wizard in the Microsoft Management Console (MMC) Snap-in to import a certificate into the Windows certificate store. This method is described in "Configuring TLS Certificates for Horizon Servers" in the *Horizon Installation* document.

Prerequisites

- Verify that you received a signed certificate from a CA. See Generate a CSR and Request a Signed Certificate from a CA.
- Perform the `certreq` operation described in this procedure on the computer on which you generated a CSR and stored the signed certificate.

Procedure

1. Open a command prompt by right-clicking on **Command Prompt** in the **Start** menu and selecting **Run as administrator**.
2. Navigate to the directory where you saved the signed certificate file such as `cert.cer`.

   For example: **`cd c:\certificates`**
3. Import the signed certificate by running the `certreq –accept` command.

   For example: **`certreq –accept cert.cer`**

Results

The certificate is imported into the Windows local computer certificate store.

What to do next

Configure the imported certificate to be used by a Horizon server. See Set Up an Imported Certificate for a Horizon Server.

Set Up an Imported Certificate for a Horizon Server

After you import a server certificate into the Windows local computer certificate store, you must take additional steps to allow a Horizon server to use the certificate.

Procedure

1. Verify that the server certificate was imported successfully.
2. Change the certificate Friendly name to **vdm**.

   vdm must be lower case. Any other certificates with the Friendly name **vdm** must be renamed, or you must remove the Friendly name from those certificates.
3. Install the root CA certificate and intermediate CA certificate in the Windows certificate store.
4. Restart the Connection Server service to allow the service to start using the new certificates.
5. If you use HTML Access, restart the Blast Secure Gateway service.

Results

To perform the tasks in this procedure, see the following topics:

- Modify the Certificate Friendly Name
- Import the Root and Intermediate Certificates into the Windows Certificate Store

For more information, see "Configure Connection Server to Use a New TLS Certificate" in the *Horizon Installation* document.

Note The *Horizon Installation* topic "Import a Signed Server Certificate into a Windows Certificate Store" is not listed here because you already imported the server certificate by using the `certreq` utility. You should not use the Certificate Import wizard in the MMC Snap-in to import the server certificate again.

However, you can use the Certificate Import wizard to import the root CA certificate and intermediate CA certificate into the Windows certificate store.

Off-loading TLS Connections to Intermediate Servers

2

You can set up intermediate servers between your Horizon servers and Horizon Client devices to perform tasks such as load balancing and off-loading TLS connections. Horizon Client devices connect over HTTPS to the intermediate servers, which pass on the connections to the external-facing Connection Server instances.

To off-load TLS connections to an intermediate server, you must complete a few key tasks:

- Import the TLS certificate that is used by the intermediate server to your external-facing Horizon servers.
- Set the External URLs on your external-facing Horizon servers to match the URL that clients can use to connect to the intermediate server.
- Allow HTTP connections between the intermediate server and the Horizon servers.

This chapter includes the following topics:

Import TLS Off-loading Servers' Certificates to Horizon Servers

If you off-load TLS connections to an intermediate server, you must import the intermediate server's certificate onto the Connection Server instances that connect to the intermediate server. The same TLS server certificate must reside on both the off-loading intermediate server and each off-loaded Horizon server that connects to the intermediate server.

If you have a mixed network environment with some intermediate servers and some external-facing Connection Server instances, the intermediate server and any Connection Server instances that connect to it must have the same TLS certificate.

If the intermediate server's certificate is not installed on the Connection Server instance, clients cannot validate their connections to Horizon. In this situation, the certificate thumbprint sent by the Horizon server does not match the certificate on the intermediate server to which Horizon Client connects.

Do not confuse load balancing with TLS off-loading. The preceding requirement applies to any device that is configured to provide TLS off-loading, including some types of load balancers. However, pure load balancing does not require copying of certificates between devices.

Important The scenario described in the following topics shows one approach to the sharing of TLS certificates between third-party components and VMware components. This approach may not suit everyone and it is not the only way to perform the task.

Procedure

1 Download an TLS Certificate from the Intermediate Server

   You must download the CA-signed TLS certificate that is installed on the intermediate server so that it can be imported into the external-facing Horizon servers.

2 Download a Private Key from the Intermediate Server

   You must download the private key that is associated with the TLS certificate on the intermediate server. The private key must be imported with the certificate into the Horizon servers.

