

Replacing vCenter Server Certificates

VMware vSphere 4.1

Certificates are automatically generated when you install vCenter Server. These default certificates are not signed by a commercial certificate authority (CA) and may not provide strong security. You can replace default vCenter Server certificates with certificates signed by a commercial CA.

This Technical Note includes the following topics:

- [“About vCenter Server Certificates”](#) on page 1
- [“Pre-Trusting Server Certificates”](#) on page 2
- [“Certificate Specifications”](#) on page 2
- [“Certificate Locations”](#) on page 2
- [“Replacing Default Server Certificates with Certificates Signed by a Commercial CA”](#) on page 3
- [“Replacing Default Server Certificates with Self-Signed Certificates”](#) on page 6
- [“Related Publications”](#) on page 8

NOTE If you have replaced the default vCenter Server certificates with certificates signed by a commercial CA, you do not need to perform the tasks in this document. You can configure server-certificate verification settings using the vSphere Client. See the *Datacenter Administration Guide* for more information.

About vCenter Server Certificates

VMware products use standard X.509 version 3 (X.509v3) certificates to encrypt session information sent over Secure Socket Layer (SSL) protocol connections between components.

For example, communications between a vCenter Server system and each ESX/ESXi host that it manages are encrypted, and some features, such as VMware Fault Tolerance, require the certificate verification provided by SSL. The authenticity of the certificate presented during the SSL handshake phase (prior to encryption), is verified by the client, which protects against “man-in-the-middle” attacks.

In new installations of vCenter Server 4.1, host certificate verification is enabled by default. Do not disable certificate verification. If a host’s certificate cannot be verified for some reason, verification can be temporarily disabled to help determine the cause of the problem.

For environments that require strong security, perform the following tasks:

- Install certificates signed by a commercial Certificate Authority (CA) on all vCenter Server systems and ESX/ESXi hosts.
- Upgrade existing VirtualCenter Server and Virtual Infrastructure Client deployments to vCenter Server 4.1 and vSphere Client 4.1.

When you replace default vCenter Server certificates, the certificates you obtain for your servers must meet the specifications described in [“Certificate Specifications”](#) on page 2.

Pre-Trusting Server Certificates

Certificates signed by a commercial certificate authority, such as Entrust or Verisign, are pre-trusted on the Windows operating system. However, if you replace a certificate with one signed by your own local root CA, or if you plan to continue using a default certificate, you must pre-trust the certificate by importing it into the local certificate store for each vSphere Client instance.

You must pre-trust all certificates that are signed by your own local root CA, unless you pre-trust the parent certificate, the root CA's own certificate. You will also have to pre-trust any valid default certificates that you will continue to use on vCenter Server.

Certificate Specifications

VMware products use standard X.509 version 3 (X.509v3) certificates. If you replace the default certificate, you must replace it with a signed certificate that conforms to the Privacy Enhanced Mail (PEM), a key format that stores data in a Base-64 encoded Distinguished Encoding Rules (DER) format.

The key used to sign the certificates must be a standard RSA key with an encryption length ranging from 512 to 2048 bits. The recommended length is 1024 bits.

The key and certificate names for vCenter Server are shown in [Table 1](#). The syntax examples create certificates and keys in the required format. Personal Information Exchange Format (PEM) enables transfer of certificates and their private keys from one computer to another or to removable media. The Microsoft Windows CryptoAPI uses the PFX format (also known as PKCS #12).

Table 1. Names of Key and Certificate Files

Server	Private Key	Certificate	PFX
vCenter Server 4.0	rui.key	rui.crt	rui.pfx

Certificate Locations

The directory locations of the keys and certificates are shown in [Table 2](#).

Table 2. Default Locations for vCenter Server Certificates

Server	Directory Location for Certificate
vCenter Server 4.0	C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\SSL\ For Windows Server 2008, C:\Program Data\VMware\VMware VirtualCenter\SSL\

The process for generating keys and certificates described in this document is the same for Windows or Linux, although the precise syntax is platform specific.

Replacing Default Server Certificates with Certificates Signed by a Commercial CA

When you replace default server certificates in a production environment, deploy new certificates in stages, rather than all at the same time. Be sure you understand the full scope of the process as it applies to your specific environment before taking any actions.

NOTE Allow time to obtain certificates from a commercial CA before starting this process.

