RSA Validation Manager 3.1
Installation Guide

The Security Division of EMC

Contact Information

Go to the RSA corporate web site for regional Customer Support telephone and fax numbers**:** **www.rsa.com**

Trademarks

RSA and the RSA logo are registered trademarks of RSA Security Inc. in the United States and/or other countries. For the most up-to-date listing of RSA trademarks, go to **www.rsasecurity.com/legal/trademarks_list.pdf**. EMC is a registered trademark of EMC Corporation. All other goods and/or services mentioned are trademarks of their respective companies.

License agreement

This software and the associated documentation are proprietary and confidential to RSA, are furnished under license, and may be used and copied only in accordance with the terms of such license and with the inclusion of the copyright notice below. This software and the documentation, and any copies thereof, may not be provided or otherwise made available to any other person.

No title to or ownership of the software or documentation or any intellectual property rights thereto is hereby transferred. Any unauthorized use or reproduction of this software and the documentation may be subject to civil and/or criminal liability.

This software is subject to change without notice and should not be construed as a commitment by RSA.

Third-party licenses

This product may include software developed by parties other than RSA. The text of the license agreements applicable to third-party software in this product may be viewed in the **thirdpartylicenses.pdf** file.

This product includes software developed by The Apache Software Foundation (**www.apache.org/**).

Note on encryption technologies

This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when using, importing or exporting this product.

Distribution

Limit distribution of this document to trusted personnel.

RSA notice

Portions of this product use technologies patented under U.S. patent numbers 5,922,074 and 6,249,873.

First printing: October 2007

Contents

Preface

About This Guide

This guide describes how to install RSA Validation Manager. It is intended for administrators and other trusted personnel. Do not make this guide available to the general user population.

RSA Validation Manager Documentation

For more information about Validation Manager, see the following documentation:

Readme. Provides information about what is new and changed in this release, as well as workarounds for known issues. The latest version of the *Readme* is available from RSA SecurCare Online at **https://knowledge.rsasecurity.com**.

Getting Started. Lists what the kit includes (all CDs, diskettes, licenses, and documentation), specifies the location of the documentation on the CD, and lists RSA Customer Support web sites.

Installation Guide. Describes detailed procedures on how to install Validation Manager.

Administrator's Guide. Provides information for your administrators about how to configure and administer Validation Manager.

Command Line Reference Guide. Provides information on using the command line utility available in Validation Manager.

RSA Validation Manager Help. Describes day-to-day administration tasks performed in the administration user interface. To view Help, click the **Help** tab on the administration user interface.

Related Documentation

For more information about products related to RSA Validation Manager, see the following:

RSA Certificate Manager documentation set. The full documentation set for RSA Certificate Manager is included in the **\Documentation** directory of the RSA Certificate Manager CD.

RSA Secured Partner Solutions directory. RSA has worked with a number of manufacturers to qualify products that work with RSA products. Qualified third-party products include virtual private network (VPN) and remote access servers (RAS), routers, web servers, and many more. To access the directory, including implementation guides and other information, go to **www.rsasecured.com**.

Getting Support and Service

RSA SecurCare Online	https://knowledge.rsasecurity.com
Customer Support Information	www.rsa.com/support
RSA Secured Partner Solutions Directory	www.rsasecured.com

RSA SecurCare Online offers a knowledgebase that contains answers to common questions and solutions to known problems. It also offers information on new releases, important technical news, and software downloads.

The RSA Secured Partner Solutions Directory provides information about third-party hardware and software products that have been certified to work with RSA products. The directory includes Implementation Guides with step-by-step instructions and other information about interoperation of RSA products with these third-party products.

Before You Call Customer Support

Make sure you have direct access to the computer running the RSA Validation Manager software.

Please have the following information available when you call:

- ❑ Your RSA Customer/License ID. You can find this number on the license certificate that shipped with the product or on the label of the license CD, as applicable.
- ❑ RSA Validation Manager software version number.
- ❑ The make and model of the machine on which the problem occurs.
- ❑ The name and version of the operating system under which the problem occurs.

1 Planning Your Deployment

Before you install RSA Validation Manager and configure the system, plan how you want to deploy Validation Manager in your organization. This chapter describes possible deployment scenarios you must consider, depending on your needs and planned system capacity. The topics include:

Planning Your Physical Deployment

Planning physical deployment is determining which software components to install and where to install them. Consider the following before you install Validation Manager:

- Do you intend to use smart cards and a hardware security module (HSM)?
- Do you want to deploy a clustering environment for improved performance, load balancing, and failover?
- What is your external communications bandwidth?
- What are your performance criteria?
- What hardware resources do you need?
- Where will you install Validation Manager?

Using a Hardware Security Module

To generate and protect private keys and provide secure hardware key management for Validation Manager, consider using an HSM for system and Online Certificate Status Protocol (OCSP) signer keys. For more information, see Appendix C, "Cryptographic Hardware Interoperability."

Clustering

In a cluster, multiple copies of Validation Manager share the work of processing OCSP certificate status requests. To its clients, the cluster appears as a single installation, but is really a group of servers acting as one. For more information, see Chapter 6, "Clustering with RSA Validation Manager."

Communications Bandwidth

In bandwidth-constrained environments where downloading large revocation lists is not practical, consider synchronizing your installation to another installation by creating synchronized revocation list-based status sources. For more information, see the *Administrator's Guide*.

Performance

Performance depends on load and processor speed. To improve performance, select less strict OCSP validation modes and levels, log only failure events to the audit log, cache OCSP responses, do not validate signatures in OCSP requests, and use an HSM for key generation.

To process a high volume of OCSP requests, balance the work load among multiple instances of Validation Manager to improve performance.

Hardware Resources

You need one machine of your choice for a typical installation of Validation Manager. To set up a cluster of Validation Manager installations, you need a separate machine for each node. You also need network connections to the external machines or access to a mail server to import data such as CA certificates and revocation lists. For more information, see Chapter 2, "System Requirements."

Where to Install

Determine where in your organization's network you want to install Validation Manager. You may want to install it behind a firewall or require secure connections.

Planning Your Logical Deployment

Logical deployment is how you use the software in your business environment. After you install Validation Manager, consider the following:

- How many certificate authorities (CAs) will you import?
- How many OCSP signers will you create?
- How many status sources will you create? Where are they located? What types are needed? Is communication to the status sources secure or non-secure?
- What events will you log? What type of logging do you require?
- Will you re-sign the system certificates?
- How many Administrators do you need to manage this installation? What method do you use to authenticate Administrators?
- What is the format of the OCSP requests and responses?

Certificate Authorities

For Validation Manager to send valid OCSP responses for certificates, you must import the CA certificates of the CAs that issued those certificates. Import the CA certificates by pasting in the certificate text or browsing to a file on the network. You can import revocation lists in the same manner.

OCSP Signers

You must create OCSP signers to send signed OCSP responses. OCSP signer keys can be software-based or hardware-based. You may want to use CA-issued OCSP signer certificates, issued by the same CAs that issued the certificates whose status has been requested.

Status Sources

Determine how many status sources to create and what types you need. You must know the hostname and port number of each status source, and they must be located on the network. Communication to each status source may be secure or non-secure.

Audit Logging

By default, Validation Manager logs all events if the operation fails. Validation Manager also logs certain events after successful completion. If required by your organization's security practices, you can configure Validation Manager to make the successful logging of an event a determining factor in whether the event itself is successful. If the event is not logged, Validation Manager cancels the entire operation.

System Certificates

The system certificates are created during installation and issued by the System CA. You may want them re-signed by an external CA or you may want to replace the System CA by an external CA.

Administrators

When you install Validation Manager, one Administrator is created. You may want to create more Administrators to share the workload or your organization may require you to add more Administrators for security practices. You need to choose a system-wide authentication method for Administrators. You can use your organization's LDAP directory for storage of Administrator information or the Validation Manager database.

OCSP Requests and Responses

You can configure Validation Manager to perform different levels of validation on all OCSP requests and to reuse OCSP responses on a system-wide or per-CA basis.

For more information on the topics in this section, see the *Administrator's Guide.*

2 System Requirements

This chapter describes the supported platforms and system requirements for an RSA Validation Manager installation.

Supported Platforms

Validation Manager is supported on the following platforms:

- Microsoft Windows Server 2003 operating system
- Sun Solaris 9 or 10 operating system
- Red Hat Enterprise Linux 5 operating system

System Configuration

The minimum system configuration is:

- On a Windows platform:
  - Pentium III 650 MHz.
  - 512 MB of memory (RAM).
  - Hard disk with 250 MB of free space for basic program installation. In addition, you might need hard disk space to accommodate a large log file size. If you plan to log OCSP transactions, your system needs approximately 1.5 MB for each 1000 OCSP transactions.
- On a Solaris platform:
  - Sun Enterprise Ultra 10S or equivalent.
  - 512 MB of memory (RAM).
  - Hard disk with 250 MB of free space for basic program installation. In addition, you might need hard disk space to accommodate a large log file size. If you plan to log OCSP transactions, your system needs approximately 1.5 MB for each 1000 OCSP transactions.
- On a Linux platform:
  - Intel Pentium 4 or equivalent.
  - 512 MB of memory (RAM).
  - Hard disk with 250 MB of free space for basic program installation. In addition, you might need hard disk space to accommodate a large log file size. If you plan to log OCSP transactions, your system needs approximately 1.5 MB for each 1000 OCSP transactions.

Browser Support for Administration

Validation Manager supports web-based administration with the following browsers.

	Windows XP	Windows 2000	Windows 2003	Windows Vista	Red Hat Linux	Solaris 9 or 10
Microsoft Internet Explorer 6.0	√	√	√			
Microsoft Internet Explorer 7.0	√		√	√		
Mozilla Firefox 2.0	√				√	
Mozilla 1.7 browser						√

Browser Settings

Enabling Javascript and UTF-8 Encoding

You must enable JavaScript and UTF-8 encoding for web-based administration. If you do not enable both, some pages of Validation Manager may not be displayed properly.

In Internet Explorer:

1. To enable JavaScript:
   a. Click **Tools > Internet Options**, and click the **Security** tab.
   b. Select the appropriate web content zone.
      If you are not sure which zone to select, you can enable JavaScript for all of the following: Internet, Local intranet, and Trusted sites.
   c. Select the Security level for the zone.
      If you are using the default security level, JavaScript is enabled. (JavaScript is enabled at the Low, Medium-low, and Medium levels.)
      If you are using a custom security level, click **Custom Level**. Scroll down to **Scripting** > **Active Scripting**, and select **Enable**.
2. To enable UTF-8 encoding, do one of the following:
   - In Microsoft Internet Explorer 6.0, click **View > Encoding > Unicode (UTF-8)**, and clear **Auto-Select**.
   - In Microsoft Internet Explorer 7.0, click **Page > Encoding > Unicode (UTF-8)**, and clear **Auto-Select**.

In Firefox:

1. To enable JavaScript, do one of the following:
   - On Windows, click **Tools > Options**, and in the **Content** tab select **Enable JavaScript**.
   - On Linux, click **Edit > Preferences**, and in the **Content** tab select **Enable JavaScript**.
2. To enable UTF-8 encoding, click **View > Character Encoding** > **Unicode (UTF-8)**.

In Mozilla:

1. To enable JavaScript:
   a. Click **Edit > Preferences > Advanced > Scripts & Plug-ins**.
   b. Under **Enable JavaScript for**, select **Navigator**.
   c. Under **Allow scripts to**, select all options except **Hide the status bar**.
2. To enable UTF-8 encoding, click **View > Character Encoding > Unicode (UTF-8)**.

Entering and Displaying International Characters

On Windows

To enter and display international characters in certificates, use Input Method Editors (IMEs).

To enable IMEs:

- On Windows 2000, in the **Control Panel**, enable appropriate IMEs through **Regional Options**.
- On Windows 2003, Windows XP, and Windows Vista, in the **Control Panel**, enable appropriate IMEs through **Regional and Language Options**.

On Solaris

To enter and display international characters in certificates, edit or add the appropriate LANG, LC_CTYPE, and LC_COLLATE variables in the **/etc/default/init** file.

On Linux

To enter and display international characters in certificates, edit or add the appropriate LANG, SUPPORTED, and SYSFONT variables in the **/etc/sysconfig/i18n** file.

Cryptographic Support

To enable strong authentication, use Secure Sockets Layer (SSL) or Transport Layer Security (TLS). You can use SSL v3 or TLS v1 for both web authentication and LDAP authentication. Validation Manager components and other PKI-related applications communicate with each other through mutually authenticated SSL/TLS sessions.

The Validation Manager architecture supports key lengths up to 4096. Validation Manager natively supports the following public key technologies:

- CA Keys: RSA
- Non-CA Keys: RSA
- Message Digests: SHA-1

RSA Public Key Technology

Validation Manager can use software-based RSA key pairs for signing, signature verification, establishing SSL/TLS communication, and for other cryptographic operations.

Validation Manager can also use hardware-based RSA private keys. An HSM provides a security solution that off-loads sensitive cryptographic processing and private key storage from the host computer or Validation Manager server. Administrators can create keys and sign OCSP responses on an HSM without the encryption algorithm or private keys ever being directly accessed by the Validation Manager host server.

Hardware Security Module Support

Validation Manager supports the use of an nCipher HSM for protecting the private keys of the OCSP Responder. Validation Manager also supports the use of an nCipher HSM or a generic PKCS #11 HSM for protecting the private keys of distinct OCSP signers. For more information on HSMs, see Appendix C, "Cryptographic Hardware Interoperability."

Interoperability Requirements

Validation Manager interoperates with the following products and applications:

OCSP Clients. Validation Manager interoperates with the following OCSP client software and applications:

- RSA Validation Client 2.0
- Tumbleweed Valicert Desktop Validator 4.2
- Computer Associates *e*Trust OCSPro client 3.6
- RSA BSAFE Cert-C 2.8 client
- Microsoft Vista OCSP client
- Mozilla Firefox OCSP client

Certificate Authorities. Validation Manager can process revocation lists from the following certificate authority software and applications:

- Verisign CA
- RSA Keon Certificate Authority 6.5.1
- RSA Certificate Manager 6.6 and later
- Microsoft Windows 2000 Microsoft CA (SP4)

OCSP Servers. Validation Manager can request certificate status from the following OCSP server software and applications:

- Tumbleweed Valicert Validation Authority 4.2
- RSA Keon Certificate Authority 6.5.1
- RSA Certificate Manager 6.6 and later
- RSA Validation Manager 3.0 and later

LDAP Directories. Validation Manager can fetch revocation lists from the following LDAP directories:

- iPlanet Directory Server 5.1
- Sun ONE Directory Server 5.2

Validation Manager is capable of fetching locally published revocation lists from RSA Keon Certificate Authority 6.5.1, and RSA Certificate Manager 6.6 and later LDAP servers.

