Solutions Enabler

Version 8.3

Installation and Configuration Guide

REV 03

EMC²

Published November 2016

EMC Corporation
Hopkinton, Massachusetts 01748-9103
1-508-435-1000 In North America 1-866-464-7381
www.EMC.com

CONTENTS

FIGURES

TABLES

Preface

As part of an effort to improve its product lines, EMC periodically releases revisions of its software and hardware. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact your EMC representative if a product does not function properly or does not function as described in this document.

Note

This document was accurate at publication time. New versions of this document might be released on EMC Online Support (https://support.emc.com). Check to ensure that you are using the latest version of this document.

Purpose

This document describes how to install and configure EMC® Solutions Enabler software.

Related documentation

The following documents provide additional information about Solutions Enabler:

EMC Solutions Enabler, VSS Provider, and SMI-S Provider Release Notes

Describes new features and any known limitations.

EMC Solutions Enabler Installation and Configuration Guide

Provides host-specific installation instructions.

EMC Solutions Enabler CLI Command Reference

Documents the SYMCLI commands, daemons, error codes and option file parameters provided with the Solutions Enabler man pages.

EMC Solutions Enabler Array Controls and Management CLI User Guide

Describes how to configure array control, management, and migration operations using SYMCLI commands.

EMC Solutions Enabler SRDF Family CLI User Guide

Describes how to configure and manage SRDF environments using SYMCLI commands.

EMC Solutions Enabler TimeFinder SnapVX CLI User Guide

Describes how to configure and manage TimeFinder SnapVX environments using SYMCLI commands.

EMC Solutions Enabler SRM CLI User Guide

Provides Storage Resource Management (SRM) information related to various data objects and data handling facilities.

EMC SRDF/Metro vWitness Configuration Guide

Describes how to install, configure and manage SRDF/Metro using vWitness.

VMAX Management Software Events and Alerts Guide

Documents the SYMAPI daemon messages, asynchronous errors and message events, and SYMCLI return codes.

The following provide additional information:

EMC VMAX3 Family Product Guide for VMAX 100K, VMAX 200K, VMAX 400K with HYPERMAX OS

Provides product information regarding the purchase of a VMAX3 Family 100K, 200K, 400K.

EMC VMAX3 Family Site Planning Guide for VMAX 100K, VMAX 200K, VMAX 400K with HYPERMAX OS

Provides planning information regarding the purchase and installation of a VMAX3 Family 100K, 200K, 400K.

EMC VMAX All Flash and VMAX3 Family Security Configuration Guide

Describes how to securely deploy a VMAX3 Family (100K, 200K, 400K) or VMAX All Flash (250F, 450F, 850F) array with HYPERMAX OS.

EMC VMAX All Flash Product Guide for VMAX 250F, 450F, 850F with HYPERMAX OS

Provides product information regarding the purchase of a VMAX 250F, 450F, 850F with HYPERMAX OS.

EMC VMAX All Flash Site Planning Guide for VMAX 250F, 450F, 850F with HYPERMAX OS

Provides planning information regarding the purchase and installation of a VMAX 250F, 450F, 850F with HYPERMAX OS.

EMC VMAX All Flash and VMAX3 Family Security Configuration Guide

Describes how to securely deploy a VMAX3 Family (100K, 200K, 400K) or VMAX All Flash (250F, 450F, 850F) array with HYPERMAX OS.

EMC VMAX Family Viewer

Illustrates system hardware, incrementally scalable system configurations, and available host connectivity offered for VMAX arrays.

E-Lab™ Interoperability Navigator (ELN)

Provides a web-based interoperability and solution search portal. You can find the ELN at https://elabnavigator.EMC.com.

SolVe Desktop

Provides links to documentation, procedures for common tasks, and connectivity information for 2-site and 3-site SRDF configurations. To download the SolVe Desktop tool, go to EMC Online Support at https://support.EMC.com and search for SolVe Desktop. Download the SolVe Desktop and load the *VMAX All Flash, VMAX3 Family, VMAX, and DMX* procedure generator.

Note

You need to authenticate (authorize) your SolVe Desktop. After it is installed, familiarize yourself with the information under Help tab.

Conventions used in this document

EMC uses the following conventions for special notices:

⚠ CAUTION

CAUTION, used with the safety alert symbol, indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.

Note

A note presents information that is important, but not hazard-related.

Note

An important notice contains information essential to software or hardware operation.

Typographical conventions

EMC uses the following type style conventions in this document:

Normal	Used in running (nonprocedural) text for: • Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus • Names of resources, attributes, pools, Boolean expressions, buttons, DQL statements, keywords, clauses, environment variables, functions, and utilities • URLs, pathnames, filenames, directory names, computer names, links, groups, service keys, file systems, and notifications
Bold	Used in running (nonprocedural) text for names of commands, daemons, options, programs, processes, services, applications, utilities, kernels, notifications, system calls, and man pages Used in procedures for: • Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus

	• What the user specifically selects, clicks, presses, or types
Italic	Used in all text (including procedures) for: • Full titles of publications referenced in text • Emphasis, for example, a new term • Variables
`Courier`	Used for: • System output, such as an error message or script • URLs, complete paths, filenames, prompts, and syntax when shown outside of running text
`Courier bold`	Used for specific user input, such as commands
`Courier italic`	Used in procedures for: • Variables on the command line • User input variables
< >	Angle brackets enclose parameter or variable values supplied by the user
[ ]	Square brackets enclose optional values
|	Vertical bar indicates alternate selections — the bar means “or”
{ }	Braces enclose content that the user must specify, such as x or y or z
...	Ellipses indicate nonessential information omitted from the example

Where to get help

EMC support, product, and licensing information can be obtained on EMC Online Support, as described next.

Note

To open a service request through EMC Online Support, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or to answer any questions about your account.

Product information

For documentation, release notes, software updates, or for information about EMC products, licensing, and service, go to EMC Online Support (registration required) at:

https://support.EMC.com

Technical support

EMC offers a variety of support options.

Support by Product

EMC offers consolidated, product-specific information on the Web at: https://support.EMC.com/products

The Support by Product web pages offer quick links to Documentation, White Papers, Advisories (such as frequently used Knowledgebase articles), and Downloads, as well as more dynamic content, such as presentations, discussion, relevant Customer Support Forum entries, and a link to EMC Live Chat.

EMC Live Chat

Open a Chat or instant message session with an EMC Support Engineer.

eLicensing support

To activate your entitlements and obtain your license files, visit the Service Center on https://support.EMC.com, as directed on your License Authorization Code (LAC) letter emailed to you.

For help with missing or incorrect entitlements after activation (that is, expected functionality remains unavailable because it is not licensed), contact your EMC Account Representative or Authorized Reseller.

For help with any errors applying license files through Solutions Enabler, contact the EMC Customer Support Center.

If you are missing a LAC letter, or require further instructions on activating your licenses through the Online Support site, contact EMC's worldwide Licensing team at mailto:licensing@emc.com or call:

- North America, Latin America, APJK, Australia, New Zealand: SVC4EMC (800-782-4362) and follow the voice prompts.
- EMEA: +353 (0) 21 4879862 and follow the voice prompts.

Your comments

Your suggestions help us improve the accuracy, organization, and overall quality of the documentation. Send your comments and feedback to: VMAXContentFeedback@emc.com

Revision history

Provides a description of document changes.

Table 1 Revision history

Revision	Description and/or change
1	This is the initial revision of the Solutions Enabler V8.3 Installation Guide.
2	Removed references to HP Alpha hardware platform.
3	Added information on user credentials (UID(0)) and updated SEMAGENT references to storsrvd.

CHAPTER 1

Installation prerequisites

This chapter explains the tasks that you should perform before installing Solutions Enabler.