3 Convert a Certificate File to PKCS#12 Format

   If you obtained a certificate and its private key in PEM or another format, you must convert it to PKCS#12 (PFX) format before you can import the certificate into a Windows certificate store on a Horizon server. PKCS#12 (PFX) format is required if you use the Certificate Import wizard in the Windows certificate store.

4 Import a Signed Server Certificate into a Windows Certificate Store

   You must import the TLS server certificate into the Windows local computer certificate store on the Windows Server host on which Connection Server is installed.

5 Modify the Certificate Friendly Name

   To configure a Connection Server instance to recognize and use an TLS certificate, you must modify the certificate Friendly name to `vdm`.

6 Import the Root and Intermediate Certificates into the Windows Certificate Store

   You must import the root certificate and any intermediate certificates in the certificate chain into the Windows local computer certificate store.

Download an TLS Certificate from the Intermediate Server

You must download the CA-signed TLS certificate that is installed on the intermediate server so that it can be imported into the external-facing Horizon servers.

Procedure

1 Connect to the intermediate server and find the TLS certificates that are presented to clients sending HTTPS requests.

2 Find and download the TLS certificate that is used for Horizon.

Example: Download an TLS Certificate from an F5 BIG-IP LTM System

This example uses F5 BIG-IP Local Traffic Manager (LTM) as an intermediate server. The example is intended to give you a general idea of how you might download a certificate from your own intermediate server.

Important These steps are specific to F5 BIG-IP LTM and may not apply to new releases or other F5 products. The steps do not apply to other vendors' intermediate servers.

Before you start, verify that the F5 BIG-IP LTM system is deployed with Horizon. Check that you completed the tasks in the F5 deployment guide, *Deploying the BIG-IP LTM System with VMware View*, located at http://www.f5.com/pdf/deployment-guides/f5-vmware-view-dg.pdf.

1 Connect to the F5 BIG-IP LTM configuration utility.

2 On the Main tab of the navigation pane, expand **Local Traffic** and click **SSL certificates**.

The utility displays a list of certificates that are installed on the system.

3 In the Name column, click the name of the certificate that is used for Horizon.

4 At the bottom of the screen, click **Export**.

The utility displays the existing TLS certificate in the **Certificate Text** box.

5 From the **Certificate File** setting, click **Download *file_name***.

The TLS certificate is downloaded as a CRT file.

Download a Private Key from the Intermediate Server

You must download the private key that is associated with the TLS certificate on the intermediate server. The private key must be imported with the certificate into the Horizon servers.

Procedure

1 Connect to the intermediate server and find the TLS certificates that are presented to clients sending HTTPS requests.

2 Find the certificate that is used for Horizon and download its private key.

Example: Download a Private Key from a F5 BIG-IP LTM System

This example uses F5 BIG-IP Local Traffic Manager (LTM) as an intermediate server. The example is intended to give you a general idea of how you might download a private key from your own intermediate server.

Important These steps are specific to F5 BIG-IP LTM and may not apply to new releases or other F5 products. The steps do not apply to other vendors' intermediate servers.

Before you start, verify that you are connected to the F5 BIG-IP LTM configuration utility.

1 On the Main tab of the navigation pane, expand **Local Traffic** and click **SSL certificates**.

The utility displays a list of certificates installed on the system.

2 In the Name column, click the name of the certificate that is used for Horizon.

3 On the menu bar, click **Key**.

4 At the bottom of the screen, click **Export**.

The utility displays the existing private key in the **Key Text** box.

5 From the Key File setting, click **Download *file_name*.**.

The private key is downloaded as a KEY file.

Convert a Certificate File to PKCS#12 Format

If you obtained a certificate and its private key in PEM or another format, you must convert it to PKCS#12 (PFX) format before you can import the certificate into a Windows certificate store on a Horizon server. PKCS#12 (PFX) format is required if you use the Certificate Import wizard in the Windows certificate store.

You might obtain certificate files in one of these ways:

- You obtain a certificate keystore file from a CA.
- You download a certificate and its private key from an intermediate server that is set up in your Horizon deployment.
- Your organization provides you with certificate files.

Certificate files come in various formats. For example, PEM format is often used in a Linux environment. Your files might have a certificate file, key file, and CSR file with the following extensions:

```
server.crt
server.csr
server.key
```

The CRT file contains the SSL certificate that was returned by the CA. The CSR file is the original certificate signing request file and is not needed. The KEY file contains the private key.

Prerequisites

- Verify that OpenSSL is installed on the system. You can download `openssl` from `http://www.openssl.org`.
- Verify that the root certificate of the SSL certificate that was returned by the CA is also available on the system.

