

Replacing VirtualCenter Server Certificates

VMware Infrastructure 3

This technical note provides information about replacing the default certificates supplied with VirtualCenter Server hosts. For environments that require strong security, VMware recommends that you take the following actions:

- Install certificates signed by a commercial Certificate Authority (CA) on all VirtualCenter Server, ESX Server, GSX Server, and VMware Server hosts;
- Upgrade existing VirtualCenter Server (and matching client) deployments to VirtualCenter 2.0.1 Patch 1, VirtualCenter 1.4.1 Patch 1, VirtualCenter 1.3.1 Patch 2, or subsequent release; and
- Enable server-certificate verification on all VirtualCenter clients (Virtual Infrastructure Client and VirtualCenter Client) and the VirtualCenter Server host.

Many VMware customers replace default certificates with signed certificates as a matter of standard operating procedure. However, if you have not replaced the default VirtualCenter Server certificate, you must do so before you can enable server-certificate verification (because of an issue with the default certificate—see [“When to Replace Default Certificates”](#) on page 2 for details).

This technical note includes these topics:

- [“Background”](#) on page 2
- [“When to Replace Default Certificates”](#) on page 2
- [“Certificate Specifications”](#) on page 3
- [“Server-Certificate Replacement Process”](#) on page 4
- [“Using OpenSSL to Create Security Artifacts”](#) on page 5
- [“Installing Certificates on Windows Client Hosts”](#) on page 8
- [“Next Steps”](#) on page 9
- [“Related Publications”](#) on page 10

NOTE If you have replaced the default VirtualCenter Server certificates with certificates signed by a commercial CA, you can enable server-certificate verification without performing any additional tasks. See Knowledge Base [“Enabling Server-Certificate Verification for Virtual Infrastructure Clients”](#) (KB article 4646606) for details.

Background

All versions of VMware products, including all releases of VirtualCenter Server prior to the patches discussed in this technical note, use X.509 certificates to encrypt session information sent over SSL (secure sockets layer protocol) connections between server and client components. For example, communications between a VirtualCenter Server host and each ESX Server host that it manages are always encrypted. However, the authenticity of the server certificate presented during the SSL handshake phase (prior to encryption), was not verified by the client, leaving clients vulnerable to “man-in-middle” attacks.

VirtualCenter 2.0.1 Patch 1, VirtualCenter 1.4.1 Patch 1, VirtualCenter 1.3.1 Patch 2, and subsequent releases resolve this issue for Windows clients by supporting the proper client behavior during the SSL handshake.

NOTE The software updates discussed in this technical note do not resolve the server-certificate verification issue for clients running on Linux or for clients using the MKS Plugin for the Mozilla Firefox browser. The issue for these client types will be handled in future releases.

Server-certificate verification is not enabled by default: you must explicitly enable the server-certificate verification. However, before enabling server-certificate verification functionality, make sure the following requirements are met:

- Each server (to which an upgraded Windows client will connect) has a valid server certificate.
- The certificate authority (CA) that signed each server’s certificate is trusted by all Windows clients. (Commercial CAs are trusted, by default, by the Windows operating system.)

In some cases, you might have to replace the default certificate prior to enabling the server-certificate verification.

When to Replace Default Certificates

Depending on your specific deployment, you might have to replace certificates on one or more servers. In some cases, you may pre-trust the default server certificates.

NOTE If you do not plan to enable server-certificate verification, you do not have to replace any certificates, nor do you have to pre-trust any certificates.

- **ESX Server, VMware Server, and GSX Server host certificate**—Default, self-signed certificates are created automatically during the installation process. These certificates are valid and do not need to be replaced. However, if you choose to continue using these certificates, you must pre-trust them on any Windows client host that connects to the servers.
- **VirtualCenter server host certificate**—Default certificates are defective and must be replaced before you enable server-certificate verification. The certificates you obtain for your servers must meet the specifications described in [“Certificate Specifications”](#) on page 3.
- **VirtualCenter Web services engine certificate**—The certificate provided with the VirtualCenter 1.x Web services engine (used by the SDK) must be replaced. The certificate provided expired on 2006 November 13 and was for demonstration purposes only. (The SDK certificate is different than the certificate used by the VirtualCenter Server.)