Introduction

An EMC Solutions Enabler install provides your host with SYMAPI, CLARAPI, and STORAPI shared libraries for use by Solutions Enabler applications, and the Symmetrix Command Line Interface (SYMCLI) for use by storage administrators and systems engineers.

SYMCLI is a specialized library of UNIX-formatted commands that can be invoked one at a time. It supports single command line entries and scripts to map and perform control operations on devices and data objects toward the management of your storage complex. It also monitors device configuration and status of devices that make up the storage environment. The target storage environments are typically VMAX arrays.

Before you begin

Before you begin to install Solutions Enabler, be sure to complete the tasks listed in this section.

General tasks

The following tasks apply to all supported platforms:

1. Obtain the software. Solutions Enabler is distributed as a platform-specific file download from the EMC Online Support at https://support.EMC.com
2. Review the interoperability information in the E-Lab™ Interoperability Navigator which can be reached at http://elabnavigator.EMC.com
3. https://support.emc.com Review the *EMC Solutions Enabler v8.3 Release Notes*.
4. If you are upgrading from a previous version, verify that all application processes that use the Solutions Enabler libraries and binaries are stopped. Stopping the application processes on page 112 provides instructions.
5. If you are upgrading from a previous version, create copies of the host database and configuration directories. These copies will be useful should you want to roll back to the previous version of Solutions Enabler. The location of these directories vary according to the operating system. Solutions Enabler Directories on page 263 provides more information.
6. EMC recommends that you read the *EMC VMAX Family Security Configuration Guide* and apply the settings after installation.

UNIX-specific tasks

The following task is specific to UNIX environments:

- AIX does not allow changes to the destination path during installation. All binaries and libraries are installed under `/opt/emc`.
  If there is insufficient disk space under `/opt`, create a soft link to `/opt/emc/` as shown below and then run the installer:

```
ln -s NewInstallationDir /opt/emc
```

The root user must have write permission on the *NewInstallationDir*.

Windows-specific tasks

Before starting the installation process, all Windows applications should be closed. This includes Windows Services and the Windows Event Viewer.

During the installation process, the **Service List** dialog will open so you can select the daemons to start. You can prepare for this by reading the section Setting up daemons for distributed application support on page 139.

z/OS-specific tasks

The following tasks are specific to z/OS Mainframe environments:

- Verify that you have a Windows host running a version of PKZIP or WinZip that supports 2.04 G compression.
  You will need the Windows host to FTP the installation files to the z/OS host.
- Install ResourcePak® Base.
  At start up, Solutions Enabler checks that a minimum version of EMC ResourcePak Base 7.6.0 is installed. However as ResourcePak versions go out of support or array requirements change, you must install the version of EMC ResourcePak Base required to support all host visible local or remote arrays.

  Note

  To work with VMAX3 arrays running HYPERMAX OS 5977 when using Solutions Enabler installed on z/OS, you need ResourcePak Base version 8.0.0.

  If you have already installed ResourcePak Base Version 7.6.0 or higher as part of another product installation, you do not need to re-install it. However, you should ensure that all recommended maintenance is applied.
- Choose an installation/configuration user account.
  To run the installation jobs, you must choose a TSO account in your system that has an OMVS segment defined in the security database. Since Solutions Enabler runs with the IBM Language Environment option POSIX(ON), the software requires that you either have a base OMVS segment defined or have access to an installation default profile. Before running any Solutions Enabler jobs, ensure that you have a correctly defined the OMVS segment.

  Note

  The installation jobs OMVS segment must be defined with UID(0).

  You should use this user's high-level qualifier when uploading the Solutions Enabler distribution file from the installation to the host.

  For more information on defining OMVS segments, see the IBM publication *z/OS Security Server RACF Security Administrators' Guide.*
- Gather the following customization information:
  - Solutions Enabler dataset name prefix
    Choose the prefix for all the product data sets to be allocated for the installation. The prefix includes the high-level qualifier and all secondary qualifiers except the last. For example, if you choose the default EMC.SSEM830 as the prefix, you will allocate EMC.SSEM830.LOADLIB, EMC.SSEM830.PARMLIB, and so on.

Note

This should be the same prefix as the one you choose when you upload the distribution file from the installation CD.

- SMP/E dataset name prefix
  Identify the prefix for the SMP/E datasets of the environment into which you have installed or will install the ResourcePak Base (EMCSCF). The default value is `EMC.SMPE`, which is the default for the ResourcePak Base product.
- SCF subsystem ID
  The EMCSCF server address space uses a z/OS subsystem identifier (SSID) to make itself known to applications that use its services. Solutions Enabler must have the same SCF SSID as the ResourcePak Base started task that you require it to use. The default is `EMC`.
- SCF linklib prefix
  Identify the prefix for the product datasets into which you have installed or will install the ResourcePak Base (EMCSCF) version 7.6.0 or higher. The default value is `EMC.SSCF760`, which is the default for the ResourcePak Base product, version 7.6.0. The EMCSCF Linklib will be added to the STEPLIB DD statement of the Solutions Enabler execution JCL.
- Disk unit name and volume serial
  Choose the unit name and a corresponding disk volume serial where you will install the Solutions Enabler product datasets. The default for unit name is SYSDA; there is no default for the volume serial.
- SYMAPI base directory
  Specify a Unix System Services directory under which SYMAPI runtime sub directories will be created.

  By default, the SYMAPI base directory is `/var/symapi`. However, during the execution of the Solutions Enabler SEMJCL installation procedure, you can change the default to any directory you want, provided that the security settings for the userids that run the Solutions Enabler jobs have read/write/execute permissions for the entire SYMAPI base directory tree.
- SYMAPI base directory space requirements
  The space requirements for the SYMAPI base directory vary according to the activities requested by clients (such as EMC Unisphere for VMAX) of the Solutions Enabler tasks. In addition, the logging options (type, detail, retention period) you select will also affect the space requirements for the SYMAPI base directory. In most cases, 50 to 100 MB should be sufficient.

  If you intend to configure the server to use `SYMAPI_LE_DUMP_LOGDIR`, you should consider providing additional space. For more information on `SYMAPI_LE_DUMP_LOGDIR`, refer to the *VMAX Management Software Events and Alerts Guide*.
- Time zone
  The time stamp on messages written by Solutions Enabler to its internal logs will use the Portable Operating System Interface (POSIX) default—normally Coordinated Universal Time (UTC). If you prefer a local time stamp, you will need to provide a POSIX-compliant time zone value.

  Configuring for local time zone on page 196 provides more information.

- Define the UNIX system services requirements:
  The following requirements apply to the userid of the installer which is the userid assigned to the started tasks or batch jobs used to run Solutions Enabler tasks

such as the SYMAPI server, storapid and event daemons. All userids running Solutions Enabler tasks must have an OMVS segment and full read/write/execute permissions to the SYMAPI base directory (by default `/var/symapi`) and all the sub-directories.

Note

The installation jobs OMVS segment must be defined with UID(0).

Note

Throughout the rest of this manual, this directory will be referred to as the *symapi_installation_directory*.

- Define the OMVS segment requirement
  When you are configuring Solutions Enabler JCL and your system to execute the SYMAPI server, you may need to add definitions to your local security system.

  If you are using IBM RACF, you may see message ICH408I when the server initializes. If you do, you must define an OMVS segment for the user or users who will run the server job. The following sample message assumes the job name and step name of the server are storsrvd:

  ```
  *ICH408I JOB(storsrvd) STEP(storsrvd) CL(process) OMVS
  SEGMENT NOT DEFINED
  ```

  If you are running the server as a started task, the user identity associated with the STC must have an OMVS segment defined. This is also true for the userid assigned to the batch job running the server (if you choose to run it that way).