Procedure

1. Copy the CRT and KEY files to the OpenSSL installation directory.

   For example: `cd c:\OpenSSL-Win32\bin`

2. Open a Windows command prompt and, if necessary, navigate to the OpenSSL installation directory.
3. Generate a PKCS#12 (PFX) keystore file from the certificate file and your private key.

   For example: `openssl pkcs12 -export -out server.p12 -inkey server.key -in server.crt -certfile CACert.crt`

   In this example, `CACert.crt` is the name of the root certificate that was returned by the certificate authority.

   The Windows certificate store also accepts a keystore that is generated with a PFX extension. For example: `-out server.pfx`
4. Type an export password to protect the PKCS#12 (PFX) file.

Import a Signed Server Certificate into a Windows Certificate Store

You must import the TLS server certificate into the Windows local computer certificate store on the Windows Server host on which Connection Server is installed.

This scenario uses a certificate file in PKCS#12 (PFX) format.

Depending on your certificate file format, the entire certificate chain that is contained in the keystore file might be imported into the Windows local computer certificate store. For example, the server certificate, intermediate certificate, and root certificate might be imported.

For other types of certificate files, only the server certificate is imported into the Windows local computer certificate store. In this case, you must take separate steps to import the root certificate and any intermediate certificates in the certificate chain.

For more information about certificates, consult the Microsoft online help available with the Certificate snap-in to MMC.

Prerequisites

Verify that the TLS server certificate is in PKCS@12 (PFX) format. See Convert a Certificate File to PKCS#12 Format.

Procedure

1. In the MMC window on the Windows Server host, expand the **Certificates (Local Computer)** node and select the **Personal** folder.
2. In the Actions pane, go to **More Actions > All Tasks > Import**.
3. In the **Certificate Import** wizard, click **Next** and browse to the location where the certificate is stored.
4. Select the certificate file and click **Open**.

   To display your certificate file type, you can select its file format from the **File name** drop-down menu.
5. Type the password for the private key that is included in the certificate file.
6. Select **Mark this key as exportable**.
7. Select **Include all extended properties**.
8. Click **Next** and click **Finish**.

   The new certificate appears in the **Certificates (Local Computer) > Personal > Certificates** folder.
9. Verify that the new certificate contains a private key.
   a. In the **Certificates (Local Computer) > Personal > Certificates** folder, double-click the new certificate.
   b. In the General tab of the Certificate Information dialog box, verify that the following statement appears: `You have a private key that corresponds to this certificate`.

What to do next

Modify the certificate Friendly name to **vdm**.

Modify the Certificate Friendly Name

To configure a Connection Server instance to recognize and use an TLS certificate, you must modify the certificate Friendly name to `vdm`.

Prerequisites

Verify that the server certificate is imported into the **Certificates (Local Computer) > Personal > Certificates** folder in the Windows Certificate Store. See Import a Signed Server Certificate into a Windows Certificate Store.

Procedure

1. In the MMC window on the Windows Server host, expand the **Certificates (Local Computer)** node and select the **Personal > Certificates** folder.

2 Right-click the certificate that is issued to the VMware Horizon server host and click **Properties**.

3 On the General tab, delete the **Friendly name** text and type **vdm**.

4 Click **Apply** and click **OK**.

5 Verify that no other server certificates in the **Personal > Certificates** folder have a Friendly name of **vdm**.

   a Locate any other server certificate, right-click the certificate, and click **Properties**.

   b If the certificate has a Friendly name of **vdm**, delete the name, click **Apply**, and click **OK**.

What to do next

Import the root certificate and intermediate certificates into the Windows local computer certificate store.

After all certificates in the chain are imported, you must restart the Connection Server service to make your changes take effect.

Import the Root and Intermediate Certificates into the Windows Certificate Store

You must import the root certificate and any intermediate certificates in the certificate chain into the Windows local computer certificate store.

If the TLS server certificate that you imported from the intermediate server is signed by a root CA that is known and trusted by the Connection Server host, and there are no intermediate certificates in your certificate chains, you can skip this task. Commonly used Certificate Authorities are likely to be trusted by the host.

Procedure

1 In the MMC console on the Windows Server host, expand the **Certificates (Local Computer)** node and go to the **Trusted Root Certification Authorities > Certificates** folder.

   - If your root certificate is in this folder, and there are no intermediate certificates in your certificate chain, skip to step 7.
   - If your root certificate is in this folder, and there are intermediate certificates in your certificate chain, skip to step 6.
   - If your root certificate is not in this folder, proceed to step 2.