3 Installation Overview

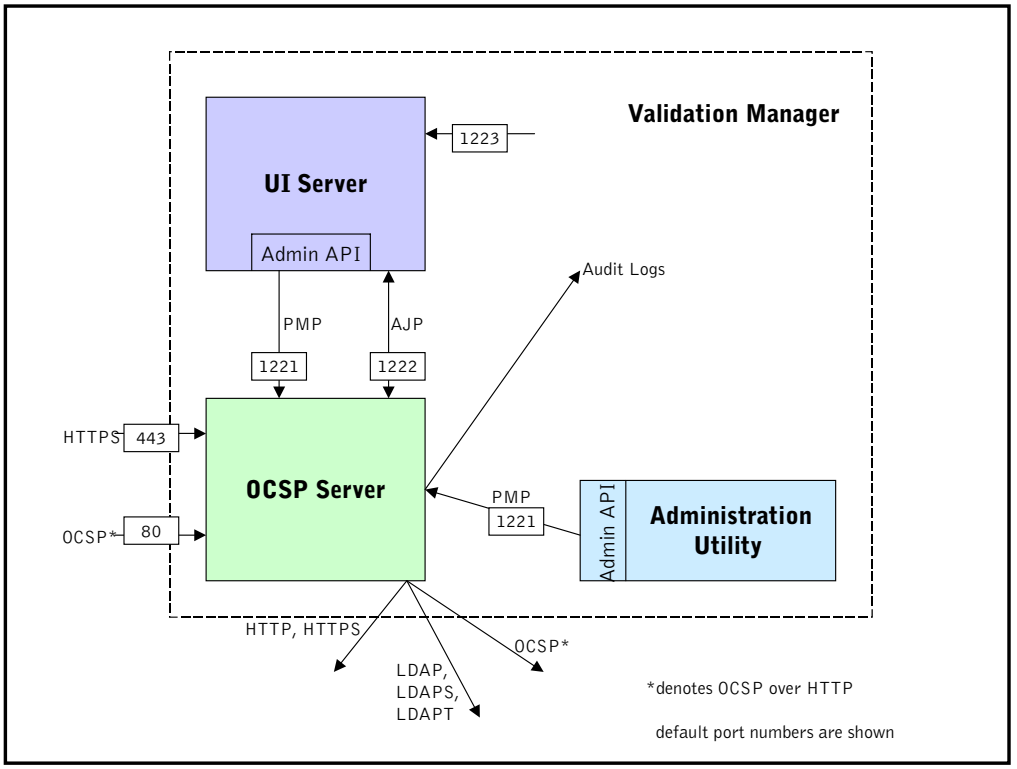
This chapter provides an overview of the RSA Validation Manager architecture, the protocols used, the CAs and certificates created during installation, and the file directory structure.

System Architecture

Typical Installation

The typical Validation Manager installation process installs two servers: the UI Server and the OCSP Server. An administration utility (vmadmin) is also installed (for more information, see the *Command Line Reference Manual*).

The following figure illustrates the Validation Manager architecture and the protocols that Validation Manager uses to communicate internally and to communicate externally with OCSP clients.



(For acronym definitions, see "Protocols Used Within Validation Manager" on page 20.)

The UI Server is the server through which you perform most administrative functions. It requires client authentication.

The administrative utility is a command line utility through which you can also perform administrative functions. It requires client authentication.

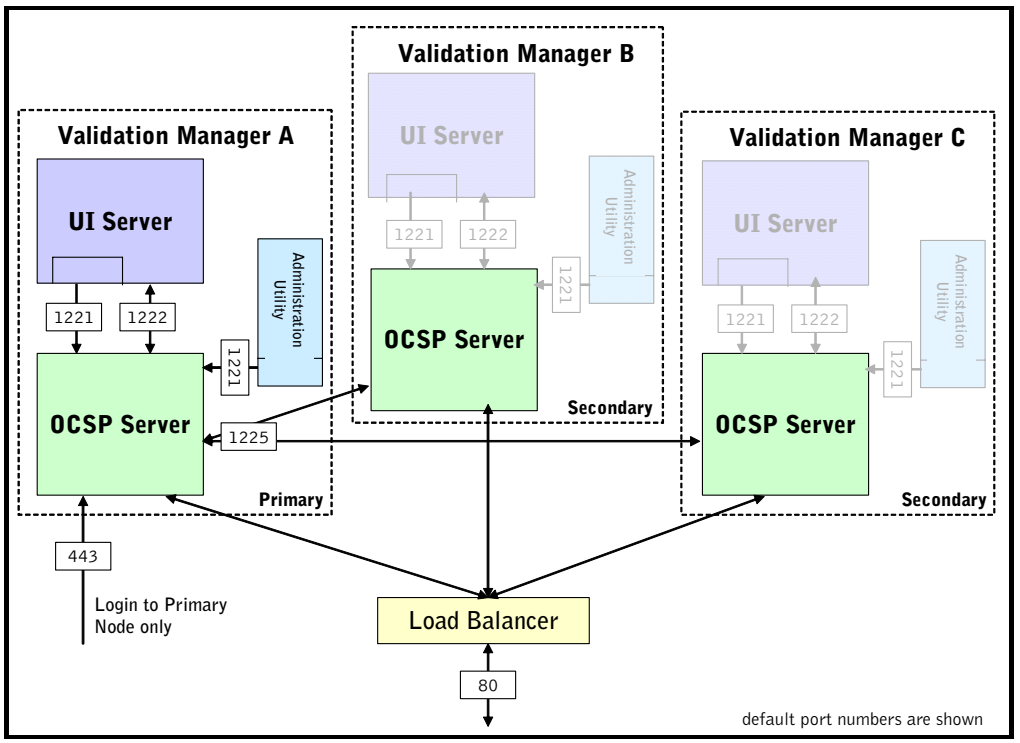
The UI Server and the administrative utility both use the administrative application programming interface (API) to manage the OCSP Server.

The OCSP Server (also known as the Validation Server) is a server that Validation Manager uses to retrieve certificate status information. It accepts HTTP-based or HTTPS-based OCSP requests and responds with the certificate's current status.

For more information, see Chapter 4, "Installing RSA Validation Manager."

Clustering

If you want to create an installation in which multiple instances of the Validation Manager OCSP Server share the processing load while appearing to users to be one server, you can install multiple Validation Managers along with a load balancer to create a cluster. The following figure illustrates a cluster of three Validation Managers.



Clustering provides these benefits:

Increased performance. A load balancer distributes OCSP request connections among the servers so that more transactions can be serviced at the same time.

High availability. If one server becomes unavailable, another picks up the OCSP request connection as seamlessly as possible.

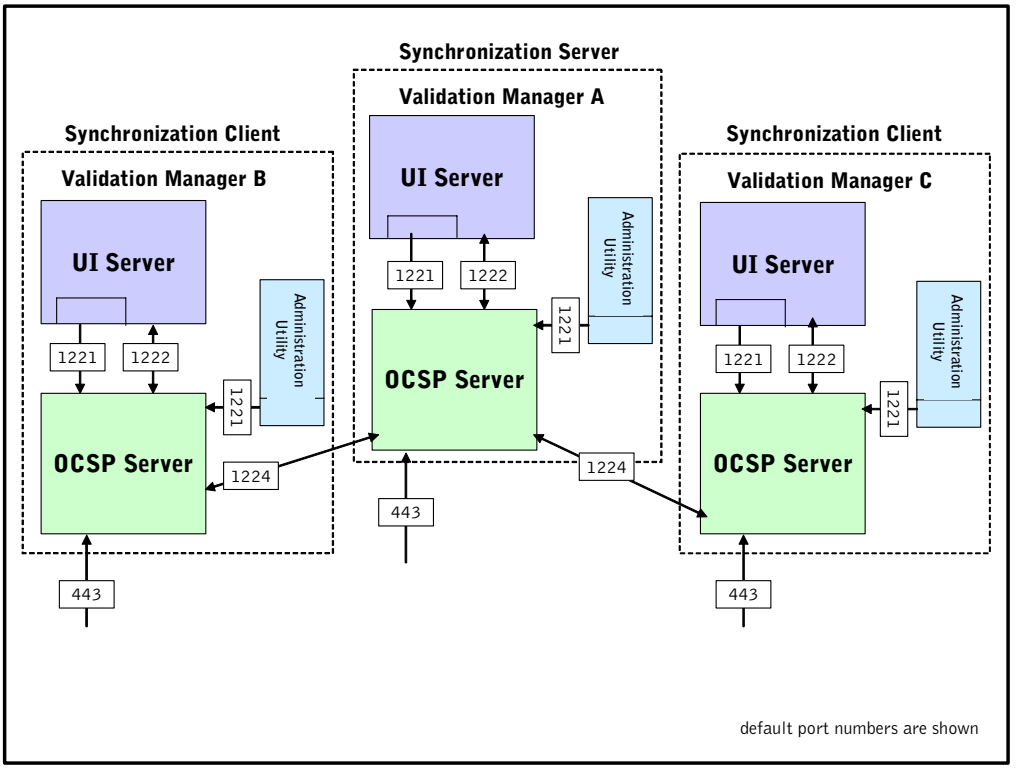
Scalability. The number of servers in your cluster can be based on the needs of your organization. As demand increases, you can add servers to the cluster.

You must install each node of the cluster on a separate machine and all machines must have the same platform type: all Windows, all Solaris, or all Linux.

For more information, see Chapter 6, "Clustering with RSA Validation Manager."

Synchronization

A typical Validation Manager installation using revocation lists may be problematic in a low bandwidth environment due to the amount of data involved in revocation list downloads. If you want to maintain up-to-date revocation list-based status information but intend to install Validation Manager with a low bandwidth connection to the network, consider configuring communications with another Validation Manager that has a higher bandwidth connection to the network. The following figure illustrates the synchronization of three Validation Managers.



The Validation Manager installation that downloads complete certificate revocation lists (CRLs) is known as the synchronization server. The other Validation Manager installations, known as synchronization clients, receive only as much data as needed to update their local stores of status data. You configure the synchronization feature on a per-CA basis.

For more information, see the *Administrator's Guide*.

Protocols Used Within Validation Manager

Validation Manager uses the following transport-layer protocols to communicate with the internal and external servers.

Protocol	Usage
HTTP	OCSP clients access the OCSP Server using basic web-access HTTP. No client or server authentication is needed.
HTTPS	Administrators access the UI Server using SSL-secured HTTP. Certificate-based client authentication is not required.
LDAP	The OCSP Server uses the Lightweight Directory Access Protocol (LDAP) to retrieve data (namely revocation lists) from external servers. Only public data is available through the LDAP protocol.
LDAPS	The OCSP Server uses SSL or TLS-secured LDAP v2 to retrieve data from secure external servers.
StartTLS	The OCSP Server uses TLS-secured LDAP v3 to retrieve data from secure external servers.
PMP	The OCSP Server receives administrative commands through the Pipeline Message Protocol (PMP). Client authenticated TLS is required.
AJP	The UI Server communicates with the OCSP Server through Apache Jserv Protocol (AJP), the Apache JK2 connector. It provides a communication link between Tomcat and Apache.

Installed CAs

When you install Validation Manager, a System CA is created. The System CA is a self-signed CA that signs all server certificates created during the installation. Do not use this CA after installation, except to re-sign the certificates it issued during installation. For more information on the purpose of these certificates, see the following section, “Issued Certificates.”

Issued Certificates

The System CA issues the following certificates during the installation process.

UI Server Certificate	This certificate allows the Validation Manager UI Server to connect to the Validation Manager OCSP Server (for PMP over TLS). Validation Manager stores this certificate as **guiadmin.p12** in **/*installed-dir*/GUIServer/webapps/vmadmin/config/**.
Command Line Utility Certificate	This certificate allows the Validation Manager command line utility to connect to the Validation Manager OCSP Server. Validation Manager stores this certificate as **admin.p12** in **/*installed-dir*/Util/**.
OCSP Responder SSL/TLS Server Certificate	This certificate is used by the OCSP Server when configured to serve OCSP over HTTPS. Validation Manager stores this certificate as **OcspServer.cert** in **/*installed-dir*/ValidationServer/tls/certs/**.
Administrator Certificate for UI Server	This certificate allows the person who installed Validation Manager to connect to the administration GUI and use all Validation Manager functions. Validation Manager stores this certificate as **UIServer.cert** in **/*installed-dir*/ValidationServer/tls/cert/**.
Administrator Certificate for Command Line Utility	This certificate allows the person who installed Validation Manager to connect to the command line utility. Validation Manager stores this certificate as **VSAdmin.cert** in **/*installed-dir*/ValidationServer/tls/cert/**.
Synchronization Server Certificate	This certificate allows this installation, when acting as a synchronization server, to connect to all synchronization clients. Validation Manager stores this certificate as **SyncServer.cert** in **/*installed-dir*/ValidationServer/tls/cert/**.
Synchronization Client Certificate	This certificate allows this installation, when acting as a synchronization client, to connect to the synchronization server. Validation Manager stores this certificate as **Client.cert** in **/*installed-dir*/ValidationServer/tls/cert/**.
Audit Log Signing Certificate	This certificate is used by Validation Manager to sign the audit logs so they are secure. Validation Manager stores this certificate as **AuditLogSigning.cert** in **/*installed-dir*/Audit/**.

Validation Manager File Directory Structure

This section identifies important directories in a Validation Manager installation. This section does not provide a complete listing of the directories and files in a standard Validation Manager installation.

- **Audit**

  Validation Manager audit log root directory

- **GUIServer**

  UI Server root directory

  - **bin**

    UI Server executables and scripts

  - **common**

    Java classes available to web applications

  - **conf**

    UI Server configuration directory

  - **logs**

    UI Server error/message logs directory

  - **server**

    internal Java classes

  - **shared**

    shared Java classes

  - **temp**

    temporary directory used by Java Virtual Machine (JVM)

  - **webapps**

    web applications (including Tomcat)

  - **work**

    scratch directory used by Tomcat

- **jre**

  Java runtime for use by installer and command line utility

- **SystemCA**

  System CA root directory

- **Uninstall**

  Uninstall root directory

  - **resource**

    temporary folder

- **Util**

  Command line utility root directory

  - **help**

    Command line help text

- **ValidationServer**

  OCSP Server root directory

  - **bin**

    OCSP Server executables and scripts

  - **conf**

    OCSP Server configuration directory

  - **db**

    OCSP Server database directory (initially empty)

  - **lib**

    Apache dynamic library directory

  - **logs**

    Apache log files directory

  - **module**

    OCSP Server module directory

  - **tls**

    OCSP Server TLS certificate and key directory

4 Installing RSA Validation Manager

This chapter describes the standard RSA Validation Manager installation process. The topics include:

- Tasks you must do before you install
- Pre-installation checklist
- Installation procedure
- Post-installation checklist

After the installation is complete, you can reset or reconfigure any defaults used by the installation.

To upgrade a previous version of Validation Manager, see Chapter 5, "Upgrading RSA Validation Manager."

To install a cluster of Validation Manager installations, see Chapter 6, "Clustering with RSA Validation Manager."