  Note

  For information on defining an OMVS segment for each user, refer to the IBM publication *z/OS Security Server RACF Security Administrator's Guide*.

  In addition, the userids must have full read/write permissions for the entire directory tree (specified during the install) of the *symapi_installation_directory*. The Solutions Enabler daemons must all run with UID(0), because they need to delete/create/modify files or directories in the installation directory.

  If these permissions are not granted to the installer or the SYMAPI tasks, then various security error messages may be issued during the install or server setup.

  For example:

  ```
  ICH408I USER(user) Group(group) Name(username) 035
  035 /var/symapi CL(DIRACC )
  FID(01C8C6E2F0F0F200010D0000000000003)
  035 INSUFFICIENT AUTHORITY TO MKDIR
  035 ACCESS INTENT(-W-) ACCESS ALLOWED(OTHER R-X)
  035 EFFECTIVE UID(0000888888) EFFECTIVE GID(0000000900)
  ```

Linux on System z-specific tasks

The following tasks are specific to Linux for IBM System z environments:

Note

Once you have completed the tasks in this section, continue with the UNIX installation procedure in Installation prerequisites on page 19.

- Verify that you have a supported version of Linux for System z.
- Verify that the installer is using root during both pre -and- post installation phases.
- If Linux on System z is running as a guest under IBM's z/VM:
  Verify that all VMAX CKD devices are defined as z/VM unsupported DASD and attached to the Linux guest. The devices must be defined to z/VM (by way of `SET RDEV`) as:

  ```
  TYpe UNSUPported DEVCLass DASD DPS Yes RESERVE_RELease Yes
  ```

  For example:

  ```
  Set RDEVice 1300 TYpe UNSUPported DEVClass DASD DPS Yes
  RESERVE_RELease Yes
  ```

  By default, these devices will all function as gatekeepers. However, you can individually manage them by way of the gatekeeper select/avoid configuration files, as required.

  MVS formatted devices (regular MVS volumes) accessible by Linux on System z will appear in the Linux device tree. However, Solutions Enabler will not "discover" them, nor will it allow you to manage them by device name (such as, `/dev/dasdf`). In certain cases, you will be able to manage these devices by device number (for example, on the `symdg` command). Any gatekeepers though must be defined as unsupported DASD.

SYMAPI home directory

The example procedures in this document assume that the Solutions Enabler `<SYMAPI_HOME>` directory is located at:

- Windows: `c:\Program Files\EMC\SYMAPI...`
- UNIX: `/var/symapi/ ...`
- z/OS: `/var/symapi/ ...`

Pathnames presented in this document use a UNIX/specific format: forward slashes (/) instead of the backslashes (\) typically used on Windows platforms.

Note

By default, the location of `<SYMAPI_HOME>` is the same for both z/OS and UNIX.

Interoperability information

For information on previously released Solutions Enabler, VSS Provider, and SMI-S Provider features, refer to the corresponding release notes located on EMC Online Support at:

https://support.EMC.com

For detailed interoperability information, refer to E-Lab Interoperability Navigator at:

http://elabnavigator.EMC.com

Solutions Enabler

Support announcements

EMC lists the End of Service Life (EOSL) dates for the Solutions Enabler versions on EMC Online Support at https://support.EMC.com. On the EMC Online Support site, click **Support** > **Support By Product** in the main navigation bar. In the **Find a Product** box, type Solutions Enabler and click the arrow. The Solutions Enabler page will appear and the Service Life details are available on the left-hand side of the page.

Solutions Enabler target revisions and adoption rates

EMC has established product target codes to ensure stable and reliable environments. As a best practice, it is recommended that you operate at the recommended target code or above to benefit from the latest enhancements and fixes.

To view the latest recommendations, search for **Solutions Enabler Target Revisions and Adoption Rates** on EMC support.

Secure client/server root certificate replacement

The Solutions Enabler root certificate is used to generate and digitally sign subject certificates for use in SSL-secured client/server communications. The certificate is stored in the `symapisrv_trust.pem` file in the *<SYMAPI_HOME>*`/config/cert` directory. The file shipped with releases of Solutions Enabler prior to V7.5 expired in July, 2014.

An updated root certificate is included with Solutions Enabler V7.5 and higher with an expiration date of November, 2021.

Upon expiration of the older certificate, any client or server hosts which have not upgraded to Solutions Enabler V7.5 or higher will experience secure session negotiation failures. EMC recommends upgrading to V8.3 or higher as soon as possible to avoid outages due to the expiration of the older certificate.

For more information on certificate files, refer to the *VMAX Family Security Configuration Guide*.

Solutions Enabler compatibility with other products

If you are using products that rely on Solutions Enabler, please review the EMC Support Matrix at www.emc.com to verify that the product version you have is supported and fully compatible with this version of Solutions Enabler.

SMI-S Provider

Supported profiles

Table 2 on page 26 shows the SMI-S Provider supported profile groupings and their namespaces.

Table 2 Profile groupings with namespaces

Profile	Namespace
Array	`root/emc`
Server	`interop`

Table 3 on page 26 lists the SMI-S profiles supported by the Array Provider of the SMI-S Provider.

Table 3 SMI-S Provider profiles

Profile	SMI-S V1.5	SMI-S V1.6
Access Points	X	X
Automated Storage Tiering[a]		X
Automated Storage Tiering Policy[a]		X
Block Server Performance	X	X
Block Services	X	X
Block Storage Views	X	X
Disk Drive Lite	X	X
Disk Sparing[a]	X	X
Extent Composition	X	X
Fan	X	X
FC Initiator Ports	X	X
FC Target Ports	X	X
FCoE Target Ports		X
Group Masking and Mapping[b]	X	X
Health	X	X
Indication	X	X
Indicator LED	X	X
iSCSI Target Ports	X	X
Job Control	X	X
Location	X	X

Table 3 SMI-S Provider profiles (continued)

Profile	SMI-S V1.5	SMI-S V1.6
Multiple Computer System	X	X
Physical Package	X	X
Pools from Volumes[b]	X	X
Power Supply	X	X
Replication Services[b]	X	X
Software	X	X
Software Inventory		X
Storage Element Protection[b]	X	X
Storage Relocation[b]		X
Thin Provisioning[b]	X	X
Volume Composition[a]	X	X

a. Only supported for VMAX 10k/20k/40k arrays.
b. This profile is considered experimental and may change in future releases. As a result, backward compatibility cannot be guaranteed with the next release. Please contact EMC for permission to use this profile.

Supported products and specifications

Table 4 on page 27 lists the SMI-S schemas and specifications supported by SMI-S Provider V8.3.

Table 4 SMI-S Provider support for SMI-S

Supported schemas and specifications
Distributed Management Task Force Common Information Model (DMTF CIM) Schema V2.42.0
Storage Management Initiative Specification (SMI-S) V1.5.0, V1.6.0, V1.6.1
EMC ECOM V2.8.3.0.0.109[a]

a. This is included as part of the SMI-S Provider installation.

Rated metrics from VMAX3 arrays

SMI-S Provider V8.3 supports returning rated metrics from VMAX3 arrays. Rated metrics are obtained from a running instance of the Unisphere for VMAX application and provide the statistics in a calculated form per unit of time. The rates returned to SMI applications enable clients to consume the data directly without the need for any formulas or derivations.

Environment and system requirements

Solutions Enabler

Consider the following when working with Solutions Enabler V8.3.

Host systems and Enginuity support

Solutions Enabler runs on a wide range of 64-bit operating systems and works with certain VMAX array versions. For detailed interoperability information, refer to E-Lab Interoperability Navigator at:

http://elabnavigator.EMC.com

Disk space requirements

Table 5 on page 28 through Table 8 on page 31 list the disk space requirements for supported platforms.