To replace a server certificate

- 1 [“Edit the OpenSSL Configuration File”](#) on page 3
- 2 [“Create Certificate-Signing Requests for vCenter Server”](#) on page 4
- 3 [“Load Replacement Certificates into Memory”](#) on page 4

Some details might not apply to every deployment.

Edit the OpenSSL Configuration File

VMware products implement the OpenSSL libraries and toolkits to generate the default certificates that are created during installation process. You can use OpenSSL to create certificate-signing requests (CSRs). You can download OpenSSL from <http://www.openssl.org>.

NOTE VMware strongly recommends that you create CSRs and other security-related artifacts on trusted, air-gapped physical hardware over which you have complete control. VMware also recommends using a hardware RNG (random-number generator) to efficiently and quickly generate random numbers that have the appropriate characteristics (sufficient degree of entropy, for example) for cryptographic purposes.

The examples shown in this document are run from a Windows host machine and assume that the OpenSSL home directory is `c:\openssl\bin`.

The default OpenSSL installation includes a configuration file, `openssl.cnf`, located in the `\bin` directory. You can preconfigure many settings in this configuration file, and you can overwrite many default values by passing values to the command line. The syntax examples in the remainder of this document assume the following settings in the OpenSSL configuration file.

- The `$dir` variable is set to the local (`.`) directory path.
- The `[req]` section of the `openssl.cnf` has a `default_keyfile` variable set to `$dir/rui.key`.
- The `[CA]` section references a `CA_default` section.
- The `[CA_default]` section references a `private_key` named `myroot.key`.

To create or modify OpenSSL configuration file for your environment

- 1 Navigate to the OpenSSL directory.
- 2 Edit the OpenSSL configuration file (`openssl.cnf`), specifying the details appropriate for your environment.

Create Certificate-Signing Requests for vCenter Server

You must generate a certificate-signing requests (CSR) for each system that requires a replacement certificate.

Before you begin this task, edit your OpenSSL configuration file (`openssl.cnf`) to suit your environment as described in “[Edit the OpenSSL Configuration File](#)” on page 3.

Refer to the OpenSSL documentation at <http://www.openssl.org> for more information about OpenSSL commands and options.

To create certificate-signing requests

- 1 Generate the RSA key for the vCenter Server system and the CSR. For example:

```
openssl req -new -nodes -out mycsr.csr -config openssl.cnf
```

- 2 When prompted, specify the fully qualified host name as the system's `commonName`.

- 3 Send the certificate request to the commercial certificate authority of your choice (for example, Entrust or Verisign) and await the return of the signed certificate.

Or, sign the request using your local root certificate authority:

```
openssl ca -out rui.crt -config openssl.cnf -infiles mycsr.csr
```

You will be prompted for the password to the root key. After executing this command, you should have a new generated (and signed) `rui.crt` for the specified system, and the private key for the system (`rui.key`).

Create the PFX File

The `rui.pfx` file is a concatenation of the system's certificate and private key, exported in the PFX format. This file is then copied to the subdirectory on the vCenter Server system.

To create the PFX file

- Export the certificate and key file together to PFX format using OpenSSL. For example:

```
openssl pkcs12 -export -in rui.crt -inkey rui.key -name rui -passout pass:testpassword -out rui.pfx
```

Load Replacement Certificates into Memory

Before you begin this procedure, use a browser to connect to the vCenter Server system and view the existing certificate. The method to view the certificate varies depending on the browser you are using. Refer to your browser's documentation for more information.

The following procedure assumes that you have acquired or generated the following files:

- X.509 certificate file with RSA public key in PEM format, named `rui.crt`
- RSA private key in PEM format, named `rui.key`
- PKCS12 bundle of the same certificate and key, named `rui.pfx`

To load the replacement certificates into memory

- 1 On the server system, locate the SSL directory for vCenter Server.
 - For Windows 2008, the location is typically `C:\Program Data\VMware\VMware VirtualCenter\SSL`.
 - For Windows 2003, the location is typically `C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\SSL`.
- 2 Back up the three existing certificate files: `rui.crt`, `rui.key`, and `rui.pfx`.
- 3 Copy the new certificate files into the SSL directory, overwriting the existing certificates.