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 valid certificate, you must pre-trust the server certificate by importing it into the local certificate store on the Windows client.

You must pre-trust all server 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 self-signed server certificates that you will continue to use on the various VMware server products. See [“Installing Certificates on Windows Client Hosts”](#) on page 8 for details.

Replacing the default VirtualCenter Server certificate is required if you intend to enable server-certificate verification after upgrading to VirtualCenter 2.0.1 Patch 1, VirtualCenter 1.4.1 Patch 1, or VirtualCenter 1.3.1 Patch 2. The certificates you obtain for your servers must meet the specifications described in this section.

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 PEM (Privacy Enhanced Mail, a key format that stores data in a Base-64 encoded DER (Distinguished Encoding Rules)) format.

The key used to sign the certificates must be a standard RSA key with an encryption length ranging from 768 to 2048 bits. The recommended length is 1024 bits.

NOTE Due to a current limitation, the VirtualCenter Server and the ESX Server hosts that it manages cannot all have 2048-bit key lengths, so if the ESX Server hosts use 2048-bit key lengths, the VirtualCenter Server key length cannot exceed 1024 bits.

The key and certificate names for various VMware server products are shown in [Table 1](#). The syntax examples create certificates and keys in the required format.

Table 1. Names of Key and Certificate Files

Server	Private Key	Certificate	PFX ¹
ESX Server 2.x (MUI)	mui.key	mui.crt	~
ESX Server 3.x, ESX Server 2.x, GSX Server 3.x, VMware Server 1.x	ruu.key	ruu.crt	~
VirtualCenter Server 1.x, VirtualCenter Server 2.x	ruu.key	ruu.crt	ruu.pfx
VI SDK (1.x)	server.pem	server.pem ²	ruu.pfx

1. PFX is Personal Information Exchange Format, a format that 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).
2. The certificate must be signed by a root CA named root.pem.

Certificate Locations

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

Table 2. Default Locations for Server Certificates

Server	Directory Location for Certificate
ESX Server 3.x, 2.x	/etc/vmware/ssl/
ESX Server 2.x (MUI)	/etc/vmware-mui/ssl/
GSX Server 3.x (Linux)	/etc/vmware/ssl/
GSX Server 3.x (Windows)	C:\Program Files\VMware\VMware GSX Server\ssl\
VirtualCenter 2.x, 1.x	C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\SSL\
VI SDK (1.x)	C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\VMA\
VMware Server 1.x (Linux)	/etc/vmware/ssl
VMware Server 1.x (Windows)	C:\Program Files\VMware\VMware Server\ssl
VMware Server 2.x (Linux)	/etc/vmware/ssl
VMware Server 2.x (Windows)	C:\Program Files\VMware\VMware Server\ssl and %ALLUSERSPROFILE%\Application Data\VMware\VMware Server\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.

Server-Certificate Replacement Process

For production environments, deploy new certificates in stages, rather than all at the same time. Be sure you understand the full scope of the process as it may apply to your specific environment before taking any actions.

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

Process Summary

The process of replacing VirtualCenter Server certificates is summarized in the steps below. Follow the process in the proper sequence, although some of the details might not apply to every deployment.

Prior to replacing certificates:

- 1 Download the updated software appropriate for your VMware product licenses from the VMware website:
 - VirtualCenter Server 2.0.1 Patch 1
 - VirtualCenter Server 1.4.1 Patch 1
 - VirtualCenter Server 1.3.1 Patch 2
- 2 Verify the md5sum values for the downloaded files, as detailed in the download instructions or readme file.