Note

A value of 0 KBs means the component is not supported on that platform.

Table 5 Disk space requirements for AIX, Solaris Sparc UNIX

Install components (in KBs)	AIX	Solaris Sparc
Persistent data files	2853	852
SSL Certificate component	75	41
Thincore components	39158	11323
Base component (base storage, base mapping, and control storage libraries)	73681	38253
Command line tools (optional component)	91170	59379
Database mappings - SRM (optional component)	3390	659
SMI-S Provider (optional component)	0	0
Java Native Interface (optional component)	126576	52668
Symrecover including PERL 5.8 for Star (optional component)	19623	18541

Table 5 Disk space requirements for AIX, Solaris Sparc UNIX (continued)

Install components (in KBs)	AIX	Solaris Sparc
Enable 64-bit component install	125290	38376

Table 6 Disk space requirements for HP-UX ia64, and Linux ia64

Install components (in KBs)	HP-UX (ia64)	Linux (ia64)
Persistent data files	2853	992
SSL Certificate component	81	50
Thincore components	38649	24089
Base Component (Base Storage, Base Mapping, and Control Storage libraries	79982	48195
Command line tools (optional component)	174732	98761
Database mappings - SRM (optional component)	934	820
SMI-S Provider (optional component)[a]	0	0
Java Native Interface (optional component)	0	0
Symrecover including PERL 5.8 for Star (optional component)	24189	20416
Enable 64-bit component install	0	0

a. SMI-S is listed strictly for sizing purposes and is installed with Solutions Enabler as part of the SMI-S Provider kit.

Table 7 Disk space requirements for LinuxPPC, Linux on System z, and Celerral

Install components (in KBs)	Linux X64	Linux PPC	Linux on System z	Celerral
Persistent data files	978	984	977	979
SSL Certificate component	116	32	32	35
Thincore Components	13466	15804	13061	10783
Base component (Base Storage, Base Mapping, and Control Storage Libraries	115087	29244	29637	32767
Command line tools (optional component)	56623	59256	56764	56144
Database mappings - SRM (optional component)	758	92	6	0
SMI-S Provider (optional component)	94226	0	0	0
Java Native Interface (optional component)	51764	0	0	0
Symrecover including PERL 5.8 for Star (optional component)	18134	17850	1617	0
Enable 64-bit component install	0	0	0	0

Table 8 Disk space requirements for Windows

Install components (in MBs)	Windows (x64)
Base component (Base Storage, Base Mapping, and control storage libraries	110
SSL Certificate component	1
Command line tools (optional component)	15
Database Mappings - SRM (optional component)	1
Java Native Interface (optional component)	39
Symrecover including PERL 5.8 for Star (optional component)	20

Client/server interoperability

The server component of Solutions Enabler V8.3 SYMAPI is compatible with the client component of older SYMAPI versions from V7.6 and up. When planning to upgrade from V7.6 to V8.3, it is possible to do so in a staged fashion, upgrading the servers first, and then the clients. If access to V8.3 enhanced features is required only from the server systems, then there is no requirement to upgrade client systems. For clients to gain access to V8.3 enhanced features, they must be upgraded.

The client component of Solutions Enabler V8.3 SYMAPI is no longer compatible with older server components than V8.3.

Secured sessions using SSL are only available when both the client and server are running Solutions Enabler V7.6 or later on platforms that support secure communication.

Non-secured sessions between SSL-capable clients/servers and a remote peer on a non SSL-capable platform are possible as long as you configure the security level of the SSL-capable clients/servers to ANY. For more information, refer to Client or server installation on page 43 and the *EMC VMAX Family Security Configuration Guide.*

Security settings

Refer to the *EMC VMAX Family Security Configuration Guide* for information on how security settings work in Solutions Enabler and how to configure them.

VSS Provider

Windows Server 2008 Hyper-V

VSS Provider V8.3 supports 64-bit Windows Server 2008 and 2008 R2 Hyper-V server virtualization for VMAX arrays. Hyper-V is installed and managed as a role under Windows Server 2008 and Windows Server 2008 R2.

VSS Provider supports the following guest operating systems with Windows server 2008 R2 (x64) as a parent operating system:

- Windows 2008 x64
- Windows 2008 R2 x64
- Windows Server 2012

Windows Server 2012 Hyper-V

VSS Provider V8.3 supports 64-bit Windows Server 2012 and 2012 R2 Hyper-V server virtualization for VMAX arrays. Hyper-V is installed and managed as a role under Windows Server 2012 and 2012 R2.

VSS Provider supports the following guest operating systems with Windows server 2012 or Windows server 2012 R2 as a parent operating system:

- Windows 2008 R2 x64
- Windows 2012
- Windows 2012 R2

Configuring the Hyper-V environment

For configuration instructions, refer to the *Hyper-V Getting Started Guide* and *Virtualization with Hyper-V: FAQ* located in the Microsoft TechNet Library.

By default, SCSI commands are filtered in Hyper-V in Windows Server 2008 R2 and Windows Server 2012. To use Solutions Enabler on a guest partition, disable the SCSI command filtering, as recommended in the *Planning for Disks and Storage* article in the Microsoft TechNet Library.

For Windows Server 2008 R2, the following PowerShell script, executed from the parent partition, disables SCSI command filtering for each guest partition listed as an argument to the script. The settings are persistent, but will require a restart of the partition to take effect. The script is provided as an example and does not include validation or error-checking:

```
$Target = $args[0]
$VSManagementService = gwmi
MSVM_VirtualSystemManagementService -Namespace
"root\virtualization"
foreach ($Child in Get-WmiObject -Namespace
root\virtualization Msvm_ComputerSystem -Filter
"ElementName='$Target'")
{
$VMData = Get-WmiObject -Namespace
root\virtualization-Query "Associators of {$Child}
Where ResultClass=Msvm_VirtualSystemGlobalSettingData
AssocClass=Msvm_ElementSettingData"
$VMData.AllowFullSCSICommandSet=$true
$VSManagementService.ModifyVirtualSystem($Child,$VMDa
ta.PSBase.GetText(1))|
out-null}
```

For Windows Server 2008 R2, the following PowerShell script, executed from the parent partition, displays the current filtering status of each guest partition listed as arguments to the script. The script is provided as an example and does not include validation or error-checking:

```
$Target = $args[0]
foreach ($Child in Get-WmiObject -Namespace
```

```
root\virtualization
Msvm_ComputerSystem -Filter "ElementName='$Target'")
{
$VMData= Get-WmiObject -Namespace
root\virtualization-Query "Associators of {$Child}
Where ResultClass=Msvm_VirtualSystemGlobalSettingData
AssocClass=Msvm_ElementSettingData"
Write-host "VirtualMachine:" $VMData.ElementName
Write-Host "CurrentlyByPassingSCSIFiltering:"
$VMData.AllowFullSCSICommandSet}
```