- 4 Using a browser on the vSphere server system, connect to `http://localhost/mob/?moid=vpxd-securitymanager&vmodl=1`
If you use a browser on another system, connect to `https://<vSphere_server_system>/mob/?moid=vpxd-securitymanager&vmodl=1`
- 5 Enter the administrator name and password for the vCenter Server system.
A Web page appears in the browser window: Managed Object Type: vpxSecurityManager.
- 6 On the Web page, under Methods, click **reloadSslCertificate**.
A pop-up browser window appears.
- 7 Click **Invoke Method**.
A message appears on the Web page: Method Invocation Result: void.
- 8 On the vCenter Server system, restart the service VMware VirtualCenter Management Webservices.
Linked Mode and other features will not function if you do not restart this service. Because the certificate thumbprint is published as Linked Mode shared information, it may take some time to replicate to the other vCenter Server instances in the Linked Mode group.
You do not need to restart vCenter Server.
- 9 Refresh the page in the browser window you opened in [Step 4](#).
If you closed the browser window, connect to the vCenter Server again using the URL in [Step 4](#).
- 10 Verify that the new certificate is installed.
If you installed the new certificate successfully, all host passwords and the database password have been re-encrypted using the new certificate. If the new certificate was not installed successfully, see [“Troubleshooting”](#) on page 5 for more information.

What to do next

Back up the old certificates on another system and remove them from the vCenter Server machine.

Troubleshooting

If you are unable to restart vCenter Server or if the old certificate still appears to be in use, you can troubleshoot the problem.

vCenter Server cannot connect to the vCenter Server database

If vCenter Server is unable to connect to the vCenter Server database, and therefore cannot be restarted, you can reset the database password by running to the following command:

```
vpxd -P pwd
```

vCenter Server cannot connect to managed hosts after it is restarted

As the host root user, reconnect each host to vCenter Server.

The new vCenter Server certificate does not appear to load

Existing open connections to vCenter Server are not forcibly closed and might still use the old certificate. To force all connections to use the new certificate, use one of the following methods:

- Restart the network stack or network interfaces on the server.
- Restart the vCenter Server service.

Replacing Default Server Certificates with Self-Signed Certificates

VMware recommends that you replace default certificates with those signed by a commercial certificate authority. If you choose to replace vCenter Server certificates with self-signed certificates, perform the following tasks in the order listed.

To replace default certificates with self-signed certificates

- 1 Read [“Using OpenSSL to Create Security Artifacts”](#) on page 6.
- 2 [“Edit the OpenSSL Configuration File”](#) on page 3.
- 3 [“Create a Local Root CA”](#) on page 6.
- 4 [“Create Certificate-Signing Requests for vCenter Server”](#) on page 4.
- 5 [“Create Self-Signed Certificates”](#) on page 6.
- 6 [“Create the PFX File”](#) on page 4.
- 7 [“Load Replacement Certificates into Memory”](#) on page 4.
- 8 [“Install Certificates on Windows Client Hosts”](#) on page 7.

Some details might not apply to every deployment.

Using OpenSSL to Create Security Artifacts

VMware products implement the OpenSSL libraries and toolkits to generate the default certificates created during installation process. VMware strongly recommends that you install certificates signed by a commercial Certificate Authority (CA). However, you have the option to use OpenSSL to create new keys and certificates and a root CA (if appropriate). You can download OpenSSL from <http://www.openssl.org>.

If you intend to create your own root CA and keys, properly secure the host system used to create local root CA certificate and its private key. The private key associated with the root CA must remain private.

NOTE VMware strongly recommends creating keys, CSRs, and other security-related artifacts on trusted, air-gapped physical hardware over which you have complete control. VMware also recommends using a hardware RNG (random-number generator) to efficiently and quickly generate random numbers that have the appropriate characteristics (sufficient degree of entropy, for example) for cryptographic purposes.

The examples shown are run from a Windows host machine and assume that the OpenSSL home directory is: `c:\openssl\bin`.

Inside the `openssl\bin` directory, you can create subdirectories to contain your keys, certificates, and other files. The syntax examples shown in the following subsections assume a flat directory structure.

NOTE The following instructions assume that a single self-signed root CA is used to sign all certificate signing requests (CSRs).

Create a Local Root CA

To replace the default certificates with certificates signed by your own local CA, you must create a root CA. The root CA's certificate must then be installed in any client systems that will connect to the managed hosts. Assuming you use the same root CA key to sign all the CSRs, you will have only one root CA certificate to install in the Windows clients.