Before You Begin

Before you install Validation Manager, complete the following administrative tasks:

- Read the *Readme*, available from the Validation Manager CD, for last-minute information or known issues in the installation process.
- Complete the "Pre-Installation Checklist" on page 26.
- If you intend to use smart cards and a hardware security module (HSM), install your HSM hardware and drivers before installing Validation Manager (see Appendix C, "Cryptographic Hardware Interoperability").

  Important: Initialize the token before beginning the Validation Manager installation.

  If you intend to use a PKCS #11 hardware device, install the device and drivers before or after installing Validation Manager (see Appendix C, "Cryptographic Hardware Interoperability").
- If you intend to verify the CD you received from RSA, see Appendix D, "Media Verification."

Pre-Installation Checklist

Before you install Validation Manager, make sure you have the following information.

Installation Directory

Location of Validation Manager installation. Must not start with white space. Default on Windows is **C:\Program Files\RSA Security\RSA Validation Manager**. Default on Solaris or Linux is ***user home directory*/ RSA_Security/RSA_Validation_Manager**.

_____

Server

Fully qualified domain name (FQDN). Used as the Common Name (CN) and in the Subject Alternative Name extension in system certificates, and in Apache configuration. Default is ***server FQDN***.

_____

OCSP Server port. Used by OCSP clients to access the OCSP Server. Default is **80**, or **8080** for non-privileged users on Solaris or Linux. _____

Administrative utility port. Used by the administrative utility to access the OCSP Server. Default is **1221**. _____

UI connector port. Connects the OCSP and UI Servers. Must be lower than 32768. Default is **1222**. _____

UI control port. Controls (shuts down) the UI Server. Default is **1223**. _____

UI Server port. Used by Administrators to access the UI Server. Default is **443**, or **8443** for non-privileged users on Solaris or Linux. _____

On Windows only:

Windows service name. Cannot contain white space. Default is **RSAValidationManager3.1**.

_____

On Solaris and Linux only:

CAUTION: Install Validation Manager as the "root" user or a member of the nfast group to allow operation with nCipher cryptographic hardware. If you install as "root", you must specify a user other than "root" to run the UI Server and OCSP Server. This user must also belong to the nfast group.

UNIX user ID. Default is ***current user***. _____

UNIX group ID. Default is ***current user's group***. _____

Cryptographic Provider

System keys: ❑ software-based (default) or ❑ nCipher

System CA

Organization name. Used as the Organization (O) in system certificates.

Organizational unit name. Used as the Organizational Unit (OU) in system certificates.

System passphrase. You must install Validation Manager with a system passphrase. It can be software-based or hardware-based. For a software-based passphrase, choose a strong passphrase at least eight characters long (no white space) with at least one alphabetical and one numerical character. For a hardware-based passphrase, follow the recommendations of the vendor.

Administrator Authentication

Default Administrator name. Default is **Administrator**.

Administrator password. You must provide a password to authenticate the default Administrator when logging into the Validation Manager GUI through a web browser. Choose a strong password at least eight characters long (no white space) with at least one alphabetical and one numerical character.

Installation Procedure

The Windows-based installation package of Validation Manager consists of both a zipped and an uncompressed distribution of setup files. The Solaris-based and Linux-based installation packages of Validation Manager consist of a compressed tar distribution of setup files.

Note: A Java Virtual Machine (JVM) is included in the installer executable to allow it to run on machines that do not already have a JVM installed.

The three supported methods of installing Validation Manager are:

GUI-Based. Graphical User Interface (GUI) method on Windows, Solaris, or Linux platforms (default).

Silent. A method that is not dependent on any interaction with the user during installation on Windows, Solaris, or Linux platforms.

Console. A method through the Solaris or Linux command line only.

GUI-Based Installation

The GUI-based installation of Validation Manager is the same on the Windows, Solaris, and Linux platforms. This is the default method of installation for all platforms.

CAUTION: On a Solaris or Linux platform, install Validation Manager as the "root" user or a member of the nfast group to allow operation with nCipher cryptographic hardware. If you install as "root", you must specify a user other than "root" in step 7. This user must also belong to the nfast group.

To perform a GUI-based installation:

1. Do one of the following:
   - On Windows, from either the CD **\Windows\uncompressed** directory or a temporary directory to which you have unzipped the zipped files, double-click **setup.bat**. (Your version of unzip or WinZip must support long filenames.)
   - On Solaris:

     Note: If the GUI-based installation is invoked from outside a graphical environment on Solaris (from a command line outside a graphical desktop), a console installation is launched.

     a. Copy the Validation Manager .tar file from the **/Solaris** directory on the CD to a temporary directory and untar it.

     b. Change the current directory to the directory where the Validation Manager installer file has been untarred.

     c. Run the **setup** file.
   - On Linux:

     Note: If the GUI-based installation is invoked from outside a graphical environment on Linux (from a command line outside a graphical desktop), a console installation is launched.

     a. Copy the Validation Manager .tar file from the **/RH_Linux** directory on the CD to a temporary directory and untar it.

     b. Change the current directory to the directory where the Validation Manager installer file has been untarred.

     c. Run the **setup** file.
2. Read the introductory text and click **Next**.

Note: You can click **Previous** to return to a previous step in the installation process. To terminate the installation at any time, click **Cancel**.

3. Select the appropriate license agreement and click **Next**.

4. Read the License Agreement and do one of the following:
   - If you accept the License Agreement, select **I accept** and click **Next**.
   - If you do not accept the License Agreement, select **I do NOT accept** and click **Cancel** > **Quit** to stop the installation.
5. Select **New Install** and click **Next**.
6. Do one of the following:
   - To accept the default directory where Validation Manager will be installed, click **Next**.
   - Enter the absolute path to the directory where you want to install Validation Manager and click **Next**.

     Important: The absolute pathname to the target directory must not start with white space. Any other white space in the pathname is retained.
7. Do one of the following:
   - On Windows, do one of the following:
     - To accept the default Windows service name, click **Next**.
     - Enter the service name under which Validation Manager will run and click **Next**.
   - On Solaris or Linux, do one of the following:

     CAUTION: If you are installing as "root", specify a user other than "root". This user must also belong to the nfast group.

     - To accept the default **UNIX uid** and **UNIX gid** for the account that will be used to run the Validation Server component of Validation Manager, click **Next**.
     - Enter the UNIX user ID and group ID and click **Next**.
8. Provide the data for configuration of Validation Manager and click **Next**.
   For more information, see "Pre-Installation Checklist" on page 26.
9. Select the cryptographic provider and click **Next**.
10. Enter a passphrase to protect the system keys, confirm the system passphrase, and click **Next**.
11. Provide information for the Administrator account:
    a. Accept the default name or enter a login name for the Administrator.
    b. Enter and confirm the Administrator password.
    c. Click **Next**.

12. Review the server configuration information and, if the information is correct, click **Install**.

    The installation proceeds. It may take a few minutes.

    Note: There is no runtime verification of PINs for the nCipher provider during the installation. If you enter an incorrect PIN, the creation of system certificates and keys fails, and the installation stops.

    If the information is incorrect, click **Previous** to reenter the information.

13. On Windows, do one of the following:
    - To accept starting all Validation Manager services, click **Next**.
    - To not start services, clear the checkbox and click **Next**.
14. To exit the installer, click **Done**.

Post-Installation Requirements

If you installed Validation Manager in a Network File System (NFS) mounted directory, modify the LockFile and SSLMutex directives as described in "RSA Validation Manager and Network File System" on page 64.

Silent Installation

You initiate a silent installation of Validation Manager through the command line using either of two methods:

- Through the entry of separate variables:

  ```
  setup -i SILENT -Dvariable=value -Dvariable=value ...
  ```

- Using a Java properties file:

  ```
  setup -f filename
  ```

  where *filename* is a text file with each line containing a variable and value.

  Important: When performing a silent installation with a properties file, the value of a variable is the complete line following the equals sign. Any special characters that are present (such as double or single quotations) may result in unexpected behavior. Trailing white space at the end of a line may cause problems. For example, if the `INPUT_CRYPTO_PROVIDER` line has trailing white space, the system certificates and keys are not created during the installation.

You must specify all of the following variables (for descriptions and defaults, see "Pre-Installation Checklist" on page 26).

Variable	Value
INSTALLER_UI	SILENT
INPUT_CRYPTO_PROVIDER	Software or nCipher

Variable	Value
INPUT_SYSTEM_PASSWORD	*passphrase* (must be eight characters; no white space, with one alphabetical and one numerical character)
USER_INSTALL_DIR	*target directory* (absolute path; must not start with white space)
INPUT_ORG	*organization*
INPUT_ ORG_UNIT	*organizational unit*
INPUT_BASE_SERVICE_NAME	*base service name* (Windows only) **Note:** Validation Manager removes any white space in the Windows service name to avoid problems during the configuration of dependent services.
INPUT_UID	*UNIX uid* (Solaris and Linux only)
INPUT_GID	*UNIX gid* (Solaris and Linux only)
INPUT_SFQDN	*server fully qualified domain name*
INPUT_OCSP_PORT	*port*
INPUT_ADMIN_PORT	*port*
INPUT_UI_PORT	*port*
INPUT_UI_CONNECTOR_PORT	*port* (must be lower than 32768)
INPUT_UI_CONTROL_PORT	*port*
INPUT_ADMIN_NAME	*name*
INPUT_ADMIN_PASSWORD	*passphrase* (must be eight characters; no white space, with one alphabetical character and one numerical character)

Important: If any of the following occur, the silent installation stops:

- You specify an existing directory as the target directory for the installation.
- You specify an existing Windows service name as the Windows service name for the installation.
- You enter an incorrect PIN. (There is no runtime verification of PINs for the nCipher provider during the installation. The creation of system certificates and keys fails.)
- You specify an nCipher provider but none is present. (There is no verification to confirm that an nCipher provider is present.)

A combination of the two methods (variables and Java properties file) is not supported.

To perform a silent installation:

Depending on your platform, do one of the following:

- On Windows, type the appropriate **RVMInstall.exe** command at the command line from either the CD **\Windows\uncompressed** directory or a temporary directory to which you have unzipped the zipped files. (Your version of unzip or WinZip must support long filenames.)
- On Solaris:
  1. Copy the Validation Manager .tar file from the **/Solaris** directory on the CD to a temporary directory and untar it.
  2. Change the current directory to the directory where the Validation Manager installer file has been untarred.
  3. Type the appropriate **setup** command at the command line.
- On Linux:
  1. Copy the Validation Manager .tar file from the **/RH_Linux** directory on the CD to a temporary directory and untar it.
  2. Change the current directory to the directory where the Validation Manager installer file has been untarred.
  3. Type the appropriate **setup** command at the command line.

Post-Installation Requirements

If you installed Validation Manager in a Network File System (NFS) mounted directory, modify the LockFile and SSLMutex directives as described in "RSA Validation Manager and Network File System" on page 64.

Console Installation

This section describes the console installation of Validation Manager for the Solaris or Linux platform through the command line.

CAUTION: Install Validation Manager as the "root" user or a member of the nfast group to allow operation with nCipher cryptographic hardware. If you install as "root", you must specify a user other than "root" in step 7. This user must also belong to the nfast group.

To perform a console installation:

1. Do one of the following:
   - On Solaris, copy the Validation Manager .tar file from the **/Solaris** directory on the CD to a temporary directory and untar it.
   - On Linux, copy the Validation Manager .tar file from the **/RH_Linux** directory on the CD to a temporary directory and untar it.
2. Change the current directory to the directory where the Validation Manager installer file has been untarred.

3. At the command line, type:

   ```
   RVMInstall
   ```

   and press ENTER.
4. Read the introductory text and press ENTER.

Note: You can use the **back** command to return to the previous step in the installation process. To terminate the installation at any time, enter the **quit** command.

5. Select the appropriate license agreement and press ENTER.
6. Read the License Agreement and do one of the following:
   - To accept the terms of the agreement, at the command prompt, type:

     ```
     y
     ```

     and press ENTER.
   - If you do not accept the terms of the agreement, type:

     ```
     n
     ```

     The installation stops.
7. To create a new installation, at the command prompt, type:

   ```
   1
   ```

   and press ENTER.
8. Do one of the following:
   - To accept the default directory where Validation Manager will be installed, press ENTER.
   - Enter the absolute path to the directory where you want to install Validation Manager and press ENTER.

     Important: The absolute pathname to the target directory must not start with white space. Any other white space in the pathname is retained.
9. Do one of the following:

   CAUTION: If you are installing as “root”, specify a user other than “root”. This user must belong to the nfast group.

   - To accept the default **UNIX uid** and **UNIX gid** for the account that is used to run the Validation Server component of Validation Manager, press ENTER.
   - Enter the UNIX user ID and press ENTER, and UNIX group ID and press ENTER.
10. Provide data for configuration of Validation Manager and press ENTER.
    For more information, see “Pre-Installation Checklist” on page 26.

11. Do one of the following:
    - To accept the software-based cryptographic provider, press ENTER.
    - Select the nCipher provider and press ENTER.
12. Enter a passphrase to protect the system keys and press ENTER.
13. Confirm the system passphrase and press ENTER.
14. Do one of the following:
    - To accept the default Administrator name, press ENTER.
    - Enter a login name for the default Administrator and press ENTER.
15. Enter a password to authenticate the default Administrator and press ENTER.
16. Confirm the Administrator password and press ENTER.
17. Review the server configuration information and, if the information is correct, press ENTER.

    The installation proceeds. It may take a few minutes.

    Note: There is no runtime verification of PINs for the nCipher provider during the installation. If you enter an incorrect PIN, the creation of system certificate and keys fails and the installation stops.

    If the information is incorrect, use the **back** command to reenter information.
18. To exit the installer, press ENTER.

Post-Installation Requirements

If you installed Validation Manager in a Network File System (NFS) mounted directory, modify the LockFile and SSLMutex directives as described in "RSA Validation Manager and Network File System" on page 64.

Troubleshooting Installation Problems

During installation, Validation Manager automatically creates a log file:

- On Windows, ***installed-dir*\RSAValidationManager3.1_InstallLog.log**
- On Solaris or Linux, **/*installed-dir*/RSA_ValidationManager3.1_InstallLog.log**

If the installation fails because of a problem with the specified installation directory, or if it fails before you choose the installation directory, Validation Manager places the log file on the desktop for Windows platforms or in the home directory of the current user on a Solaris or Linux platform.

Validation Manager creates an additional log file containing the information entered during installation at **/*installed-dir*/user_input.log**. The passphrase and password you entered during installation are not shown.

The **SystemCA** application generates a log file during installation. This log file can be found at **/*installed-dir*/SystemCA/SystemCA.log**.

For assistance on troubleshooting installation problems, see Appendix B, "Troubleshooting RSA Validation Manager."

Post-Installation Checklist

If you want to use a generic PKCS #11 hardware device to protect your OCSP signer keys, but you did not install the device before installing Validation Manager, install the device and drivers now. For more information, see Appendix C, "Cryptographic Hardware Interoperability."