For Windows Server 2012 R2, the following PowerShell script, executed from the parent partition, disables SCSI command filtering for each guest partition. The settings are persistent, but will require a restart of the partition to take effect. The script is provided as an example and does not include validation or error-checking:

```
$VSManagementService = gwmi Msvm_VirtualSystemManagementService -
namespace "root\virtualization\v2"

function disablefiltering{
foreach ($Child in Get-WmiObject -Namespace root\virtualization\v2
Msvm_ComputerSystem -Filter "ElementName='$Target'"){
$VMData = Get-WmiObject -Namespace root\virtualization\v2 -Query
"Associators of {$Child}
  Where ResultClass=Msvm_VirtualSystemSettingData"
$VMData.AllowFullSCSICommandSet=$true
$VSManagementService.ModifySystemSettings($VMData.PSBase.GetText(1))
 | Out-Null
queryfiltering
}
If ($Child){ Break }
Else{ write-host -back Red "Could not find Virtual Machine $Target
on this Server" }
}

function enablefiltering{
foreach ($Child in Get-WmiObject -Namespace root\virtualization\v2
Msvm_ComputerSystem -Filter "ElementName='$Target'"){
$VMData = Get-WmiObject -Namespace root\virtualization\v2 -Query
"Associators of {$Child}
  Where ResultClass=Msvm_VirtualSystemSettingData"
$VMData.AllowFullSCSICommandSet=$false
$VSManagementService.ModifySystemSettings($VMData.PSBase.GetText(1))
 | Out-Null
queryfiltering
}
If ($Child){ Break }
Else{ write-host -back Red "Could not find Virtual Machine $Target
on this Server" }
}

function queryfiltering{
foreach ($Child in Get-WmiObject -Namespace root\virtualization\v2
Msvm_ComputerSystem -Filter "ElementName='$Target'"){
$VMData = Get-WmiObject -Namespace root\virtualization\v2 -Query
"Associators of {$Child}
  Where ResultClass=Msvm_VirtualSystemSettingData"
  Write-host -back darkgreen "Virtual Machine:" $VMData.ElementName
  Write-Host -back darkgreen "Currently ByPassing SCSI Filtering:"
$VMData.AllowFullSCSICommandSet
}
If ($Child){ Break }
Else{ write-host -back Red "Could not find Virtual Machine $Target
on this Server" }
```

```
}

$Target = Read-Host 'Enter Virtual Machine Name'
$Action = Read-Host 'Enter Filtering Action (Disable, Enable,
Query)'

if ($Action -eq 'Disable'){ disablefiltering }
else
{
 if ($Action -eq 'Enable'){ enablefiltering }
else
 {
  if ($Action -eq 'Query'){ queryfiltering }
else { write-host -back Red 'Invalid Action Value: Value must be
"Disable", "Enable" or "Query."' }
}
}
```

Note

For more information, refer to *EMC Symmetrix with Microsoft Hyper-V Virtualization* available at: https://support.EMC.com.

Configuring child partition

To authorize Solutions Enabler access, use the SYMCLI `symcfg` command as shown in the following syntax example:

```
symcfg authorization add -host HostName -username UserName -
password PassWord -hyperv
```

Where:

- *HostName* — Hyper-V parent hostname/IP address
- *UserName* — Domain\username of parent Hyper-V server

  Note

  If the Hyper-V server is not under any domain, *HostName* should be appended for *Domain*, for example: *HostName**UserName*

- `PassWord` — Password of parent Hyper-V server

VMAX gatekeeper requirements

At least three unique gatekeeper devices must be assigned to each Hyper-V child partition, as a pass-through disk, to allow Solutions Enabler access from the child partition to the VMAX array.

Based on the number of applications running on a child partition, more gatekeepers may be required. Refer to the appropriate release notes, or installation guide for gatekeeper recommendations for other applications.

Note

For specific gatekeeper sizing recommendations for all VMAX configurations, refer to EMC Knowledgebase article EMC 255976.

Hyper-V connectivity support issues

Fibre Channel and iSCSI connectivity to the Hyper-V server is supported for VMAX arrays running HYPERMAX OS 5977 and Enginuity 5876.

VSS Provider V8.3 does not support snapshot creation using iSCSI connected devices on guest virtual machines hosted on the Hyper-V server though devices connected through Fibre Channel are supported on guest VMs.

Windows Server hotfix information

Ensure that all Microsoft Windows patches are up to date. The following Windows Server hotfix must be applied before installing and running VSS Provider.

For all Windows Server 2008 R2 editions listed in Table 9 on page 35, Microsoft hotfix #KB975688 is required. The fix can be downloaded from the knowledge base article.

Table 9 Microsoft Server 2008 R2 editions for hotfix

Windows editions
Windows Server 2008 Standard x64 Edition with SP1 or SP2
Windows Server 2008 Enterprise x64 Edition with SP1 or SP2
Windows Server 2008 R2 Standard x64 Edition with SP1
Windows Server 2008 R2 Enterprise x64 Edition with SP1

Solutions Enabler compatibility

VSS Provider V8.3 requires that Solutions Enabler V8.3 is installed. VMAX arrays managed using VSS Provider must be running HYPERMAX OS 5977 or Enginuity 5876.

Authorizing connectivity in Solutions Enabler

Components within your storage environment require authorization information to provide access for Solutions Enabler. The SYMCLI `symcfg authorization` command is used to supply this information.

VMware virtual servers

VSS Provider supports all the platforms listed in Table 9 on page 35 running as a virtual server on VMware ESX Server, for both Fibre Channel and iSCSI connectivity. The following versions of the VMware ESX Servers are supported:

- VMware ESX Server 4.0 (vSphere 4.0) (Update 1)
- VMware ESX Server 4.1 (Update 1)
- VMware ESXi server 4.1 (Update 1)
- VMware ESXi server 5.0 (Update 1)
- VMware ESXi server 5.1 (Update 1)
- VMware ESXi server 5.5
- VMware ESXi server 6.0

Refer to VMware vSphere and ESX documentation sets for detailed configuration instructions for ESX Server. You can find the most up-to-date VMware technical documentation on the VMware website.

VMware configuration guidelines for ESX virtual server

To configure an ESX virtual server to properly run the VSS Provider, follow these configuration steps:

Procedure

1. Install VMware tools on each virtual server where the VSS Provider is installed.
2. After creating your virtual machine, run the `vicfg.exe` utility to create an entry for the symcfg authorization database to configure communication with ESX Server.
3. For a virtual machine running on VMware ESX Server 4.0, configure the virtual machine with the fully qualified domain name (FQDN).

VMware configuration guidelines for ESXi virtual server

To configure an ESXi virtual server to properly run the VSS Provider, follow these configuration steps:

Procedure

1. Install VMware tools on each virtual sever where the VSS Provider is installed.
2. Use the SYMCLI `symcfg` command as shown in the following example:

```
symcfg authorization add -host HostName -username UserName -
password PassWord -namespace NameSpace -port Port -vmware
```

Where:

- *HostName* — ESXi server hostname/IP address
- *UserName* — username of ESXi server. Should be a root user.
- *PassWord* — password of ESXi server
- *NameSpace* — namespace which qualifies the VMware web service address
- *Port* — port at which the VMware web service is listening

Additional VMware virtual server support issues

Note the following support issues when running VSS Provider with VMware virtual servers:

- For VMAX arrays, the SPC-2 port flag must be set on all front-end ports to which the virtual server is connected.
- For VMAX arrays, the ACLX port flag must be enabled on the front-end directors.
- Fibre Channel connectivity to the ESX Server is supported. iSCSI connectivity is not supported for VMAX3 arrays running HYPERMAX OS 5977.
- For iSCSI support for VMAX 10K, 20K, 40K arrays running Enginuity 5876, the iSCSI initiator name on the ESX Server and virtual machine must be the same. Refer to your VMware documentation for enabling iSCSI on virtual machines.

- At least three unique gatekeeper devices must be assigned to each ESX/ESXi VM.

SMI-S Provider

VMAX gatekeeper requirements

When using the SMI-S Provider V8.3 to manage VMAX arrays, it is recommended that six gatekeepers be present for use by the provider.

GNU Compiler Collection (GCC) standard C++ library requirements

SMI-S Provider V8.3 requires the GNU Compiler Collection (GCC) standard C++ library `/usr/lib/libstdc++.so.6` for its dynamically linked C++ binaries. This generally comes with `libstdc++ rpm`, which is found in systems with GCC version 3.4.0 and higher, or systems with `libstdc++` version 3.4.0 and higher.