Generating certificate-signing request:

- 3 Obtain OpenSSL software as needed. See [“Certificate- and Key-Generation Tool”](#) on page 5 for details.
- 4 Backup any existing certificates and keys (storing them in a secure location) so that you can restore them if you have problems connecting to any servers after replacing the keys or certificates.
- 5 Create certificate-signing requests (CSR) for the VirtualCenter Server host, and, optionally, for any ESX Server host, GSX Server host, and VMware Server host certificates that you want to replace. You have the option of pre-trusting the default certificates for ESX Server, GSX Server, and VMware Server hosts on the Windows client, because these certificates are valid.
- 6 Use the fully-qualified domain name for the hostname name in the certificate-signing request and set the expiration date to a suitable time in the future, according to your security constraints. See [“Creating Certificate Signing Requests for Server Hosts”](#) on page 6 for details.
- 7 Send all CSRs to a commercial certificate authority (CA), such as Entrust or Verisign.
- 8 Alternatively, you can use your own local root CA to sign the certificate signing requests. See [“Creating a Local Root CA”](#) on page 6 for details.

Replacing the Default Certificate on the Server Host:

- 9 Copy the signed certificates to the appropriate locations on all servers, as shown in [Table 2 “Default Locations for Server Certificates,”](#) on page 3.
- 10 For VirtualCenter Server (both versions 1.x and 2.x), create the .pfx file and copy to the location specified in [“Creating the PFX File”](#) on page 8.
- 11 Connect to the VirtualCenter Server host from the appropriate client tool (VirtualCenter Client or VI Client, depending on the version of the VirtualCenter you are using), and from the client tool, perform these tasks:
 - a Power-off all VMs running on any servers (ESX Server, GSX Server, and VMware Server) being managed by the VirtualCenter Server.

- b After the VMs have stopped running, disconnect the servers from the pool of servers being managed by VirtualCenter Server.
- 12 Upgrade VirtualCenter Server instances to version appropriate for your licenses (VirtualCenter Server 2.0.1 Patch 1 (Build 33643), VirtualCenter Server 1.4.1 Patch 1 (Build 33425), VirtualCenter 1.3.1 Patch 2, or later releases).
 - 13 Upgrade all clients to the version appropriate for the VirtualCenter Server host (Virtual Infrastructure Client or VirtualCenter Client).
 - 14 On each Windows client, install (as Administrator) any default certificates (from ESX Server host, GSX Server host, or VMware Server host) systems, and install your local root CA (if you used one to sign your own certificates). See [“Installing Certificates on Windows Client Hosts”](#) on page 8.

Enabling Server-Certificate Verification and Re-connecting Servers:

- 15 Enable server-certificate verification on all clients, including the VirtualCenter Server host, by using the registry (.reg) file provided in KB 4646606.
- 16 Restart all servers to ensure that new certificates are loaded into memory.
- 17 From the VirtualCenter Server, connect to ESX Server, GSX Server, and VMserver hosts.
- 18 Reconnect all servers to the VirtualCenter Server.
- 19 Power-on all VMs that you shut down earlier in the process.

Certificate- and Key-Generation Tool

VMware products implement the OpenSSL libraries and toolkits to generate the default certificates created during installation process. You can use OpenSSL to create new keys and certificates, a root CA (if appropriate), and certificate-signing requests. You can download OpenSSL from <http://www.openssl.org>.

If you are going 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.

Using OpenSSL to Create Security Artifacts

The following sections include syntax examples for:

- [“Creating a Local Root CA”](#)
- [“Creating Certificate Signing Requests for Server Hosts”](#)
- [“Creating Self-Signed Certificates”](#)
- [“Copying the Replacement Certificate to the Server Host”](#)

The examples shown are run from a Windows host machine, so make changes as needed if you are using Linux.

The examples 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 the like. For simplicity's sake, the syntax examples shown in the following subsections assume a flat directory structure.

Configuration File for OpenSSL

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 defaults by passing values to the command line. The syntax examples in the remainder of this document assume that:

- 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/ru1.key`.
- The `[CA]` section references a `CA_default` section.
- The `[CA_default]` section references a `private_key` named `myroot.key`.

Create or modify your own `openssl.cnf` for the specifics of your environment. The OpenSSL commands shown in the following sections of this technical note are for example purposes only.

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

Creating a Local Root CA

To replace the default certificates with certificates signed by your own local CA, you must start by creating a root CA. The root CA's certificate must then be installed in any client host systems that you will use to connect to the servers. 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, prior to enabling the server-certificate verification.