The following example creates a new root CA and an RSA key:

```
C:\OpenSSL\bin>openssl req -new -x509 -extensions v3_ca -keyout myroot.key -out myroot.crt -days
3650 -config openssl.cnf
```

Create Self-Signed Certificates

If you choose to install self-signed certificates, you can create them using OpenSSL.

To create a self-signed certificate

- 1 Create a text file named `openssl.cnf` with the configuration settings shown in the following example.

NOTE Modify all entries so they are specific to your environment. Providing the `commonName` is mandatory.

```
[ req ]
default_bits           = 1024
default_keyfile        = rui.key
distinguished_name     = req_distinguished_name
#Don't encrypt the key
encrypt_key            = no
prompt                 = no
string_mask = nombstr

[ req_distinguished_name ]
countryName            = US
stateOrProvinceName   = California
localityName           = Palo Alto
0.organizationName     = VMware, Inc.
emailAddress           = ssl-certificates@vmware.com
commonName              = <NAME_OF_SERVER_THAT_WILL_HAVE_CERTIFICATE>
```

- 2 Create the self-signed certificate (`rui.key` and `rui.crt`) by running the following command:

```
openssl req -nodes -new -x509 -keyout rui.key -out rui.crt -days 3650 -config openssl.cnf
```

NOTE This command assumes that the `openssl.cnf` file is in the same folder as where the certificate is generated. If the certificate is in another folder, supply the full path with the `openssl.cnf` file name.

- 3 Create backups of the original, default certificate and key to a safe location, in case you have problems and must restore your system to its previous state.

Copy the newly generated self-signed certificate (`rui.key` and `rui.crt`) to the vCenter Server location specified in [Table 2, “Default Locations for vCenter Server Certificates,”](#) on page 2.

Install Certificates on Windows Client Hosts

The vSphere Client uses the local Windows certificate store during the server-certificate verification process. After you have valid certificates on all servers, you can add the certificates and root CAs necessary to verify the server certificates.

NOTE If you obtained certificates signed by a commercial CA, you do not need to perform this task.

If you created your own root CA certificate and used it to sign server certificates, you must import the root certificate into each vSphere Client where you will enable server-certificate verification. The vCenter Server system is a client of the ESX/ESXi to which it connects, so you must import the new certificate into the vCenter Server system. The root CA (or other server certificates) must be imported into the certificate store associated with the proper Windows account for the type of vCenter Server system (server or client), as follows:

- For the vCenter Server host, you must install the root CA (or certificate) as Administrator, since the certificate must be available to the Windows service.
- For vSphere Client systems, log in to the Windows host system using the regular user credentials that you use to connect to the vCenter Server system.

Before you begin this task, use the security-conscious mechanism of your choice (for example, nonwritable media or a known-trusted server) and make the signing certificate (`rui.crt` or equivalent) available for import to the client hosts and the vCenter Server host.

NOTE The `.crt` file comprises the digital signature plus the public key only—not the private key.

To install certificates on Windows client hosts

- 1 Launch the Certificates Microsoft Management Console (MMC) snap-in.
- 2 Navigate to the %SystemRoot%\System32\ directory on the Windows system and find the certmgr.msc file.
- 3 Right-click the certmgr.msc file.

If you are importing the certificate to the vCenter Server host system:

- a Select **Run as** from the pop-up menu.
 - b Enter the Administrator credentials specific to the Windows local Administrator group in the dialog.
- 4 Click **OK** to continue.
 - 5 Install the local root CA certificate used to sign server certificates into the Windows certificate store.
 - 6 On the Certificates pane, click the Trusted Root Certification Authorities folder.
 - 7 From the Action menu, select **All Tasks followed by Import** to launch the Certificate Import Wizard.

The Certificate Import Wizard lets you navigate to the location of the certificate file and import it into the Trusted Root Certification Authorities folder.

If you created your own local root CA and used it to sign all server certificates, you need only import the local root CA certificate.

Related Publications

For more information about using certificates with vCenter Server systems, see the VMware vSphere 4.1 documentation for vCenter Server 4.1, including the *ESX Configuration Guide*, *ESXi Configuration Guide*, and the *vSphere Datacenter Administration Guide*.

If you have comments about this documentation, submit your feedback to: docfeedback@vmware.com

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

Copyright © 2009 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>. VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

Item: EN-000176-00