You can start the Validation Manager services from the Services dialog box on Windows platforms or from the command line on Solaris or Linux platforms. After starting Validation Manager, log in to the Validation Manager GUI Server with the default Administrator name and password entered during installation, and perform the following required tasks:

- Configure which events are logged
- Create additional Administrators
- Configure unattended startup
- Create an OCSP signer (required)
- Create a status source (required)
- Import at least one CA certificate (required)
- Install additional Validation Manager installations to create a cluster (optional)

For more information on these tasks, including starting Validation Manager, see the *Administrator's Guide*.

5 Upgrading RSA Validation Manager

This chapter describes how to upgrade to RSA Validation Manager 3.1. The topics include:

- Tasks you must do before you upgrade
- Pre-upgrade checklist
- Upgrade procedure
- Post-upgrade checklist
- Upgrading a cluster of Validation Manager installations

RSA Validation Manager 3.1 supports upgrades from RSA Validation Manager 3.0. If you are upgrading from RSA Validation Manager 3.0 build100 or later, contact RSA Customer Support for a drop-in upgrade.

You can also use the upgrader to move an existing RSA Validation Manager 3.1 installation to another machine or platform.

Note: RSA Validation Manager 3.1 does not support upgrades from RSA Keon Validation Server 2.0.

Upgrading to RSA Validation Manager 3.1 requires a source directory (an initially installed product) and a target directory for the upgraded product. An archived upgrade package is created as part of the upgrade process. Your previous installation is left intact. You can continue to use the previous installation as a reference, if necessary, and uninstall it when you are ready.

The upgrade package is created from:

- The original installation database modified into a portable and upgradable format
- The **user_input.log** file created during the original installation
- System certificates, keys, requests (PKCS #10 format), and PKCS #12 files created during the original installation

Important: All Administrators are automatically transferred to the upgraded product. Status cache data, imported revocation lists, local revocation indications, and customized configuration files are not preserved over upgrades.

Before You Begin

Before you upgrade Validation Manager, complete the following administrative tasks:

- Back up your entire existing installation. The upgrade process should not modify your original installation, however, if you customized your installation, you may need to refer to the backup to re-create some customizations.
- Verify that you have sufficient disk space available. You need an additional amount of space roughly equal to the space used by your existing installation.
- To upgrade to another machine or platform, transfer any hardware-based system keys and any hardware-based OCSP signer keys to the other machine or platform. For information, see Appendix C, "Cryptographic Hardware Interoperability" or your HSM documentation.
- Shut down all Validation Manager services. For more information, see the *Administrator's Guide*.
- If you intend to verify the CD you received from RSA, see Appendix D, "Media Verification."
- If you are upgrading from RSA Validation Manager 3.0 build31 on a Windows platform, edit the ***installed-dir*\Util\vmbundle.bat** file:

  a. Locate the following lines:

  ```
  @REM  Change to this directory.
  cd "installed_dir\Util"
  ```

  b. Change the second line to:

  ```
  pushd "installed_dir\Util"
  ```

  c. Locate the following lines:

  ```
  del /f /q default_conf.properties
  del /f /q pki.zip
  del /f /q dbxml.zip
  del /f /q *.db.xml
  ```

  d. Change these lines to:

  ```
  del default_conf.properties /f /q
  del pki.zip /f /q
  del dbxml.zip /f /q
  del *.db.xml /f /q
  ```

Pre-Upgrade Checklist

Before you upgrade Validation Manager, make sure you have the following information:

Original Installation Directory

Location of the original installation of Validation Manager.

_____

Target Installation Directory

Location of the upgraded Validation Manager installation (must not start with white space).

_____

Upgrade Package Directory

If using a previously prepared upgrade package, location of the archived upgrade package.

_____

Upgrade Procedure

This section describes how to upgrade Validation Manager. The upgrade procedure starts as part of the Validation Manager installation procedure. For completeness, the upgrade procedure includes the steps in common with the installation procedure.

The supported methods of upgrading Validation Manager are the same as for installation: GUI-based, silent, and console. By default, the method used for upgrading is assumed to be the same as that used for installation.

GUI-Based Upgrade

Creating the Upgrade Package

Prerequisite

If you are upgrading from RSA Validation Manager 3.0 build31 on Windows, edit the ***installed-dir*\Util\vmbundle.bat** file before you begin the upgrade. For instructions see, "Before You Begin" on page 38.

To create the upgrade package through the GUI-based upgrade:

1. Do one of the following:
   - On Windows, from either the CD **\Windows\uncompressed** directory or a temporary directory to which you have unzipped the zipped files, double-click **setup.bat**. (Your version of unzip or WinZip must support long filenames.)

- On Solaris:

  Note: If the GUI-based installation is invoked from outside a graphical environment on Solaris (from a command line outside a graphical desktop), a console installation is launched.

  a. Copy the Validation Manager .tar file from the **/Solaris** directory on the CD to a temporary directory and untar it.

  b. Change the current directory to the directory where the Validation Manager installer file has been untarred.

  c. Run the **setup** file.

- On Linux:

  Note: If the GUI-based installation is invoked from outside a graphical environment on Linux (from a command line outside a graphical desktop), a console installation is launched.

  a. Copy the Validation Manager .tar file from the **/RH_Linux** directory on the CD to a temporary directory and untar it.

  b. Change the current directory to the directory where the Validation Manager installer file has been untarred.

  c. Run the **setup** file.

2. Read the introductory text and click **Next**.

Note: You can click **Previous** to return to the previous step in the upgrade process. To terminate the upgrade at any time, click **Cancel**.

3. Select the appropriate license agreement and click **Next**.
4. Read the License Agreement and do one of the following:
   - If you accept the License Agreement, select **I accept** and click **Next**.
   - If you do not accept the License Agreement, select **I do NOT accept** and click **Cancel** > **Quit** to stop the upgrade.
5. Select **Upgrade** and click **Next**.
6. Enter the directory of the original installation and click **Next**.
7. To accept stopping all Validation Manager services, click **Next**.
   The upgrade package, **bundle.zip**, is created in the ***/installed-dir***/**Util** directory.
8. Do one of the following:
   - To continue the upgrade on the same machine, click **Next**.
     Complete the upgrade according to the instructions in the following section, "Completing the Upgrade," beginning with step 7.

- To continue the upgrade on a different machine or platform, click **Cancel** to stop the upgrade process.

  A message is displayed stating that Validation Manager will not be installed. This message can be ignored.

  Complete the upgrade according to the instructions in the following section, "Completing the Upgrade," beginning with step 1.

Completing the Upgrade

Prerequisite

If you are completing the upgrade on a different machine or platform than the original installation, ensure that the upgrade package is accessible from the machine to which you are upgrading.

To complete the upgrade through the GUI-based upgrade:

1. Do one of the following:
   - On Windows, from either the CD **\Windows\uncompressed** directory or a temporary directory to which you have unzipped the zipped files, double-click **setup.bat**. (Your version of unzip or WinZip must support long filenames.)
   - On Solaris:

     Note: If the GUI-based installation is invoked from outside a graphical environment on Solaris (from a command line outside a graphical desktop), a console installation is launched.

     a. Copy the Validation Manager .tar file from the **/Solaris** directory on the CD to a temporary directory and untar it.

     b. Change the current directory to the directory where the Validation Manager installer file has been untarred.

     c. Run the **setup** file.
   - On Linux:

     Note: If the GUI-based installation is invoked from outside a graphical environment on Linux (from a command line outside a graphical desktop), a console installation is launched.

     a. Copy the Validation Manager .tar file from the **/RH_Linux** directory on the CD to a temporary directory and untar it.

     b. Change the current directory to the directory where the Validation Manager installer file has been untarred.

     c. Run the **setup** file.
2. Read the introductory text and click **Next**.

Note: You can click **Previous** to return to the previous step in the upgrade process. To terminate the upgrade at any time, click **Cancel**.

3. Select the appropriate license agreement and click **Next**.
4. Read the License Agreement and do one of the following:
   - If you accept the License Agreement, select **I accept** and click **Next**.
   - If you do not accept the License Agreement, select **I do NOT accept** and click **Cancel** > **Quit** to stop the upgrade.
5. Select **Upgrade** and click **Next**.
6. Enter the path to the upgrade package created in step 7 of "Creating the Upgrade Package" and click **Next**.
7. Enter the directory of the upgrade installation.

   Important: The absolute pathname to the target directory must not start with white space. Any other white space in the pathname is retained.

8. Depending on your platform, do one of the following:
   - On Windows, do one of the following:
     - To accept the current Windows service name, click **Next**.
     - Enter a new service name under which Validation Manager will run and click **Next**.
   - On Solaris or Linux, do one of the following:

     CAUTION: If you are installing as "root", specify a user other than "root". This user must belong to the nfast group.

     - To accept the current **UNIX uid** and **UNIX gid** for the account that will be used to run the Validation Server component of Validation Manager, click **Next**.
     - Enter a new UNIX user ID and group ID and click **Next**.
9. Provide new configuration data for Validation Manager or accept current values, and click **Next**.

   For more information, see "Pre-Installation Checklist" on page 26.
10. Review the configuration data and click **Install**.

    The upgrade proceeds. It may take a few minutes.
11. On Windows, do one of the following:
    - To accept restarting all Validation Manager services, click **Next**.
    - To not restart services, clear the checkbox and click **Next**.
12. To exit the upgrader, click **Done**.

Silent Upgrade

You initiate a silent upgrade of Validation Manager through the command line using either of two methods:

- Through the entry of variables:

  ```
  setup -i SILENT -Dvariable=value -Dvariable=value ...
  ```

- Using a Java properties file:

  ```
  setup -f filename
  ```

  where *filename* is a text file with each line containing a variable and value.

Important: When performing a silent upgrade with a properties file, the value of a variable is the complete line following the equals sign. Any special characters that are present (such as double or single quotations) may result in unexpected behavior. Trailing white space at the end of a line may cause problems.

You must specify the following variables (for descriptions and defaults, see "Pre-Installation Checklist" on page 26).

Variable	Value
INSTALLER_UI	SILENT
INPUT_UPGRADE_SOURCE	*original directory* (not needed if upgrading from a previously prepared upgrade package)
USER_INPUT_UPGRADE_SOURCE	*original directory* (specifies to create an upgrade package - not needed if upgrading from a previously prepared upgrade package) **Note:** If you are upgrading without a previously prepared upgrade package, you must specify both INPUT_UPGRADE_SOURCE and USER_INPUT_UPGRADE_SOURCE.
INPUT_UPGRADE_ARCHIVE	*location of upgrade package* (only needed if upgrading from a previously prepared upgrade package) **Note:** Use the GUI or console upgrade procedure to create an upgrade package.
INPUT_CRYPTO_PROVIDER	Software or nCipher
INPUT_SYSTEM_PASSWORD	*passphrase* (must be eight characters; no white space, with one alphabetical character and one numerical character)
USER_INSTALL_DIR	*target directory* (absolute path; must not start with white space)

Variable	Value
INPUT_ORG	*organization*
INPUT_ ORG_UNIT	*organizational unit*
INPUT_BASE_SERVICE_NAME	*base service name* (Windows only) **Note:** Validation Manager removes any white space in the Windows service name to avoid problems during the configuration of dependent services.
INPUT_UID	*UNIX uid* (Solaris and Linux only)
INPUT_GID	*UNIX gid* (Solaris and Linux only)
INPUT_SFQDN	*server fully qualified domain name*
INPUT_OCSP_PORT	*port*
INPUT_ADMIN_PORT	*port*
INPUT_UI_PORT	*port*
INPUT_UI_CONNECTOR_PORT	*port* (must be lower than 32768)
INPUT_UI_CONTROL_PORT	*port*
INPUT_ADMIN_NAME	*name*
INPUT_ADMIN_PASSWORD	*passphrase* (must be eight characters; no white space, with one alphabetical character and one numerical character)

A combination of the two methods (variables and Java properties file) is not supported.

To upgrade Validation Manager to a new machine or platform, use the GUI or console upgrade to create an upgrade package (for instructions, see "GUI-Based Upgrade" on page 39 or "Console Upgrade" on page 45), then specify the INPUT_UPGRADE_ARCHIVE variable in the silent upgrade.

Prerequisite

If you are upgrading from RSA Validation Manager 3.0 build31 on Windows, edit the ***installed-dir*\Util\vmbundle.bat** file before you begin the upgrade. For instructions see, "Before You Begin" on page 38.

To perform a silent upgrade:

Depending on your platform, do one of the following:

- On Windows, type the appropriate **RVMInstall.exe** command at the command line from either the CD **\Windows\uncompressed** directory or a temporary directory to which you have unzipped the zipped files. (Your version of unzip or WinZip must support long filenames.)
- On Solaris:
  1. Copy the Validation Manager .tar file from the **/Solaris** directory on the CD to a temporary directory and untar it.
  2. Change the current directory to the directory where the Validation Manager upgrader file has been untarred.
  3. Type the appropriate **setup** command at the command line.
- On Linux:
  1. Copy the Validation Manager .tar file from the **/RH_Linux** directory on the CD to a temporary directory and untar it.
  2. Change the current directory to the directory where the Validation Manager upgrader file has been untarred.
  3. Type the appropriate **setup** command at the command line.

Console Upgrade

This section describes the console upgrade of Validation Manager for the Solaris or Linux platform through the command line.

To upgrade Validation Manager to a new machine or platform, the upgrade process consists of two steps: creating the upgrade package and upgrading using that package during a separate invocation of the upgrader.

To perform a console upgrade:

1. Do one of the following:
   - On Solaris, copy the Validation Manager .tar file from the **/Solaris** directory on the CD to a temporary directory and untar it.
   - On Linux, copy the Validation Manager .tar file from the **/RH_Linux** directory on the CD to a temporary directory and untar it.
2. Change the current directory to the directory where the Validation Manager upgrader file has been untarred.
3. At the command line, type:

   ```
   RVMInstall
   ```

   and press ENTER.
4. Read the introductory text and press ENTER.

Note: You can use the **back** command to return to the previous step in the upgrade process. To terminate the upgrade at any time, use the **quit** command.

5. Select the appropriate license agreement and press ENTER.
6. Read the License Agreement text and do one of the following:
   - To accept the terms of agreement, at the command prompt, type:

     ```
     y
     ```

     and press ENTER.
   - If you do not accept the terms of the agreement, type:

     ```
     n
     ```

     The upgrade stops.
7. To upgrade an existing installation, at the command prompt, type:

   ```
   2
   ```

   and press ENTER.
8. Do one of the following:
   - Enter the directory of the original installation and press ENTER.
   - Enter the path to the previously prepared upgrade package and press ENTER.
9. All services must be shut down before upgrading. Do one of the following:
   - To accept stopping all Validation Manager services, type:

     ```
     y
     ```

     and press ENTER.
     The upgrade package is created at ***installed-dir*/Util/bundle.zip**.
   - To not stop services, type:

     ```
     n
     ```

     The upgrade stops.
10. Do one of the following:
    - To continue the upgrade using this upgrade package, press ENTER.
    - To exit the upgrader, type:

      ```
      quit
      ```

      and press ENTER.
11. Do one of the following:

    CAUTION: If you are installing as “root”, specify a user other than “root”. This user must belong to the nfast group.