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
```

If you are using the VirtualCenter 1.x SDK, you must create a duplicate of the root CA by copying it to the necessary filename, as follows:

```
C:\OpenSSL\bin>copy myroot.crt root.pem
```

The steps in the following section show how to use this `root.pem` to create the additional artifacts needed for VirtualCenter 1.x SDK.

Creating Certificate Signing Requests for Server Hosts

Whether you create your own signed certificate or purchase a certificate from a commercial certificate authority, you must create a certificate signing request for each server that requires a replacement certificate.

- 1 Navigate to the OpenSSL directory.
- 2 Edit the OpenSSL configuration file (`openssl.cnf`), specifying the details appropriate for your environment.
- 3 Generate the RSA key for the server and the certificate signing request (CSR):

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

When prompted, specify the fully qualified hostname as the server's `commonName`. At the end of this step, you should have a private key (`ru1.key`) and a companion certificate-signing request (`mycsr.csr`).

For VirtualCenter 1.x SDK copy the server key to the proper filename for the SDK, as follows:

```
C:\openssl\bin>copy ru1.key server.pem
```

- 4 Send the certificate request to the commercial certificate authority of your choice and await the return of the signed certificate for the server.

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

```
openssl ca -out ru1.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) `ru1.crt` for the specified server, and the private key for the server (`ru1.key`).

Creating Self-Signed Certificates

This section describes creating a self-signed certificate.

- 1 Create a text file `openssl.cnf` with the configuration settings for `openssl`.
- 2 The content of this file is as follows:

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

```
[ req ]
default_bits           = 1024
default_keyfile        = ru1.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>
```

- 3 Run the following command to create the self-signed certificate (`ru1.key` and `ru1.crt`):

```
openssl req -nodes -new -x509 -keyout ru1.key -out ru1.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.

- 4 Create backups of the original certificate files.
- 5 Copy the newly generate self-signed certificate (`ru1.key` and `ru1.crt`) to the location specified in [Table 2 "Default Locations for Server Certificates."](#)

Additional Steps if Certificate Checking is Enabled in VirtualCenter:

The certificate checking option is a non-default setting in VirtualCenter. Certificate checking is explained in the [ESX Server 3 Configuration Guide](#).

NOTE Using a self-signed certificate on ESX Server 2.5.5 is not possible with the certificate checking option enabled in VirtualCenter. Follow the ["Server-Certificate Replacement Process"](#) if the certificate checking option is enabled in VirtualCenter and you want to replace your certificate on ESX Server 2.5.5.

The following steps are necessary if certificate checking is enabled in VirtualCenter:

- 1 Reboot the ESX host or VMware Server after changing certificates.
- 2 Reset the password of the VirtualCenter Database. This password is randomly generated. Enter the following command in the command line interface on the system where VirtualCenter is installed:

```
cd C:\Program Files\VMware\Infrastructure\VirtualCenter Server\
vpxd -p
```

- 3 Exit the VI Client.
- 4 Restart the VirtualCenter service.
- 5 Restart the VI Client.

Copying the Replacement Certificate to the Server Host

Before replacing a certificate or key on any server, copy the default certificate and key to a safe location, in case you have problems and must restore your server to its previous state.

For ESX Server host, GSX Server host, and VMware Server host systems, copy the signed certificate and key (`ru1.crt`, `ru1.key`) to the appropriate location on the server. See [Table 2 “Default Locations for Server Certificates,”](#) on page 3, for details.

For VirtualCenter Server host, you must also copy the signed certificate and key (`ru1.crt`, `ru1.key`) to the appropriate location:

C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\SSL

In addition, you must create a PFX-formatted certificate file specific for Windows.

Creating the PFX File

The `ru1.pfx` file is a concatenation of the server’s certificate and private key, exported in the PFX format. This file is then copied to the sub-directory on the VirtualCenter server host system, as detailed in the next steps:

- 1 Export the certificate and keyfile together to the PFX format, using OpenSSL as follows:

```
openssl pkcs12 -export -in ru1.crt -inkey ru1.key -name ru1 -passout pass:testpassword -out ru1.pfx
```

- 2 Copy the `ru1.pfx` to this folder on the VirtualCenter server host (Windows) system:

C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\SSL

- 3 If applicable, for VirtualCenter 1.x SDK:

- a Copy the `root.pem` and the `server.pem` to the server, to this directory:

C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\VMA

- b Using OpenSSL, generate the PFX for the VirtualCenter 1.x SDK by concatenating the certificate and the private key, as in:

```
openssl pkcs12 -export -in server.pem -inkey server.key -name ru1 -passout pass:testpassword -out ru1.pfx
```

- c Navigate to the VirtualCenter server installation directory:

```
cd "C:\Program Files\VMware\VMware VirtualCenter"
```

- d Use VirtualCenter’s command-line utility (`vma.exe`) to update the `vmaconfig.xml` file and Windows Registry with the full pathname to the new certificate and key, as shown in these four commands:

```
vma -config "C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\VMA\vmaconfig.xml" -update -sslCert "C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\VMA\server.pem"
vma -config "C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\VMA\vmaconfig.xml" -update -sslCAChain "C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\VMA\root.pem"
vma -config "C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\VMA\vmaconfig.xml" -update -sslPrivateKey "C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\VMA\server.pem"
vma -config "C:\Documents and Settings\All Users\Application Data\VMware\VMware VirtualCenter\VMA\vmaconfig.xml" -update -sslPassphrase <passphrase>
```

Re-start the server hosts and follow other steps listed in the process summary, [“Process Summary”](#) on page 4.

Installing Certificates on Windows Client Hosts

The VI Client and VC Client use 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 have 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 Windows client on which you want to enable server-certificate verification. The VirtualCenter Server host system is a client of the ESX Servers to which it connects, so you must import the new certificate into the server host system, as well as all client host systems. The root CA (or other server certificates) must be imported into the certificate store associated with the proper Windows account for the type of VirtualCenter system (server or client), as follows:

- For the VirtualCenter Server host, the root CA (or certificate) must be installed as Administrator, since the certificate must be available to the Windows service.
 - For other VI Client or VC Client host systems, logon to the Windows host system using the regular user credentials that you use to connect to the VirtualCenter server or ESX Server (or other server) systems.
- 1 Using the security-conscious mechanism of your choice (nonwritable media or a known-trusted server, for example), make the signing certificate (ru1.crt or equivalent) available for import to the client hosts and the VirtualCenter Server host.

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

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

If you are importing the certificate into the VirtualCenter Server host system:

- Select Run as... from the popup menu.
- Enter the Administrator credentials specific to the Windows local Administrator group in the dialog.

- 6 Click OK to continue. The Certificates pane displays.
- 6 Install the local root CA certificate used to sign server certificates into the Windows certificate store.
- 7 Click the Trusted Root Certification Authorities folder in the Certificate pane.
- 8 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. If your ESX Server, GSX Server, or VMware Server hosts continue using the default self-signed certificates, you must also import those certificates into the Trusted Root Certification Authorities folder.

Next Steps

You can now enable server-certificate verification on the Virtual Infrastructure Windows clients. See Knowledge Base article 4646606, “Enabling Server-certificate Verification for Virtual Infrastructure Clients”) for details. The KB includes a registry (.reg) file that must be run on each client host system.

You must also enable the server-certificate verification on the VirtualCenter Server host system. Because the VirtualCenter Server host is a client to all ESX Server hosts, GSX Server hosts, and VMware Server hosts that it manages, run the registry file on the VirtualCenter Server host system as well.

Related Publications

Knowledge Base Article	Location
Enabling Server-Certificate Verification for Virtual Infrastructure Clients	http://kb.vmware.com/kb/4646606
Replacing or Regenerating an SSL Certificate for the Management Interface	http://kb.vmware.com/kb/1843
Configuration Program vmware-config Might Set Incorrect Permissions on SSL Key Files	http://kb.vmware.com/kb/2467205
Intermittent SSL Warnings Appear During Logoff	http://kb.vmware.com/kb/2169
Resetting the HTTP Session Timeout and Regenerating the SSL Certificate After Upgrading ESX Server	http://kb.vmware.com/kb/1517

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, the VMware "boxes" logo and design, Virtual SMP, and VMotion are registered trademarks or trademarks 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