Before installing SMI-S Provider V8.3 in RedHat Enterprise Linux and SuSE systems, verify that `compat-libstdc++ rpm` is already installed, which provides the compatible C++ libraries.

For example, run the following commands to check for these compatible C++ libraries:

```
# rpm -qa | grep libstdc++
```

```
compat-libstdc++-33-3.2.3-47.3
libstdc++-3.4.5-2
libstdc++-devel-3.4.5-2
compat-libstdc++-296-2.96-132.7.3
libstdc++-4.4.7.3.el6.x86_64
libstdc++-4.4.7.3.el6.i686
```

```
# rpm -ql libstdc++-3.4.5-2
```

```
/usr/lib/libstdc++.so.6
/usr/lib/libstdc++.so.6.0.3
```

```
# rpm -ql libstdc++-4.4.7-3.el6.x86_64
```

```
/usr/lib64/libstdc++.so.6
/usr/lib64/libstdc++.so.6.0.13
```

```
# rpm -ql libstdc++-4.4.7-3.el6.i686
```

```
/usr/lib/libstdc++.so.6
/usr/lib/libstdc++.so.6.0.13
```

If you do not have the correct version installed, obtain and install it before proceeding with the SMI-S Provider installation.

Run the following command to install the library:

```
# rpm -ivh compat-libstdc++*.rpm
```

WBEM infrastructure

SMI-S Provider V8.3 utilizes an EMC-based WBEM (Web-Based Enterprise Management) infrastructure called EMC CIM Object Manager (ECOM). This WBEM infrastructure is used for both proxy and embedded environments across all EMC hardware and software platforms to ensure consistent implementation and experience across EMC products.

For detailed information about ECOM, see the *ECOM Deployment and Configuration Guide.*

z/OS-specific requirements

The following are the z/OS-specific requirements.

Note

The following Solutions Enabler features are not supported on z/OS: RDF daemon, SRM, and Star. For more information, refer to Table 22 on page 140.

Platform requirements

EMC Solutions Enabler for z/OS runs on all IBM supported releases of z/OS, and it requires a pre-existing SMP/E environment.

Some of the z/OS components that Solutions Enabler for z/OS uses are:

- Language Environment services.
- UNIX System Services socket support.
- TCP/IP protocol stack.

Note

Only IBM TCP/IP has been qualified by EMC. Support for other TCP/IP protocol stacks must be requested through the EMC Request for Price Quotation (RPQ) process.

There are no special requirements to enable IBM TCP/IP support.

z/OS-specific directory structure requirements

With the introduction of SSL-protected client/server sessions, the installation process looks for the installer's instructions about where to place the SYMAPI base directory. The base directory specifies a high-level location where the standard SYMAPI directory will reside. Since use of SSL was optional, the Unix System Services directories were not required to be created.

The SYMAPI directory structure is required on any host running Solutions Enabler V7.6 or higher. Configuration files must reside in the `config` directory under the base directory, and log files will be stored in the `log` directory.

Unix System Services file system requirements

The following are z/OS Unix System Services file system requirements:

Logging
The server, base, and event daemon write data to log files in the Unix System Services file system. Summary log data is written to `SYSPRINT DD` , but the comprehensive detail is written to Unix System Services files.

SYMAPI log file
Solutions Enabler writes all SYMAPI log data to a standard dated log file in the SYMAPI log directory.

Unix System Services file system options

The following Unix System Services file system options can be configured to meet your environment:

SYMAPI database
MVS datasets (via DD SYM$DB) are not supported. The Unix System Services file system will always be used to store the database.

Avoid, Gatekeeper Avoid and Select, and INQ files
Starting with release V7.6, Solutions Enabler does not read select or avoid files using JCL definitions. In other words, relevant DD statements (`SYM$AVD`, `SYM$GAVD`, `SYM$GSEL`, and/or `SYM$INQ`) are no longer supported in JCL. If they are present, any `SymInit` received will fail with an error message `SYMAPI_C_FILE_TYPE_NOT_SUPPORTED`.

DD statements such as `SYM$ENV` and `SCR$xxxx` are still valid.

For more information on the avoidance and selection files, refer to Avoidance and selection files on page 190.

Running z/OS as a guest

When running z/OS as a guest under the z/VM operating system, the TimeFinder and SRDF utilities require special consideration. Devices must be defined to z/VM `(SET RDEV)` as:

```
TYpe UNSUPported DEVCLass DASD DPS Yes RESERVE_RELease Yes
```

These devices must be attached to the z/OS guest.

Note

VM does not allow volumes defined as unsupported to be attached to SYSTEM, or used to IPL a virtual machine.

Virtual memory requirements

Solutions Enabler software always uses allocated memory above the 16 MB line. The actual region required depends on many factors such as the number of active tasks and connections, the number of managed VMAX arrays, and devices. It is not unusual for Solutions Enabler tasks (especially the server and base daemons) to consume many hundreds of megabytes of memory. If this is a possibility, consult with your system programmer to ensure that paging environments are adjusted accordingly.

EMC recommends specifying `REGION=0M` on the JOB card or EXEC card for the following jobs:

- `#10ECCIN`
- `#STORSRV` and any other JCL which uses `#STORSRV` as a model
- `#STORAPI` and any other JCL which uses `#STORAPI` as a model
- `#STOREVT` and any other JCL which uses `#STOREVT` as a model
- `#STORGNS` and any other JCL which uses `#STORGNS` as a model

These members are distributed with `REGION=0M` already specified on the EXEC cards. Your site may have SMF or JES exits or security rules established which restrict the use of `REGION=0M`. Check with your system programmer to verify that the submitting user has the authority to use `REGION=0M`.

Backward/forward compatibility for applications

Solutions Enabler V8.3 can only read databases previously written by Solutions Enabler V7.6 or higher. Database files earlier than V7.6 must be rebuilt. For details on rebuilding the SYMAPI and Base Daemon databases, see Knowledgebase article 000009813.

In client/server mode, Solutions Enabler V8.3 servers only support clients running Solutions Enabler V7.6 or higher.

Note

SYMAPI database access is not forward compatible because a SYMAPI library cannot access a database created by a newer version of a SYMAPI application. If, for example, the version of the local library becomes out of sync with the version of the local SYMAPI database (as a V7.5 SYMAPI library call from a SYMAPI client attempting to access a V8.3 database) it will return error: `SYMAPI_C_DB_FILE_TOO_NEW`. This restriction relates only to local databases. In client/server environments, accesses to a server database of a later version are automatically resolved by the SYMAPI, which performs all necessary translation of information between the client and the server.

Storage systems

This section identifies storage system array models, operating software versions, and configuration requirements for the supported VMAX arrays.

VNX or CLARiiON arrays

From Solutions Enabler V8.0, VNX and CLARiiON arrays are no longer supported.

SMI-S Provider array support

SMI-S Provider V8.3 supports the following VMAX storage families:

- VMAX3 Family (VMAX 100K, 200K and 400K)
- VMAX Family (VMAX 10K, 20K and 40K)

VSS Provider array support

VSS Provider supports VMAX Family arrays with Enginuity 5876 and VMAX3 Family arrays with HYPERMAX OS 5977.

Supported HYPERMAX OS
VMAX arrays managed using VSS Provider must be running Enginuity 5876 or HYPERMAX OS 5977.

Connectivity
Both Fibre Channel and iSCSI connectivity are supported for VMAX arrays running Enginuity 5876 and VMAX3 arrays running HYPERMAX OS 5977.