    - To accept the current **UNIX uid** and **UNIX gid** for the account that will be used to run the Validation Server component of the Validation Manager, press ENTER.
    - Enter a new UNIX user ID and press ENTER, and UNIX group ID and press ENTER.

12. Provide new configuration data for Validation Manager or accept current values, and press ENTER.

    For more information, see "Pre-Installation Checklist" on page 26.

    The upgrade proceeds. It may take a few minutes.

13. To exit the upgrader, press ENTER.

Troubleshooting Upgrade Problems

During the upgrade, Validation Manager automatically creates a log file:

- On Windows, ***installed-dir****RSAValidationManager3.1_InstallLog.log**
- On Solaris or Linux, ***/installed-dir***/**RSA_ValidationManager3.1_InstallLog.log**

If the upgrade fails because of a problem with the specified original or upgrade directories, or if it fails before the upgrade directory has been chosen, Validation Manager places the log file on the desktop on a Windows platform or in the home directory of the current user on a Solaris or Linux platform.

Validation Manager also creates an additional log file containing the information entered during the upgrade at ***/installed-dir***/**user_input.log**.

For assistance on troubleshooting upgrade problems, see Appendix B, "Troubleshooting RSA Validation Manager."

Post-Upgrade Checklist

Immediately after upgrading Validation Manager, you may want to perform these tasks:

- Configure which events to log and which events must be logged to determine event success.
- (Optional) Change the passphrases (see the changepp utility described in the *Administrator's Guide*).

You must restart the Validation Manager services from the Services dialog box on Windows platforms if you elected not to restart them during the upgrade, or from the command line on Solaris or Linux platforms.

For more information on these tasks, including restarting Validation Manager, see the *Administrator's Guide*.

Upgrading a Cluster

If you are upgrading a cluster of Validation Manager installations, upgrade the primary node before you upgrade the secondary nodes.

A cluster continues to function with a combination of RSA Validation Manager 3.1 nodes and RSA Validation Manager 3.0 nodes. However the available functionality depends upon which node is processing the request. RSA Validation Manager 3.0 nodes cannot provide RSA Validation Manager 3.1 functionality.

6 Clustering with RSA Validation Manager

This chapter describes how to set up a cluster of RSA Validation Manager installations. The topics include:

- Tasks you must do before you set up a cluster
- Setting up the primary node
- Setting up the secondary nodes

Clustering is supported on Windows, Solaris, and Linux platforms. All machines in the cluster must have the same platform type.

To appear to OCSP clients to be one OCSP server, you must use a third-party load balancer. When used with a load balancer, a cluster of Validation Manager nodes provides OCSP status without interruption if all but one of the nodes fails. However, performance will likely degrade. Failed nodes can be restarted and will automatically join the cluster again. The configuration and operation of a load balancing component, whether software or hardware, is outside the scope of this document. Contact RSA Customer Support for information on which load balancers Validation Manager supports.

Each node in the cluster is a separate installation of Validation Manager. You can only administer Validation Manager through the primary node. Secondary nodes act like the primary node in all ways except that only the primary can write to the database. If the primary node fails (for any reason), one of the secondary nodes takes over as the primary.

You may want the cluster to act as a synchronization server. You must configure each node in the cluster as a synchronization server and the load balancer must be set up between the synchronized installations and the cluster. For more information on synchronization, see the *Administrator's Guide*.

Note: The database (DB) used in Validation Manager is the Berkeley DB. For information on the Berkeley DB, go to **www.oracle.com/database/berkeley-db/index.html**.

Before You Begin

Before you set up a cluster of Validation Managers, make sure you have the following information.

Primary Node

Hostname and the clustering VirtualHost port number of the Validation Manager installation that is the primary node. Default port number is **1225**.

Cluster Nodes

Hostname and the clustering VirtualHost port number of the Validation Manager installations that are the secondary nodes. Default port number is **1225**.

_____

_____

_____

_____

Setting Up the Primary Node

The initial installation can be a new installation of Validation Manager that you are about to install or a previous version of Validation Manager that you are about to upgrade. (You do not need to configure the initial installation as the master once other nodes are added to the cluster.)

To set up the primary node, you must:

1. Install Validation Manager on one of the designated machines or upgrade a current installation.
2. Start Validation Manager services, if you did not start them during installation.
3. Perform tasks to bring Validation Manager into a useful, functioning state.
4. Stop Validation Manager services.
5. Modify the **httpd.conf** file to add the clustering VirtualHost definition and any required clustering-related directives.
6. Back up the Validation Manager installation (now the primary node).
7. Restart Validation Manager services.

Install or Upgrade RSA Validation Manager

For instructions, see Chapter 4, "Installing RSA Validation Manager" or Chapter 5, "Upgrading RSA Validation Manager."

Start RSA Validation Manager

If you did not start Validation Manager services during installation, start them now. For instructions on starting Validation Manager, see the *Administrator's Guide*.

Using RSA Validation Manager the First Time

Important: RSA recommends that you perform these tasks before setting up the cluster to avoid additional copying of HSM data and unnecessary restarting of secondary nodes.

Perform the following tasks listed in "Post-Installation Checklist" on page 35:

- Create an OCSP signer
- Create a status source
- Import at least one CA certificate

For more information on these tasks, see the *Administrator's Guide*.

Stop RSA Validation Manager

For instructions on stopping Validation Manager, see the *Administrator's Guide*.

Modify the httpd.conf file

To modify httpd.conf to set up the primary node:

1. Back up ***/installed-dir*/ValidationServer/conf/httpd.conf**.
2. In a text editor, open the **httpd.conf** file.
3. Ensure that the clustering VirtualHost definition is not commented out (lines do not start with #). For example:

```
Listen 1225
<VirtualHost _default_:1225>
...
</VirtualHost>
```

   Important: If port number 1225 is already in use on this machine, you must select another port number. Remember to replace 1225 with the new port number in all locations on the primary node and on all secondary nodes.

4. Ensure that the Clustering directive is set to **on**. For example:

```
Clustering on
```

5. Ensure that the ClusterDefaultMaster directive is set to **true**. For example:

```
ClusterDefaultMaster true
```

6. For every node you want to add to the cluster, enter the node hostname (as a fully qualified domain name) and port number in a ClusterNode directive. For example:

```
ClusterNode node1.user.net 1225
ClusterNode node2.user.net 1225
```

Note: RSA recommends that you specify the hostnames and port numbers of every node in the cluster in the **httpd.conf** file of the primary node to provide better fault tolerance. You can add a node later that is not specified in the **httpd.conf** file and the cluster continues to function correctly (including the new node). To restore the highest order of fault tolerance, you can add the new node to the **httpd.conf** file of every node by entering a new ClusterNode directive. You must restart Validation Manager services on every node for any changes to take effect.

7. Optionally, modify any of the other clustering directives.

Important: If the directive does not appear in **httpd.conf**, Validation Manager uses the default value. Changing some directives may have an impact on performance or database consistency. RSA recommends that you consider your priorities before making any such changes.

Directive	Description
ClusterPriority	Specifies the order in which nodes take over as the primary, if the default primary fails. It is ignored if ClusterDefaultMaster is set to **true**. The default is **1000**, the maximum is **2147483648**.
ClusterElectionTimeout	Specifies how long, in seconds, a node waits for an election to complete. The default is **5**.
ClusterNoFlush	Specifies whether database transactions are written to disk when committed. Setting this directive to **false** may enhance performance. The default is **true**.
ClusterWaitForPermanent	Specifies whether a node waits for a replication message (messages sent amongst nodes by the Berkeley DB to support database replication) to be permanently recorded before responding to the message. This may necessitate waiting for any missing messages (messages can be received out of order). The default is **true**.
ClusterMaxRequiredAcks	Specifies the level of data consistency between the nodes based on the maximum number of positive acknowledgements to wait for when broadcasting replication messages to other nodes. Set to a value greater than the number of nodes to maximize consistency or set to a value less than the number of nodes to enhance performance at the expense of data consistency. The default is **1000**, the maximum is **2147483648**.

Directive	Description
ClusterAckSlack	Specifies the level of data consistency between the nodes based on the elapsed time between positive acknowledgements from any particular node. The node does not wait for an acknowledgement from a node if it has previously received a positive acknowledgement from that node within the number of seconds specified. Increase the value to enhance performance at the expense of data consistency. The default is **0**.

8. On Solaris or Linux, in the **workerMPM** section:
   - Locate the MaxClients directive and change the value to at least **200**.

     CAUTION: RSA recommends that, in association with the preceding change to the MaxClients directive in the **httpd.conf** file, you change the file descriptor limit to **1024**. The file descriptor is the maximum number of open files and network connections that a process is allowed to have open at the same time. (If you intend to increase MaxClients further for other reasons, consider increasing the file descriptor limit as well.)

     RSA recommends that you have a basic understanding of Apache administration before making these changes. Otherwise, for more information, contact RSA Customer Support.

   - Locate the ThreadLimit directive and change the value to at least **200**.
   - Locate the MaxSpareThreads directive and change the value to at least **200**.
   - Locate the ThreadsPerClient directive and change the value to at least **200**.
9. Save and close the file.

Back Up the Validation Manager Installation

Back up the Validation Manager database and all system certificates, keys, and requests from the primary node.

To back up the database:

1. Change the current directory to **/*installed-dir*/Util/**.
2. At the command prompt, type:

   ```
   vmarchive
   ```

   Validation Manager creates a backup of the database and all system certificates, keys, and requests in **/*installed-dir*/Util/archive.zip**.

Restart Validation Manager

For instructions on starting Validation Manager, see the *Administrator's Guide*.

Setting Up Secondary Nodes

The other nodes in the cluster are new installations of Validation Manager. (You do not need to configure the initial installation as the primary once other nodes are added to the cluster.)

To set up a secondary node, you must:

1. Install Validation Manager on another of the designated machines.
2. Save a copy of the primary Validation Manager backup on the new installation.
3. Modify the **httpd.conf** file to add the clustering VirtualHost definition and any required clustering-related directives.
4. Copy any hardware-based system keys or any hardware-based OCSP signer keys created during the installation and initial configuration of the primary node.
5. Restart Validation Manager services.

Important: After database replication is complete with the cluster, a "Replication initialization complete" message appears in the trace log file. RSA recommends that you do not add any additional nodes to the cluster until after this message is logged.

Install Validation Manager

For installation instructions, see Chapter 4, "Installing RSA Validation Manager." Do not start the Validation Manager services at the end of the installation or after the installation is finished.

Copy the Primary Validation Manager Database

Copy the Validation Manager database and all system certificates, keys, and requests from the primary node.

Important: If a secondary node is shut down (for any reason), RSA recommends that you back up the primary node (this can be performed while the primary is running), and copy only the database to the secondary node. To copy the database only, type:

```
vmrestore archive.zip
```

You can then restart the secondary node.

Prerequisites

If a load balancer is used to route OCSP requests from external clients to Validation Manager, **OCSPServer.cert** and **SyncServer.cert** must contain the FQDN of the load balancer. (External applications expect any server certificate to contain the same FQDN as the one they are sending the request to.) By default, the system certificates **UIServer.cert**, **OCSPServer.cert**, and **SyncServer.cert** on the secondary node have the fully qualified domain name (FQDN) of the primary node after copying the database and the other files.

Modify the appropriate template files (**ocsp.template** and **syncserver.template** respectively) to change the current FQDN to the FQDN that you want (specifically, the lines beginning with `CN=` and `2.5.29.17=`). For more information on generating new system certificates, see the *Administrator's Guide.*

If you intend to use a load balancer, RSA recommends that you make the template modifications before running the vmrestore utility.

To save a copy of the primary Validation Manager database on the secondary node:

1. Change the current directory to /***installed-dir***/**Util/**.
2. Copy /***installed-dir***/**Util/archive.zip** from the primary node to /***installed-dir***/**Util/archive.zip** on the secondary node.
3. At the command prompt, type:

   ```
   vmrestore archive.zip replicate system passphrase
   ```

   where

archive.zip	The database backup created on the primary node (see "Back Up the Validation Manager Installation" on page 53).
replicate	Specifies that the database, system certificate and keys, and OCSP signer certificate requests are saved.
system passphrase	The system passphrase you entered when you installed Validation Manager on the secondary node. It is required to re-sign **UIServer.cert**, **OCSPServer.cert**, and **SyncServer.cert**.

Modify the httpd.conf file

To set up the secondary node:

1. Back up /***installed-dir***/**ValidationServer/conf/httpd.conf**.
2. In a text editor, open the **httpd.conf** file.
3. Ensure that the clustering VirtualHost definition is not commented out (lines do not start with #). For example:

   ```
   Listen 1225
   <VirtualHost _default_:1225>
   ...
   </VirtualHost>
   ```

   Important: If port number 1225 is already in use on this machine, you must select another port number. Remember to replace 1225 with the new port number in all locations on the primary node and on all secondary nodes.

4. Ensure that the Clustering directive is set to **on**. For example:

   ```
   Clustering on
   ```

5. Ensure that the ClusterDefaultMaster directive is set to **false**. For example:

```
ClusterDefaultMaster false
```

6. For every other node you want to add to the cluster (including the primary node), enter the node hostname (as a fully qualified domain name) and port number in a ClusterNode directive. For example:

```
ClusterNode node1.user.net 1225
ClusterNode node2.user.net 1225
```

Note: RSA recommends that you specify the hostnames and port numbers of every node in the cluster in the **httpd.conf** file of the secondary node to provide better fault tolerance. You can add a node later that is not specified in the **httpd.conf** file and the cluster continues to function correctly (including the new node). To restore the highest order of fault tolerance, you can add the new node to the **httpd.conf** file of every node by entering a new ClusterNode directive. You must restart Validation Manager services on every node for any changes to take effect.

7. Optionally, modify any of the other clustering directives.

Important: If the directive does not appear in **httpd.conf**, Validation Manager uses the default value. Changing some directives may have an impact on performance or database consistency. RSA recommends that you consider your priorities before making any such changes.

Directive	Description
ClusterPriority	Specifies the order in which nodes take over as the primary, if the default primary fails. It is ignored if ClusterDefaultMaster is set to **true**. The default is **1000**, the maximum is **2147483648**.
ClusterElectionTimeout	Specifies how long, in seconds, a node waits for an election to complete. The default is **5**.
ClusterNoFlush	Specifies whether database transactions are written to disk when committed. Setting this directive to **false** may enhance performance. The default is **true**.
ClusterWaitForPermanent	Specifies whether a node waits for a replication message (messages sent amongst nodes by the Berkeley DB to support database replication) to be permanently recorded before responding to the message. This may necessitate waiting for any missing messages (messages can be received out of order). The default is **true**.

Directive	Description
ClusterMaxRequiredAcks	Specifies the level of data consistency between the nodes based on the maximum number of positive acknowledgements to wait for when broadcasting replication messages to other nodes. Set to a value greater than the number of nodes to maximize consistency or set to a value less than the number of nodes to enhance performance at the expense of data consistency. The default is **1000**, the maximum is **2147483648**.
ClusterAckSlack	Specifies the level of data consistency between the nodes based on the elapsed time between positive acknowledgements from any particular node. The node does not wait for an acknowledgement from a node if it has previously received a positive acknowledgement from that node within the number of seconds specified. Increase the value to enhance performance at the expense of data consistency. The default is **0**.