2 Right-click the **Trusted Root Certification Authorities > Certificates** folder and click **All Tasks > Import**.

3 In the **Certificate Import** wizard, click **Next** and browse to the location where the root CA certificate is stored.

4 Select the root CA certificate file and click **Open**.

5 Click **Next**, click **Next**, and click **Finish**.

6 If your server certificate was signed by an intermediate CA, import all intermediate certificates in the certificate chain into the Windows local computer certificate store.

   a Go to the **Certificates (Local Computer) > Intermediate Certification Authorities > Certificates** folder.

   b Repeat steps 3 through 6 for each intermediate certificate that must be imported.

7 Restart the Connection Server service to make your changes take effect.

8 If you use HTML Access, restart the Blast Secure Gateway service.

Set Horizon Server External URLs to Point Clients to TLS Off-loading Servers

If TLS is off-loaded to an intermediate server and Horizon Client devices use the secure tunnel to connect to Horizon, you must set the secure tunnel external URL to an address that clients can use to access the intermediate server.

You configure the external URL settings on the Connection Server instance that connects to the intermediate server.

If you have a mixed network environment with some intermediate servers and some external-facing Connection Server instances, External URLs are required for any Connection Server instances that connect to the intermediate server.

Note You cannot off-load TLS connections from a PCoIP Secure Gateway (PSG) or Blast Secure Gateway. The PCoIP external URL and Blast Secure Gateway external URL must allow clients to connect to the computer that hosts the PSG and Blast Secure Gateway. Do not reset the PCoIP external URL and Blast external URL to point to the intermediate server unless you plan to require TLS connections between the intermediate server and the Horizon server.

Set the External URLs for a Connection Server Instance

You use Horizon Console to configure the external URLs for a Connection Server instance.

Prerequisites

- Verify that the secure tunnel connections are enabled on the Connection Server instance.

Procedure

1 In Horizon Console, click **Settings > Servers**.

2 Select **Connection Servers**, select a Connection Server instance, and click **Edit**.

3 Type the secure tunnel external URL in the **External URL** text box.

   The URL must contain the protocol, client-resolvable host name and port number.

For example: **https://myserver.example.com:443**

Note You can use the IP address if you have to access a Connection Server instance when the host name is not resolvable. However, the host that you contact will not match the TLS certificate that is configured for the Connection Server instance, resulting in blocked access or access with reduced security.

4 Verify that all addresses in this dialog allow client systems to reach this Connection Server instance.

5 Click **OK**.

Allow HTTP Connections From Intermediate Servers

When TLS is off-loaded to an intermediate server, you can configure Connection Server instances to allow HTTP connections from the client-facing, intermediate devices. The intermediate devices must accept HTTPS for Horizon Client connections.

To allow HTTP connections between Horizon servers and intermediate devices, you must configure the `locked.properties` file on each Connection Server instance on which HTTP connections are allowed.

Even when HTTP connections between Horizon servers and intermediate devices are allowed, you cannot disable TLS in Horizon. Horizon servers continue to accept HTTPS connections as well as HTTP connections.

Note If your Horizon clients use smart card authentication, the clients must make HTTPS connections directly to Connection Server. TLS off-loading is not supported with smart card authentication.

Procedure

1 Create or edit the `locked.properties` file in the TLS/SSL gateway configuration folder on the Connection Server host.

For example: *`install_directory`*`\VMware\VMware View\Server\SSLgateway\conf\locked.properties`

2 To configure the Horizon server's protocol, add the `serverProtocol` property and set it to `http`.

The value `http` must be typed in lower case.

3 (Optional) Add properties to configure a non-default HTTP listening port and a network interface on the Horizon server.

- To change the HTTP listening port from 80, set `serverPortNonTLS` to another port number to which the intermediate device is configured to connect.

- If the Horizon server has more than one network interface, and you intend the server to listen for HTTP connections on only one interface, set `serverHostNonTLS` to the IP address of that network interface.

4 Save the `locked.properties` file.

5 Restart the Connection Server service to make your changes take effect.

Example: locked.properties file

This file allows non-TLS HTTP connections to a Horizon server. The IP address of the Horizon server's client-facing network interface is 10.20.30.40. The server uses the default port 80 to listen for HTTP connections. The value `http` must be lower case.

```
serverProtocol=http
serverHostNonTLS=10.20.30.40
```