VMAX array configuration requirements
Configuration requirements for using VSS Provider with VMAX arrays are as follows:

- Director flags
  When using the VSS Provider with storage arrays, the following director flags must be enabled on all directors connecting to the VSS host:
  - VCM director flag (VCM_state) — Enables the Volume Logix software on the VMAX array so that the VSS Provider can perform device masking. If this flag is not enabled, then the VSS Provider fails to create and import snapshots, due to the lack of device masking capabilities.
  - SPC-2 (SPC2_Protocol_Version) director flag— Forces the VMAX array to report its device identifiers in a way that VSS recognizes. If this flag is not enabled, then the VSS service fails all snapshots before the VSS Provider is even called.
  - ACLX director flag — Must be enabled on the directors of VMAX arrays. This director flag enables the Auto-provisioning Groups software on the array so that the VSS Provider can perform device masking. If this flag is not enabled, then the VSS Provider fails to create and import snapshots, due to the lack of device masking capabilities.
- VMAX array masking view
  At least one masking view must be present before proceeding with any VSS Provider operations.
- TimeFinder Mirror
  VSS Provider requires a BCV to be paired with the source LUN. This requires performing a full Establish operation at some point. Multiple BCVs are supported for a given source LUN. Currently synchronized BCVs are used first, followed by the oldest split BCV (longest time since last split).

  TimeFinder Mirror is not supported when `EnforceDefaultToClone` is set to `True` in the registry.

  VSS Provider supports both Thin BCV (TDEV+BCV) and thick BCV device configurations on VMAX 10K, 20K, and 40K arrays running Enginuity 5876.
- TimeFinder Clone
  TimeFinder Clone is supported only through EMC Requestors, which require the VSS requestor to handle all configuration requirements when `EnforceDefaultToClone` is set to `False` in the registry. VSS expects the target clone to be in the Created or Recreated state when `RetainCloneSession` is set to `False` in the registry.
- TimeFinder VP Snap
  VSS Provider supports TimeFinder VP Snap only when the registry key `EnforceVPSnap` is set to `True`. With differential snapshots, VSS Provider looks first for a valid VP Snap replica. If a VP Snap session does not exist, the provider exits with a valid error message.
- Remote (SRDF ®) TimeFinder Mirror (Remote BCV)

VSS Provider supports an R1 to R2 - Remote BCV configuration. The SRDF link must be synchronous and in the Synchronized state. Beyond this point, the rules of local TimeFinder Mirror take over.

VSS does not provide a way to differentiate between local and remote snapshots. However, VSS Provider coordinates the two, and gives preference to local snapshots before remote snapshots. This means that if both local and remote BCVs are configured, the local BCV will be used in the snapshot. To force VSS Provider to use Remote BCVs, set the registry key "RemoteSnapshotsOnly" outlined in Remote snapshots on page 236.

Remote (SRDF) TimeFinder Mirror is not supported when `EnforceDefaultToClone` is set to `True` in the registry.

- Remote (SRDF) TimeFinder Clone (RClone, TDEV)
  Remote TimeFinder Clone is supported only through EMC requestors, which require the VSS requestor to handle all configuration requirements when `EnforceDefaultToClone` is set to `False` in the registry.
- Remote (SRDF) TimeFinder VP Snap
  Remote TimeFinder VP Snap is supported only when the registry key `EnforceVPSnap` is set to `True`.

Note

All of the above described VSS-supported TimeFinder Mirror and TimeFinder Clone operations support only Thin devices (TDEVs and TDEV-BCVs).

Supported replication technologies

Table 10 on page 42 lists the EMC replication technologies that are supported with VSS Provider.

Table 10 VSS Provider supported replication technologies

Array	Plex snapshot	Differential snapshot
VMAX arrays running Enginuity 5876	TimeFinder Mirror TimeFinder Clone Remote (over SRDF) TimeFinder Mirror Remote (over SRDF) TimeFinder Clone	TimeFinder VP Snap TimeFinder Snap Remote (over SRDF) TimeFinder VP Snap Remote (over SRDF) TimeFinder Snap
VMAX3 arrays running HYPERMAX OS 5977	SnapVX plex TimeFinder/Mirror [a] TimeFinder/Clone [b, c] Remote (over SRDF) SnapVX plex Remote (over SRDF) TimeFinder/Mirror [a] Remote (over SRDF) TimeFinder/Clone [b, c]	SnapVX differential Remote (over SRDF) SnapVX differential [b] TimeFinder VP Snap [d] Remote (over SRDF) TimerFinder VP Snap [d]

Table 10 VSS Provider supported replication technologies (continued)

a. Not supported when registry key EnforceDefaultToClone is set to TRUE.
b. Requires the use of EMC requestors NMM, RM, TFIM.
c. Does not require the use of EMC Requestors NMM,RM,TFIM when registry key EnforceDefaultToClone is set to TRUE.
d. Supported only when registry key EnforceVPSnap is set to True.

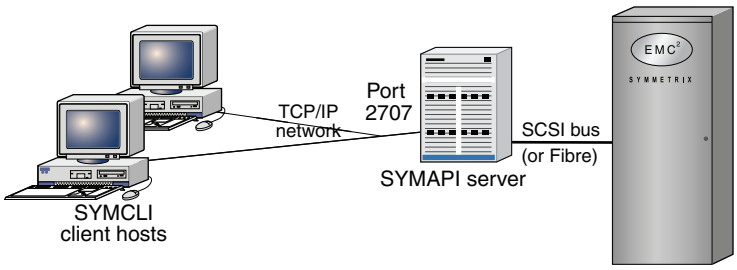
Client or server installation

If your computer is locally connected to a VMAX array, go to Installation on page 51.
If your computer is a client or the SYMAPI server, read the following sections.

Remote connection

You can run SYMCLI as a client to a remote SYMAPI server to manage a remotely-controlled VMAX array. The following diagram shows a VMAX array in the client/server system.

Figure 1 A VMAX array in the client/server system



Client/server IP communication

The SYMAPI client and server are both capable of negotiating sessions over the traditional Internet Protocol Version 4 (IPv4) and the newer Internet Protocol Version 6 (IPv6).

All hosts that use TCP/IP for communications use at least IPv4, a protocol well known to many applications. Newer versions of host operating systems will also support configuration of IPv6 local addresses, routing, and Domain Name Services as well. For the foreseeable future, many networks are likely to be running with dual protocol stacks activated, where communications will take place over IPv4 most of the time. Applications such as Solutions Enabler can also detect the presence of IPv6 configuration and use it whenever possible.

In UNIX, Linux, and Microsoft Windows Server environments, the SYMAPI server and client will interoperate with both IPv6 and IPv4 protocols on hosts that are configured to run both. The protocol selected by the server and the client depends on the exact configuration of the host, router, and DNS servers in your network, and on the settings in the Solutions Enabler network services configuration file.

Client/server security

Solutions Enabler uses Secure Socket Layer (SSL) protocol to enable secure communication in a client/server system. Using open source SSL (OpenSSL) technology, the client and server communicate over an authenticated, encrypted connection.

When a client attempts to connect to a server, the two machines exchange a handshake in which they both identify their security expectations and capabilities. If their security capabilities are the same, the two will negotiate the appropriate type of session (secure or non-secure). If their security capabilities are different, either the client or the server will reject the session.

The SYMAPI client and server are initially configured to communicate via secure sessions. You must modify this behavior if a platform in the environment does not support secure communications. The *EMC VMAX Family Security Configuration Guide* provides instructions on modifying this default behavior.

Table 11 on page 44 lists the host operating systems that support SSL.