8. On Solaris or Linux, in the **workerMPM** section:
   - Locate the MaxClients directive and change the value to at least **200**.

     CAUTION: RSA recommends that, in association with the preceding change to the MaxClients directive in the **httpd.conf** file, you change the file descriptor limit to **1024**. The file descriptor is the maximum number of open files and network connections that a process is allowed to have open at the same time. (If you intend to increase MaxClients further for other reasons, consider increasing the file descriptor limit as well.)

     RSA recommends that you have a basic understanding of Apache administration before making these changes. Otherwise, for more information, contact RSA Customer Support.

   - Locate the ThreadLimit directive and change the value to at least **200**.
   - Locate the MaxSpareThreads directive and change the value to at least **200**.
   - Locate the ThreadsPerClient directive and change the value to at least **200**.
9. Save and close the file.

Copy Hardware-Based Keys

If any hardware-based system keys or any hardware-based OCSP signer keys were created during the installation and initial configuration of the primary node, ensure you copy them to the HSMs on the secondary node.

For more information on copying keys to HSMs, see Appendix C, "Cryptographic Hardware Interoperability."

Restart Validation Manager

For instructions on starting Validation Manager, see the *Administrator's Guide*.

Note: Restart Validation Manager on all secondary nodes whenever you create a new OCSP signer at the primary node. Validation Manager only loads keys at startup.

7 Managing Audit Logs

This chapter describes how to manage the RSA Validation Manager audit log files. The topics include:

- Configuring logging
- Managing audit log files

Configuring Logging

The OCSP Server and UI Server log operational events, such as certificate and revocation list import, to a file called an audit log. You can use the Validation Manager Graphical User Interface (GUI) or the command line utility to:

- Enable logging
- Determine when to start a new audit log
- Determine when to sign the audit log
- Configure which events Validation Manager logs

For more information, see the *Administrator's Guide* or the *Command Line Reference Manual*.

Managing Audit Log Files

Validation Manager records log entries for operational and system events, and distributes the audit logs in Extensible Markup Language (XML) format. You can view the audit logs in any text editor or through the use of **cat**, **less**, or **tail** UNIX commands. You can verify an audit log to determine if it has been tampered with.

Validation Manager names audit logs based on the date they are created. For example: vm_auditlog_*YYYYMMDD*[_*N*].xml.

If Validation Manager creates multiple audit logs on the same day, they are numbered consecutively. For example: vm_auditlog_20040620.xml, vm_auditlog_20040620_1.xml, vm_auditlog_20040620_2.xml. Audit logs are stored in the ***/installed-dir*/Audit** directory.

CAUTION: Validation Manager does not back up or delete audit logs, or verify if there is sufficient disk space remaining to log new events or create new audit logs. If there is no disk space remaining, no further logging occurs, and Validation Manager does not display an error message to warn you.

RSA recommends that you do not rename any audit log files.

8 Uninstalling RSA Validation Manager

This chapter describes how to uninstall the RSA Validation Manager software.

You are prompted for confirmation before all services, files, directories, and components of Validation Manager are removed from the installation directory. After the uninstallation is finished, you can remove any files not removed by the uninstaller.

If you installed Validation Manager using the silent installation, the uninstallation process, by default, is also silent (no user interaction is required).

Uninstalling Validation Manager on a Windows Platform

To uninstall Validation Manager:

1. Stop all Validation Manager services.
2. Click **Start > Programs > RSAValidationManager3.1 > Uninstall RSA Validation Manager 3.1**.

   Note: If you changed the Windows service name during installation, the name you specified appears instead of **RSAValidationManager3.1**.

3. If you want all services, files, directories, and components of Validation Manager removed, click **Uninstall**.

   To stop the uninstallation, click **Cancel**.
4. To exit the uninstaller, click **Done**.

Post-Uninstallation Requirements

- During the uninstallation process, Validation Manager may not delete some folders. Delete these folders manually.
- If you installed Validation Manager in an NFS mounted directory, some folders may not be deleted. Delete these folders manually.

Uninstalling Validation Manager on a Solaris or Linux Platform

To uninstall Validation Manager:

1. Stop all Validation Manager services.
2. Change the current directory to the parent directory of /***installed-dir***/.

   For example, if Validation Manager was installed into **/home/username/RSA_Security/RSA_Validation_Manager**, change the current directory to **/home/username/**.

3. At the command prompt, type:

   ```
   /relative path to uninstaller/Uninstall/
   UninstallRSAValidationManager
   ```

   and press ENTER.

   For example, using the example from step 2, type:

   ```
   /RSA_Security/RSA_Validation_Manager/Uninstall/
   UninstallRSAValidationManager
   ```

4. If you want the services, files, directories, and components of Validation Manager created during installation removed, press ENTER to start the uninstallation.

   To stop the uninstallation, use the **quit** command.

   Any files or directories created after installation are not removed.

Post-Uninstallation Requirements

- During the uninstallation process, Validation Manager may not delete some folders. Delete these folders manually.
- If you installed Validation Manager in an NFS mounted directory, some folders may not be deleted. Delete these folders manually.

A Configuring RSA Validation Manager

This appendix describes the server configuration files found in RSA Validation Manager.

CAUTION: Incorrect changes to the configuration files can disable Validation Manager or make it unusable. For detailed instructions on how to configure Validation Manager, see the appropriate chapter in this guide, the *Administrator's Guide*, or the SecurCare Online knowledge database. Discuss any additional changes you want to make with RSA Customer Support.

RSA Validation Manager Configuration Files

Modifications to a configuration file take effect only after the modified server is restarted.

OCSP Server Configuration Settings

The OCSP Server configuration file defines aspects of the Validation Manager OCSP Server. This file is located at **/*installed-dir*/ValidationServer/conf/httpd.conf**.

Many of the settings in this configuration file are self-explanatory or explained in the documentation for Apache. These settings, such as the document root, are generic to any web server. A number of other settings represent Validation Manager variables such as the name and location of the trace log file.

Passphrase Settings

The passphrase and PIN prompting configuration file defines the behavior of the passphrase prompting service. Validation Manager generates the file during installation. It contains information required by the prompting service (whether Validation Manager prompts for a passphrase or PIN). This file is located at **/*installed-dir*/ValidationServer/bin/ppprompt.conf**.

SSL/TLS Configuration Settings

The SSL/TLS configuration file defines the aspects of secure communications with the Validation Manager UI and OCSP Servers. This file is located at **/*installed-dir*/ValidationServer/conf/ssl.conf**.

Within this file, you can set the type of authentication to use to access the Validation Manager GUI (by default, Validation Manager supports only user ID and password authentication) and look up the location of the system certificate and key files in case they must be replaced.

Unattended Startup Configuration Settings

Unattended startup is the starting or restarting of Validation Manager without having to enter passphrases. If you configure Validation Manager for unattended startup and the server shuts down, you can set up Validation Manager to automatically restart when power is restored and the server restarted.

You configure Validation Manager for unattended startup by creating a **startup.conf** file in the ***/installed-dir*** directory and verifying that the StartupFile directive is in the ***/installed-dir*/ValidationServer/bin/ppprompt.conf** file. On a Solaris or Linux platform, make the file permission and ownership of the file the same as the Validation Manager installation.

Important: Creating a **startup.conf** file is a convenient means to safeguard against the accidental shutdown of Validation Manager, however, you must be sure to safeguard the .conf file. Placing a passphrase in a text file or .conf file on a hard disk is not a secure practice because passphrases are stored in clear text.

RSA Validation Manager and Network File System

If you installed Validation Manager in an NFS mounted directory, you must modify the LockFile and the SSLMutex directives to place the lockfile and the mutual exclusion semaphore on a local disk.

The LockFile directive is located in the ***/installed-dir*/ValidationServer/conf/httpd.conf** file. The SSLMutex directive is located in the ***/installed-dir*/ValidationServer/conf/ssl.conf** file.

The syntax of the LockFile directive is:

```
LockFile file path
```

CAUTION: Avoid putting the lockfile in a world-writable directory. Someone could create a denial-of-service attack by creating lockfiles in such a directory. For more information, go to **http://httpd.apache.org/docs-2.0/mod/mpm_common.html#lockfile**.

The syntax of the SSLMutex directive is:

```
SSLMutex file: file path
```

CAUTION: RSA recommends that the SSLSessionCache directive remain commented out. Enabling the external session caching can lead to memory leaks and occasional server failures. Within Validation Manager, SSL sessions are cached internally by default. For more information on SSL directives, go to **http://httpd.apache.org/docs-2.0/mod/mod_ssl.html**.

B Troubleshooting RSA Validation Manager

This appendix contains solutions to problems that you may encounter during the RSA Validation Manager installation or upgrade process.

If a problem occurs during installation or upgrade, check the following:

- For last-minute information or bugs that exist in the installation or upgrade process, see the *Readme*, available from the /**Documentation** directory on the product CD.
- Check one of the following log files or the following section for a possible solution.

During the installation or upgrade of Validation Manager, log files are created. If an error occurs during Validation Manager installation or upgrade, check the following log files to find out what error or errors occurred:

- ***installed-dir****RSAValidationManager3.1_InstallLog.log** logs installation events on Windows platforms
- /***installed-dir***/**RSA_ValidationManager3.1_InstallLog.log** logs installation events on Solaris and Linux platforms
- /***installed-dir***/**user_input.log** logs data entered during the installation

Solutions

Entered incorrect passphrase or password as confirmation

If you enter an incorrect passphrase or password as confirmation during installation, Validation Manager prompts you twice to enter the correct confirmation. If you made a mistake entering the initial passphrase or password, type `quit` to stop the installation. In this case, all directories created as a part of the installation are removed.

Validation Server service does not start up

If you tried to start services, either during installation or afterward, and the Validation Server service did not start, you may have entered a port number that was already in use, and the Validation Server service could not bind to the port. You can find more information in the system logs.

Do one of the following:

- Uninstall and reinstall Validation Manager. Verify all the port numbers you enter.

- If you know which port number is at fault, modify the appropriate configuration file or files.

  To change the port number, make changes in all appropriate files. The following table shows the virtual hostname and the corresponding configuration files in which you must make changes.

Virtual Host	Configuration Files
OCSP Server	**/ValidationServer/conf/httpd.conf** **/ValidationServer/bin/ppprompt.conf**
UI Server	**/ValidationServer/conf/ssl.conf** **/ValidationServer/conf/worker2.properties** **/ValidationServer/bin/ppprompt.conf** **/Util/Default.conf**
vmadmin application	**/ValidationServer/conf/ssl.conf** **/ValidationServer/bin/ppprompt.conf** **/Util/Default.conf**

C Cryptographic Hardware Interoperability

This appendix provides information on how RSA Validation Manager and cryptographic hardware products work together.

Validation Manager is interoperable with nCipher nForce and nShield hardware security modules (HSMs) as well as other generic PKCS #11 hardware devices. You can use nCipher HSMs for system keys and OCSP signer keys, whereas you can only use a generic PKCS #11 device for OCSP signer keys. Validation Manager can support only one HSM at a time.

For more information about RSA Secured products, visit the RSA Customer Support web site for RSA Secured Implementation Guides.

Note: The RSA Interoperability Lab tests products on an ongoing basis. For additional interoperability information, visit the RSA Customer Support web site.

nCipher nForce and nShield

Manufacturer: nCipher Corporation Ltd.

Web site: www.ncipher.com

Introduction

nCipher nForce and nShield are devices that generate and protect private keys, and provide secure hardware key management for the Validation Manager. nForce and nShield are based on the powerful nCipher encryption acceleration technology.

When using nCipher hardware (nForce and nShield), keys are never revealed to the outside world or even to the main memory of the Validation Manager installation in an unencrypted format, vastly increasing the security of key data. Validation Manager uses nCipher hardware to generate keys using true hardware-based random number generation, encrypt keys for secure storage, and guarantee key security in highly sensitive applications where federal standards level security is critical. Depending on how they are used, nCipher hardware devices are Federal Information Processing Standard 140 (FIPS 140-1) Level 3 compliant.

Validation Manager supports the use of nCipher hardware modules nShield and nForce, and the nCipher software packages as shown in the following table.

Microsoft Windows Server 2003	Sun Solaris 9 or 10	Red Hat Enterprise Linux 5
10.01	10.02	10.15

Functionality

Keypair Generation

Validation Manager can create keys using nCipher smart cards. These smart cards support key generation for key pairs based on RSA algorithms.

Key Storage

Validation Manager supports the storage of cryptographic keys using nCipher smart cards.

Signing with Keys

Validation Manager supports the usage of keys stored in nCipher key storage for signing operations.

Important: Ensure that the required nCipher smart card is inserted into the nCipher card reader before any Validation Manager operations if the card set is non-persistent. The smart card can be removed from the card reader if the card set is persistent.

Configuration

To use the KeySafe application, you must install Java 1.3 or later before you install the nCipher hardware and software.

Installing nCipher Hardware

Full installation instructions are available in the nCipher *Getting Started Guide* on the nCipher installation CD. Instructions for upgrading the module firmware are available in the nCipher *nForce User Guide* or *nShield User Guide.*

Installing the nCipher Server Software

Installation instructions are available in the nCipher user guide on the nCipher installation CD.

Note: On some older versions of the nCipher hardware, the server software cannot be upgraded to v6.14 because the older versions of the server software are no longer supported by nCipher.

If the module type code is “2” or “4” (obtained by running the enquiry command), the nCipher software cannot be upgraded beyond v5.x.

nCipher Security World

Before you can use the nCipher HSM, you must create a Security World. A Security World consists of one or more hardware modules, a set of smart cards, and some encrypted data stored on a computer.

In order to create a Security World, you must set the hardware module in pre-initialization mode, create a Security World using the KeySafe application or the new-world command, and finally set the hardware module in operational state. Detailed information on the Security World can be found in the nCipher user guide.

Card Sets in the Security World

A Security World is designed to ensure that all keys remain secure throughout their life cycle. Within a given security world, there are two types of card sets: an Administrator Card Set and Operator Card Sets.

The Administrator Card Set controls access to recovery functions, and is created during Security World initialization. There is only one Administrator Card Set for each Security World.

Operator Card Sets control access to application keys. Each user can access only the keys protected by the Security World and protected by their Operator Card Set. When you initialize a smart card, you must provide new Operator Card Set passphrases. Validation Manager uses the Operator Card Set smart cards to protect public and private keys. For more information on smart cards, see the nCipher user guide.

Note: Erase the smart cards used as Operator Cards before reinitializing the nCipher module. Otherwise, these cards must be discarded because they cannot be used, erased, or reformatted without the old Security World key.

nCipher Smart Card Labels

Smart card labels are defined when the smart card is initialized. They are a name that you attach to the smart card to help you organize your smart cards and keep track of what each one is used for. Labels can only be changed by reinitializing the smart card.

Initializing nCipher Smart Cards for Use with Validation Manager

Initialize nCipher smart cards before you use them with Validation Manager. Use one of the following methods to initialize a new smart card:

- Using the nCipher KeySafe utility

  Initialize nCipher smart cards using the nCipher KeySafe utility. For instructions, see “Creating Operator Card Sets” in the nCipher user guide.

- Using createoc-simple

  ```
  path/createoc-simple [--force] module slot label persist timeout
  ```

  where:

--force	Allows for the overwriting of non-blank cards
module	Module number of the HSM, usually **1**
slot	Usually **0**
label	Name of token
persist	Must be **yes** or **no**
timeout	Must be **0**

For example:

```
createoc-simple 1 0 token1 no 0
```

Refer to the nCipher user guide on the nCipher installation CD. Instructions for using createoc-simple are available in “Creating Operator Card Sets.”

- Using createocs or ckinittoken

  Refer to the nCipher user guide on the nCipher installation CD. Instructions for using createocs are available in “Creating Operator Card Sets”.

Important: Validation Manager only supports card sets with K=1.

Validation Manager Installation

If nCipher hardware and server software are installed before Validation Manager, the smart card must be initialized prior to beginning the Validation Manager installation.

CAUTION: Initialize the nCipher smart card with a passphrase.

You must install Validation Manager by the “root” user or by a user belonging to the nfast user group to allow operation with nCipher hardware.

Adding nForce or nShield Support to an Existing Validation Manager Installation

To add smart card support to an existing Validation Manager installation that does not use smart cards, you do not need to reinstall Validation Manager.

To add nCipher smart card support to an existing installation:

1. Install the nCipher hardware and server software as described in previous sections. Make sure the nFast server is running.
2. Initialize a smart card and insert it into the reader.
3. Stop and restart the Validation Manager services.

Recovery Features

All of the following recovery options require that the recovery option must have been enabled when creating the Security World.

Loss of Smart Card

To recover from the loss of an nCipher smart card, you can use the nCipher replaceocs or sw-racs utility to replace any lost smart card. For detailed instructions, see “Replacing Operator Card Sets” or “Replacing the Administrator Card Set” in the nCipher user guide. For the replacement card to work with the existing servers that used the original smart cards:

- Name the replacement card set the same name as the original card set
- Clear and remove the old card set from the Security World

Loss of Hardware

You must replace the actual hardware module (nForce or nShield). After replacing the nCipher hardware, you must recreate the Security World using the new-world command, as documented in "Adding a Module to the Security World" in the nCipher user guide. If the replacement module had been used with another Security World, then you must initialize that module using initunit command before using new-world.

Use this procedure to move the Security World from one machine to another, including across platforms.

Loss of Hard Drive Data

If files containing the Security World data are lost (for example, the **/kmdata** directory), restore the files from backup. Any data created since the last backup is lost. If the complete nCipher software installation is damaged or lost, follow the procedure in the previous section, "Loss of Hardware," provided a backup of the Security World data exists.

Useful nCipher Commands

- **enquiry** confirms versions of server software and module firmware
- **ckcheckinst** confirms the nCipher PKCS #11 library version and determines the label of any cards inserted in the reader
- **nfkminfo** obtains information about the Security World, such as, whether the recovery option is enabled

Generic PKCS #11 Devices

To use a PKCS #11 hardware device, modify the **PKCS11** directive in the Passphrase and PIN prompting configuration file **/*installed-dir*/ValidationServer/bin/ppprompt.conf**. Provide the target slot and filename path to the PKCS #11 shared object. For example, if you want to connect Validation Manager to a Chrysalis Luna SA HSM on a Windows platform, modify the directive in the **ppprompt.conf** file as follows:

```
PKCS11 "2,c:\Program Files\LunaSA\cryptoki.dll"
```

See the RSA Customer Support web site for RSA Secured Implementation Guides for the currently supported PKCS #11 devices.

D Media Verification

With media verification you can verify the integrity of the product shipped to you. You can verify that the distribution media has not been altered since it was signed by RSA.

Media verification is a two-step process. The first step is to validate the media. A second, optional step, if you want a higher degree of assurance, is to validate the verification certificate and the media verification utility.

Validating the Media

To validate the distribution media:

1. At the command prompt, change the directory to the top directory of the product CD.
2. Run the media verification utility from the top directory of the product CD. Do one of the following:
   - On Windows platforms, type:

     ```
     .\Windows\Util\mediaverify.exe .
     ```
   - On Solaris platforms, type:

     ```
     ./Solaris/Util/mediaverify .
     ```
   - On Linux platforms, type:

     ```
     ./RH_Linux/Util/mediaverify .
     ```

3. Review the success message at the end of the output and verify that the media verification utility did not report any errors.

   If there are any errors, consider your distribution to be suspect. Contact RSA Customer Support.

   If the verification is successful, you see output like the following example (output varies depending on the content of the CD).

```
.\Windows\Util\mediaverify.exe: Version 1.0
Signature File: .\media.sig

Verify Signature - passed.
Signer Id:  8afb - 7bfb - 7337 - 6410 - 77d7 - fb12 - 890d - 19fd
Digest provider: XCSP Default Provider

.\Windows\Util\mediaverify.exe - passed
.\Windows\Util - directory passed
.\Windows\uncompressed\setup.bat - passed
.\Windows\uncompressed\RVMInstall.exe - passed
.\Windows\uncompressed\RSAValidationManagerv30README.pdf - passed
.\Windows\uncompressed\Resource1.zip - passed
.\Windows\uncompressed - directory passed
.\Windows\RSAVM-3.0build31r-WIN32.zip - passed
.\Windows - directory passed
.\Solaris\Util\mediaverify - passed
.\Solaris\Util - directory passed
.\Solaris\RSAVM-3.0build31r-sparc-sun-solaris.tar - passed
.\Solaris - directory passed
.\Documentation\thirdpartylicense.pdf - passed
.\Documentation\RSAValidationManagerv30README.pdf - passed
.\Documentation\RSAValidationManagerInstallationGuide.pdf - passed
.\Documentation\RSAValidationManagerGettingStartedGuide.pdf - passed
.\Documentation\RSAValidationManagerCommandLineReferenceManual.pdf - passed
.\Documentation\RSAValidationManagerAdministratorsGuide.pdf - passed
.\Documentation - directory passed
. - directory passed


File list verification complete.

      14 files scanned in 7 directories.
      14 files passed
       0 files failed.
       0 directories failed verification.
       0 files were found that were not in the media signature file.
       0 files were not found that are in directories scanned in the media si
ture file.
```

Validating the Certificate and the Utility

Before you validate the verification certificate and the media verification utility, obtain the following from RSA:

- Verification certificate
- MD5 code of the media verification utility for your platform (Windows, Solaris, or Linux)

To obtain the verification certificate and the MD5 code for your media verification utility, go to the RSA SecurCare Online web site and follow the links to **Documentation > Guides & Manuals > RSA Validation Solution > RSA Validation Manager Media Validation Utilities**.

If you want a higher degree of assurance, you can contact your account manager to obtain the verification certificate and the MD5 code.

Validating the Verification Certificate

To validate the verification certificate:

1. At the command prompt, change the directory to the top directory of the product CD.
2. Run the media verification utility with the -signercert and -verifycert options. Do one of the following:
   - On Windows platforms, type:

     ```
     .\Windows\Util\mediaverify.exe -signercert certfile
     -verifycert yes .
     ```

   - On Solaris platforms, type:

     ```
     ./Solaris/Util/mediaverify -signercert certfile
     -verifycert yes .
     ```

   - On Linux platforms, type:

     ```
     ./RH_Linux/Util/mediaverify -signercert certfile
     -verifycert yes .
     ```

   where *certfile* is the name of the verification certificate that you obtained from RSA.
3. Verify that the media verification utility did not report any errors.

   If the validation is successful, you see the message, “Specified certificate matches the certificate found in the media signature file.”

   The media verification utility raises an error if the certificate specified by the -signercert option does not match the certificate used to sign the media. If this happens, consider your distribution to be suspect. Contact RSA Customer Support.

Validating the Media Verification Utility

You need an MD5 summing utility, such as OpenSSL, to validate the media verification utility.

To validate the media verification utility:

1. Do one of the following:
   - On Windows platforms, run your MD5 summing utility on the media verification utility, **mediaverify.exe**, in the **\Windows\Util** directory.
   - On Solaris platforms, run your MD5 summing utility on the media verification utility, **mediaverify**, in the **/Solaris/Util** directory.
   - On Linux platforms, run your MD5 summing utility on the media verification utility, **mediaverify**, in the **/RH_Linux/Util** directory.
2. Verify that the MD5 sum you calculate matches the one you obtained from RSA.

   If they do not match, consider your distribution to be suspect. Contact RSA Customer Support.

Glossary

Abstract Syntax Notation One (ASN.1)

An International Standards Organization (ISO) standard notation for defining the syntax of information data. It defines a number of simple data types and specifies a notation for referencing these types and for specifying the values of these types.

Administrator

A person, possibly with an end-entity certificate, who has access to the administration interface of Validation Manager. Administrator tasks may include installing RSA Validation Manager, setting up the OCSP Signers, status sources, and CAs for which Validation Manager serves status.

ARL

See **Authority Revocation List**.

ASN.1

See **Abstract Syntax Notation One**.

Audit Log

A tamper resistant log Validation Manager uses to record operational and configuration changing events.

Authentication

A process by which people or applications who receive a certificate can verify the identity of the certificate owner and the validity of the certificate. Certificates identify the author of a message or entity, such as a web server or client.

Authority Revocation List (ARL)

A list of CA certificates that a CA has revoked or suspended. You can use ARLs to check the status of CA certificates offline.

Backend

A collection of functions within the OCSP server that are invoked when an OCSP request is made.

Base64

See **Privacy Enhanced Mail (PEM) Format**.

CA

See **Certificate Authority**.

CA Certificate

A certificate that identifies a CA. When a CA issues a certificate to a client, a server, or other entity, the CA private key signs the certificate. You can verify the signature using the public key in the CA certificate. See also **Root CA**.

Cache Lifetime

A period of time during which Validation Manager can reuse a previously generated and signed response.

CA Purposes

CA purposes define the use of the CA within Validation Manager. For example, if the purpose of a CA is to provide certificate status, Validation Manager processes status requests for certificates issued by that CA. By default, Validation Manager assigns the following purposes to a CA: provide certificate status, verify OCSP clients, and verify remote secure servers.

Certificate

Certificates verify the identity of an individual, organization, web server, or hardware device. They also ensure non-repudiation in business transactions, as well as enable confidentiality through the use of public-key encryption. PKI uses three main kinds of certificates: CA certificates, server certificates (also referred to as SSL certificates), and end-entity certificates.

Certificate Authority (CA)

An entity that issues and manages certificates within a PKI. You create and manage CAs using a CA software application, such as RSA Certificate Manager.

Certificate Extension

See **X.509 v3 Certificate Extension**.

Certificate Policy (CP)

A policy that explains the conditions and limitations of use for a digital certificate.

Certificate Revocation List (CRL)

A list of revoked and suspended certificates (CA or end-entity) for a particular CA. You can use CRLs to check the status of certificates offline. See also **Complete CRL** and **Delta CRL**.

Certification Practice Statement (CPS)

A statement of an organization's security policies for the issuance and management of certificates.

Client Certificate

See **End-Entity Certificate**.

Clustering

The use of multiple Validation Manager installations and separate machines to form what is a highly available, fault tolerant system. To an OCSP client, the cluster appears to be one system. You can configure Validation Manager to support clustering with the addition of a load balancer.

Within the cluster, there is one primary installation, where Validation Manager is administered, and any number of secondary installations. Both primary and secondary installations can receive OCSP requests from the load balancer.

Complete CRL

A list that contains the serial numbers of certificates that a CA has revoked or suspended.

CRL

See **Certificate Revocation List**.

Cryptographic Provider

The library Validation Manager uses for private-key cryptographic operations (such as key pair generation and digital signatures). The method is either software-based or hardware-based (using nCipher).

Delta CRL

A list that contains the serial numbers of certificates that a CA has suspended, reinstated, or revoked since the last complete CRL.

Digital Signature Algorithm (DSA)

A digital signature algorithm used in the Digital Signature Standard (DSS) created by the U.S. government. For more information, see the standard designation **FIPS 186-2+ChangeNotice** at http://csrc.nist.gov/publications/fips/fips186-2/fips186-2-change1.pdf.

Distinguished Encoding Rules (DER)

An ASN.1 encoding standard used for signature calculation for end-entity certificates and revocation lists (that is, CRLs, delta CRLs and ARLs). Also known as **Binary**.

Distinguished Name (DN)

The combination of attributes in a certificate forms the certificate DN. The following attributes are the most commonly used attributes:

- Common Name
- User ID
- E-mail Address
- Organizational Unit
- Organization
- Locality
- State or Province
- Country
- Domain Component

To avoid potential problems, all CAs in the PKI, including trusted CAs, must have a unique DN.

End-Entity Certificate

A certificate issued to an entity that cannot itself issue certificates (that is, the entity is not a CA). Because the entity that requests such a certificate is sometimes referred to as the client, end-entity certificates are sometimes called client certificates.

End User (or End-Entity)

An individual, group, or organization that either requests or holds an end-entity certificate. An end user can also be an individual who requests an end-entity certificate for a hardware device (such as a router), a server, a software application, or a piece of code. An end user that requests a certificate is sometimes called a requestor. An end user that is issued a certificate is sometimes called a certificate owner, certificate subject, or end-entity. An end user that relies upon someone else's certificate to verify that person's identity is sometimes called an end user, certificate user, or relying party.

Enterprise

An organization that uses computers and applications. In general use, this term applies to businesses or organizations that operate on a large scale. These organization's applications are often referred to as enterprise applications.

Entity

A person, organization, or device (such as a router). In a PKI, an entity is anyone or anything you can issue a certificate to.

Expired Status Data

The freshness of a revocation list or status value in a status source in Validation Manager. A list or status value is expired once the refresh time plus the grace period elapse.

Extension

See **X.509 v3 Certificate Extension**.

FIPS 140-1 Level 2 & 3
FIPS 140-2 Level 2 & 3

A standard developed by the National Institute of Standards and Technology (NIST) for implementation of cryptographic modules. Level 3 provides greater security than Level 2.

Firewall

A system designed to prevent unauthorized access to or from a private network.

Fresh Status Data

The freshness of a revocation list or status value. A list or status value is fresh if the refresh time of its status source has not elapsed. For example, if the refresh time for a status source in Validation Manager to retrieve a new list or status value has not arrived, the list or status value within the Validation Manager database for that status source is considered fresh.

Forwarding

An OCSP client request triggers Validation Manager to send a second OCSP request to a remote OCSP server and use the remote server's response to construct its own response.

Fully Qualified Domain Name (FQDN)

The full name of a system, consisting of its local host name and its domain name. For example, "venera" is a host name and "venera.isi.edu" is the FQDN.

Grace Period

A period of time during which Validation Manager can reuse a stale status value, but must also attempt to obtain a newer status value. For example, when a remote OCSP response is in its grace period, and Validation Manager cannot fetch a new response for the same certificate, Validation Manager uses the status value from the previous OCSP response. The grace period specifies how long after the refresh time that the previous status value is valid. A status value expires once its refresh time and grace period elapse.

Hardware Security Module (HSM)

The module that performs cryptographic functions and stores cryptographic keys in a secure fashion.

Hypertext Transfer Protocol (HTTP)

A set of rules for exchanging files (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.

Web browsers are HTTP clients that send requests to server machines. Users enter page requests by either typing a URL or clicking a hypertext link. The browser builds an HTTP request for the user and sends it to the Internet Protocol (IP) address indicated in the URL. The HTTP daemon in the destination server receives the request and, after any necessary processing, returns the requested page. See also **HTTPS**.

HTTPS

HTTP over an SSL/TLS connection.

Identity Certificate

A certificate that links a public key value to a real-world entity such as a person, a computer, or a web server. Server certificates, CA certificates, and most end-entity certificates are all examples of identity certificates.

Key Pair

A public key and a private key associated with an entity that needs to authenticate its identity electronically or to sign or encrypt data. The public key is published, and the corresponding private key is kept secret. You can only decrypt data encrypted with the public key with the private key.

Key Size

The size (in bits) of the key pair used to sign status responses. A larger key size provides greater security. Validation Manager supports 1024, 2048, and 4096 bit keys.

Known CA

A CA that is known to the system. A CA becomes known to the system when you import the CA certificate into the system.

LDAP Directory

An LDAP-based directory is a database. You can search for and retrieve attribute-value pairs. You can configure directories to use (or support) authentication and access control protection. The schema of a directory describes the objects in the directory.

LDAPS

LDAP over SSL/TLS connection. See also **StartTLS**.

Lightweight Directory Access Protocol (LDAP)

The standard Internet protocol for accessing directory servers over a network. LDAP is a “lightweight” (smaller amount of overhead) version of Directory Access Protocol (DAP), which is part of X.500, a standard for directory services in a network. There are two currently supported versions, LDAP versions 2.0 and 3.0. See also **LDAPS** and **StartTLS**.

Locally Revoked Certificate

A certificate that is revoked within Validation Manager, but not revoked by a CA. A locally revoked certificate is not listed on a revocation list. Validation Manager returns a status of revoked for all enquires on the status of this certificate.

Load Balancer

A software or hardware product that routes incoming data to one of a number of possible resources or applications.

Nickname

A user-friendly character string that uniquely identifies a CA, OCSP signer, status source, or certificate recipient.

Nonces

Random numbers used in security protocols to prove that a message is part of a current message exchange.

Non-repudiation

A concept that prevents the author of a message from denying having created that message at a later date (that is, repudiation cannot occur). Digital signatures help ensure the non-repudiation of transactions.

OCSP

See **Online Certificate Status Protocol**.

OCSP Client

The entity that issues a certificate status request to an OCSP Responder. The OCSP client suspends acceptance of the certificate until the responder returns the certificate status.

OCSP Forwarding

One of two ways Validation Manager queries a remote OCSP server. During OCSP forwarding, a client request triggers Validation Manager to send a second OCSP request to a remote OCSP server, then use the remote server response to construct a second response to send to the client. Validation Manager can also use OCSP proxying to query a remote OCSP server.

OCSP Performance

The number of OCSP responses per second an OCSP server can process.

OCSP Proxying

One of two ways Validation Manager queries a remote OCSP server. During OCSP proxying, Validation Manager passes client requests unchanged to the remote OCSP server and returns the remote server response unchanged to the client. Validation Manager can also use OCSP forwarding to query a remote OCSP server.

OCSP Request

A client issues an OCSP request to obtain the status of a certificate. The client suspends acceptance of the certificate until it receives an OCSP response.

OCSP Responder

The OCSP Responder accepts certificate status requests from OCSP-enabled clients, looks up a certificate status, and responds with the certificate's current status.

OCSP Response

Validation Manager obtains the status of a certificate and returns an OCSP response to the client who issued the certificate status request.

OCSP Signer

An entity that signs OCSP responses.

Online Certificate Status Protocol (OCSP)

A protocol, defined in RFC 2560, that enables applications to check the status of a certificate every time the certificate is used. If you configure your PKI to use OCSP, CRLs are unnecessary for end users.

Online Validation

Online validation occurs when a CA can be queried directly about a certificate's validity every time the certificate is used.

Operator Card Set (OCS)

A card set within the nCipher security world that is used to generate, protect, and access the private keys created within it.

PKCS #7

The Cryptographic Message Syntax Standard. For more information on the standard, go to **www.rsasecurity.com/rsalabs/pkcs/pkcs-7/**.

PKCS #10

The Certification Request Syntax Standard. For more information on the standard, go to **www.rsasecurity.com/rsalabs/pkcs/pkcs-10/**.

PKCS #11

The Cryptographic Token Interface Standard. For more information on the standard, go to **www.rsasecurity.com/rsalabs/pkcs/pkcs-11/**.

PKI Performance

The number of revocation list per hour that an OCPS server can import.

PKIX (Public Key Infrastructure X.509)

The evolving Internet Engineering Task Force (IETF) standard for PKI using X.509 certificates. For more information on the standard, go to **www.ietf.org/html.charters/pkix-charter.html**.

Privacy Enhanced Mail (PEM) format

PEM was originally created to provide secure e-mail services on the Internet, but it became too unwieldy for widespread use. Now, "PEM format" usually refers to the Base64 encoding algorithm that was part of the PEM proposal.

PEM encoding is useful for presenting binary data in a text-readable form. (For example, to allow you to copy and paste data between applications.) Also known as **Base64**.

Private Key

The private part of a public-key key pair. With Validation Manager, private keys are generated on the OCSP server whenever an OCSP signer is created. Private keys must be securely stored to prevent unauthorized access and accidental deletion.

A digital signature involves encrypting a message digest with a private key and allows anyone with the corresponding public key to decrypt the message digest to be certain of who sent the message and that it has not been tampered with.

You can decrypt information encrypted with a public key with the corresponding private key.

Proxying

See **OCSP Proxying**.

Public Key

The public and widely distributed part of a public-key key pair. For example, a certificate contains information about the certificate subject, the certificate signer, and a public key value. In general, you can only decrypt information encrypted with a public key with the corresponding private key.

Public-Key Cryptography Standards (PKCS)

A set of standard protocols developed by RSA for making secure information exchange possible. The standards include RSA encryption, password-based encryption, and cryptographic message syntax. For more information on standards, go to **www.rsasecurity.com/rsalabs/pkcs/**.

Public Key Infrastructure (PKI)

A system for publishing, distributing, and managing the public key values used in public key cryptography. All PKIs involve issuing public key certificates to individuals, organizations, and other entities and verifying that these certificates are valid.

Refresh Time

The time after which Validation Manager attempts to retrieve a fresh status value. A status value is considered stale after its status source's refresh time elapses.

Response Caching

The process of reusing a previously generated and signed response.

Revocation

Revoking a certificate invalidates it and removes all of its privileges in the PKI. Revocation is necessary if the CA administrator wants to invalidate the certificate before it expires. Administrators revoke certificates by marking them as invalid in the Secure Directory. Users of the PKI are notified of the revoked status of a certificate during online validation or with revocation lists.

Revoking a CA invalidates the CA certificate and removes all PKI privileges of the CA. Revoke a CA only if you have organizational-based security concerns and only as a last resort.

Rivest-Shamir-Adleman (RSA)

A highly secure cryptography method created by the three founders of RSA: Professors Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman.

RSA uses a two-part key. The private key is kept by the owner; the public key is published. Data that is encrypted using the recipient's public key can only be decrypted by the recipient's private key, and vice-versa.

The RSA algorithm is computation intensive. Therefore, it is often used to create a digital envelope, which holds an RSA-encrypted symmetric key (often 3-DES or AES) and symmetric key-encrypted data. This method encrypts the secret symmetric key so that it can be transmitted over the network, but encrypts and decrypts the actual message using the much faster symmetric key algorithm.

The RSA algorithm is also used for authentication using digital signatures. In this case, the sender's private key is used for signing, and the sender's public key is used for verification. The RSA algorithm is also implemented in hardware. As RSA chips get faster, RSA encoding and decoding will add less overhead to the operation.

Root CA

A CA whose certificate is self-signed (that is, the issuer and the subject are the same). A root CA is at the top of a hierarchy.

Secure Hash Algorithm (SHA-1)

An algorithm developed by the U.S. National Institute of Standards & Technology (NIST). SHA-1 is used to create a cryptographic hash (or "fingerprint") of a message or data. SHA-1 is considered to be somewhat stronger than MD5. SHA-1 is defined in FIPS Publication 180-2, the Secure Hash Standard (SHS).

Secure Sockets Layer (SSL)

A protocol layer created by Netscape to manage the security of message transmissions in a network. Security is achieved through encryption. "Sockets" refers to the sockets method of passing data back and forth between client and server programs in a network or between program layers in the same computer.

Security World

A security world consists of at least one hardware module, a set of smart cards, and encrypted data stored on a computer.

Server Certificate

An end-entity certificate issued to a server. Servers present their certificates to web browsers so browsers can verify (authenticate) the identity of the server. Server certificates are sometimes called SSL or TLS certificates.

Signer

See **OCSP Signer**.

Signing

A process by which a digital signature is affixed to a file, document, or certificate as proof that it has not been tampered with and that the author is who claims to be the author.

Signer Certificate

A certificate, signed by a known CA, that the signer includes in signed status responses. To create a signer certificate, the CA signs a signer certificate request that you send to the CA. You can use the same request to obtain signer certificates from different CAs.

S/MIME

Microsoft and Netscape include S/MIME in the latest versions of their e-mail clients. Other vendors of message products also endorse S/MIME.

MIME itself, described in the IETF standard RFC 1521, defines the structure of an electronic message. S/MIME allows the message body to include encryption information and a digital certificate. S/MIME has extended the syntax provided in PKCS #7. For more information on the standard, go to **www.ietf.org/html.charters/smime-charter.html**.

SSL Client Authentication

The process whereby a server authenticates a client by verifying the end-entity certificate presented by the client during SSL operations.

SSL-LDAP

See **LDAPS**.

SSL Server Authentication

The process whereby a client application authenticates a server by verifying the certificate chain presented by the server during SSL operations, starting with a CA trusted by the client.

Stale Status Data

The freshness of a revocation list or status value. A list or status value is considered stale once the refresh time of its status source elapses. For example, if the refresh time for Validation Manager to retrieve a new list or status value has passed, the list or status value within the Validation Manager database is considered stale up until the time when the grace period elapses.

Status

The validity of a certificate: active, reinstated, revoked, or suspended.

Status Data Caching

The process of reusing previously obtained status data.

Status Source

A location and method for obtaining the status of certificates.

StartTLS

A method for opening a non-TLS connection, and then changing it into a TLS-protected connection. It is the standard way to use TLS for LDAP v3.

Suspension

The process of marking a certificate as temporary invalid. The end-user presenting the suspended certificate is denied access where the certificate previously allowed access. Reinstating a certificate returns all removed PKI privileges.

Suspending a CA certificate marks it as temporarily invalid and removes all of the CA's PKI privileges. Reinstating a CA certificate returns all removed PKI privileges.

System CA

The CA created during installation of Validation Manager to issue the server certificates.

Synchronization

The use of multiple Validation Manager installations to provide support for the synchronization of revocation data in a low bandwidth environment.

The synchronization server is a Validation Manager installation that has the most current revocation data. A synchronization client is a Validation Manager installation that requests revocation data from a synchronization server. The synchronization server and clients exchange certificates to authenticate each other.

Synchronization Performance

The number of status values per second that can be updated between two servers.

System Log

An operating system specific file that Validation Manager uses to record systemic events not related to regular operations or configuration changes.

Trace Log

A file containing information suitable for debugging purposes.

TLS Client Authentication

The process whereby a server authenticates a client by verifying the end-entity certificate presented by the client during TLS operations.

TLS Server Authentication

The process whereby a client application authenticates a server by verifying the certificate chain presented by the server during TLS operations, starting with a CA trusted by the client.

Transport Layer Security (TLS)

Internet protocol that provides privacy between server and client. TLS ensures that no third party may eavesdrop or tamper with any message. TLS is the successor to SSL; however, they are not interoperable.

UTF-8 Encoding

An ASCII compatible multibyte Unicode and UCS encoding, used by current browsers, Java and Plan 9.

Validation

The process of verifying that a certificate is valid. Validation can occur online or through the use of revocation lists.

Validation Manager

A server that accepts requests from clients to check the validity of certificates. Validation Manager supports the Online Certificate Status Protocol (OCSP).

Validation Manager Installation

An instance of Validation Manager. This may comprise a single machine hosting single instances of the various Validation Manager servers, or a farm of servers residing behind a Network Address Translator machine such as a load balancer. The servers within a Validation Manager installation are generally under a single administrative domain.

Validity

Whether a certificate is valid or invalid. A certificate is valid if it has not expired and a CA has not suspended or revoked it.

Web Server

An Apache-based server that is the primary interface to Validation Manager.

X.509

An International Standards Organization (ISO) standard that describes a basic electronic format for digital certificates.

X.509 v3 Certificate Extension

Certificate extensions, including extensions for PKIX, SET, and SSL. The RSA Validation Solution supports X.509 v3 that conform to version 3 of the X.509 standard and specify additional constraints or capabilities on the certificate subject.

Acronyms

API	application programming interface
ARL	authority revocation list
ASN.1	Abstract Syntax Notation One
CA	certificate authority
CP	certificate policy
CPS	certification practice statement
CRL	certificate revocation list
DER	Distinguished Encoding Rules
FQDN	fully qualified domain name
GUI	graphical user interface
HSM	hardware security module
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol (over an SSL connection)
I18N	Internationalization
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IPSec	IP Security Protocol
ISO	International Standards Organization
ITU/CCITT	International Telecommunication Union
LDAP	Lightweight Directory Access Protocol
LDAPS	Lightweight Directory Access Protocol (over an SSL/TLS connection)
MD5	Message Digest 5

MSIE	Microsoft Internet Explorer
OCSP	Online Certificate Status Protocol
PEM	Privacy Enhanced Mail format
PIN	personal identification number
PKCS	Public-Key Cryptography Standards
PKI	public key infrastructure
PKIX	Public Key Infrastructure (X.509)
RAM	random access memory
RSA	Rivest-Shamir-Adleman
S/MIME	Secure Multi-Purpose Internet Mail Extensions
SHA-1	Secure Hash Algorithm
SSL	Secure Sockets Layer
SSL-LDAP	Lightweight Directory Access Protocol over a Secure Sockets Layer connection
TLS	Transport Layer Security
UCS	Universal Character Set (the superset of all other character sets)
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UTF-8	UCS Transformation Format
VPN	virtual private network
XML	Extensible Markup Language

Index