Table 11 Host operating system support for SSL

Supported operating system	
AIX (64-bit)	
HP-UX (64-bit) HP-UX Itanium (64-bit)	
Linux Itanium (64-bit) Linux AMD (64-bit)	
Solaris (64-bit)	
Windows AMD (64-bit)	
z/OS	

Client/server system installation

The following information outlines procedures for installing Solutions Enabler in a client/server system:

Procedure

1. Install Solutions Enabler software in the machine designated as the client, according to the procedures in Installation on page 51.
2. Install the same Solutions Enabler software in the machine designated as the server, according to the procedures in Installation on page 51.
3. Edit the `netcnfg` file in the client machine to include the host name or IP address of the server. SYMCLI through a remote server on page 210 provides instructions.
4. Issue a `stordaemon start storsrvd` command on the server machine. SYMCLI through a remote server on page 210 provides instructions.
5. Set environment variables `SYMCLI_CONNECT` and `SYMCLI_CONNECT_TYPE` on the client. SYMCLI through a remote server on page 210 provides instructions.

Installation checklist

This section provides operating system-specific checklists with high-level installation and configuration steps that advanced Windows and UNIX users may find useful:

- Windows installation check list on page 45
- UNIX installation check list on page 47

Windows installation check list

Table 12 Windows installation check list

Task	More Information	Done
Pre-Installation		
Ready the environment for Solutions Enabler.	For instructions and requirements, refer to Before you begin on page 20 and Environment and system requirements on page 28, respectively.	
Installation		
1. Download the installation package	N/A	
2. Start the installation wizard by running the following: `se8300-WINDOWS-x64.exe`	For information on running the installation from the command line, refer to Using the command line on page 70. If you select the custom installation option, Table 16 on page 68 describes the available options.	
Post installation		
1. Enable the Solutions Enabler features with the following command: `symlmf add`	For more information, refer to "Licensing your software" on page 85.	
2. Build the SYMAPI database by entering the following command: `symcfg discover`	For more information, refer to "Building the SYMAPI database" on page 107.	
3. Set the environment variables so you can directly	For more information, refer to "Setting environment	

Table 12 Windows installation check list (continued)

Task	More Information	Done
access the SYMCLI commands by ensuring that the following SYMCLI directory is appended to the MS-DOS variable path: `C:\Program Files\EMC\SYMCLI\bin`	variables" on page 108.	
4. *Optional*: Read the *EMC VMAX Family Security Configuration Guide* and apply related security settings.	For more information, refer to EMC VMAX Family Security Configuration Guide.	
5. *Optional*: Modify the scope/performance of the SYMCLI commands with the `gkavoid`, `gkselect`, `inqfile`, `symavoid` files.	For more information, refer to Avoidance and selection files on page 136.	
6. *Optional*: Create an options file to modify the default behavior of Solutions Enabler. This file is initially installed as `README.options` in the SYMAPI configuration directory.	For more information, refer to "Changing the default behavior of SYMCLI" on page 112.	
7. *Optional*: Configure the necessary daemons for the environment.	For instructions, refer to: • Setting up daemons for distributed application support on page 139 • "Managing the base daemon" on page 119 • "Setting up the event daemon for	

Table 12 Windows installation check list (continued)

Task	More Information	Done
	monitoring" on page 120	

UNIX installation check list

Table 13 UNIX installation check list

Task	More Information	Done
Pre-Installation		
Ready the environment for Solutions Enabler.	For instructions and requirements, refer to Before you begin on page 20 and Environment and system requirements on page 28, respectively.	
Installation		
1. Download the installation package	For operating system-specific commands, refer to Step 1: Download the installation package on page 52.	
2. Run the installation script. For example, to run the full interactive script, enter the following command: `./se8300_install.sh`	For information on running alternative installation methods, such as silent, incremental, or response file, refer to Step 2: Run the install script on page 52.	
3. Verify the installation by entering the following command: `./se8300_install.sh - check`	For more information, refer to Verifying your installation on page 63.	

Table 13 UNIX installation check list (continued)

Task	More Information	Done
4. Optional: Remove the temporary file: `/tmp/emc_app_data_path`	For more information, refer to Removing temporary file on page 64.	
Post installation		
1. Enable the Solutions Enabler features with the following command: `symlmf add`	For more information, refer to eLicensing on page 122.	
2. Build the SYMAPI database by entering the following command: `symcfg discover`	For more information, refer to Building the SYMAPI database on page 132.	
3. For Linux Kernel 2.4, compile the SCSI generic driver into the kernel or compile it as a loadable kernel module.	For instructions, refer to the README file in the top-level directory of the Linux source package.	

4. Set the environment variables so you can directly access the SYMCLI commands:
For UNIX C shell, ensure the following SYMCLI directory is appended to variable PATH:
`set path = ($path /usr/symcli/bin)`
For UNIX Korn and Bourne shell, ensure the following SYMCLI directory is appended to variable PATH:
`PATH=$PATH:/usr/symcli/bin`
`export PATH`
	For more information, refer to Setting environment variables on page 132.	

Table 13 UNIX installation check list (continued)

Task	More Information	Done

5. Set the environment variable so you can directly access the online help (man pages):
For UNIX C shell, ensure the following man page directories are added to variable `MANPATH`:
`set MANPATH = ($MANPATH /usr/storapi/man /usr/storapi/storman)`
For UNIX Korn and Bourne shell, ensure the following man page directories are added to variable `MANPATH`:
`MANPATH= $MANPATH:/usr/storapi/man:/usr/storapi/storman`
`export MANPATH`
	For more information, refer to Setting environment variables on page 132.	

6. Configure an adequate number of semaphores into the UNIX kernel to meet the SYMCLI semaphore requirements.	For more information, refer to Managing database and gatekeeper locking on page 134.	
7. *Optional*: Read the EMC VMAX Family Security Configuration Guide and apply related security settings.	For more information, refer to EMC VMAX Family Security Configuration Guide.	
8. *Optional*: Modify the scope/performance of the SYMCLI commands with the `gkavoid`, `gkselect`,	For more information, refer to Avoidance and selection files on page 136.	

Table 13 UNIX installation check list (continued)

Task	More Information	Done
`inqfile`, `symavoid` files.		
9. *Optional*: Create an options file to modify the default behavior of Solutions Enabler. This file is initially installed as `README.options` in the SYMAPI configuration directory.	For more information, refer to Changing the default behavior of SYMCLI on page 137.	
10. *Optional*: Configure the necessary daemons for the environment.	For instructions, refer to: • Setting up daemons for distributed application support on page 139 • Managing the base daemon on page 145 • Setting up the event daemon for monitoring on page 148	

CHAPTER 2

Installation

This chapter explains how to install/upgrade Solutions Enabler and its components.

Note

As an alternative to the in-depth UNIX and Windows procedures in this chapter, Installation checklist on page 44 provides operating-system-specific checklists with high-level installation and configuration steps that advanced users may find useful.

Installing Solutions Enabler on UNIX and Linux

This section describes how to install/upgrade Solutions Enabler on UNIX and Linux hosts.

Please consider the following before starting the installation procedure:

- Solutions Enabler V8.3 is fully upgradeable, that is, you do not have to remove the previous version before installing V8.3.
- Before starting this procedure, be sure to review pre-install considerations in Installation prerequisites on page 19.
- The default responses to the prompts in this section are in brackets [ ].

Step 1: Download the installation package

To download the installation package:

Procedure

1. Log onto the host system as `root`.
2. Open a browser and visit the EMC online support website at https://support.EMC.com.
3. Download the installation package for your platform and extract the content to a temporary directory.

Note

To download the software for Solutions Enabler V8.3, please contact your EMC representative.

Step 2: Run the install script

To run the installation script:

Procedure

1. Change directory to the location of the Solutions Enabler kit by entering the following:

```
cd /tmp_directory
```

2. Select an installation method from Table 14 on page 52, and then run the appropriate command. For descriptions of the command options, refer to Table 15 on page 56.

Table 14 Installation method

Method	Command	Comments
Interactive	`./se8300_install.sh -install`	Starts the interactive script documented in the remainder of this

Table 14 Installation method (continued)

Method	Command	Comments
		chapter. When using this method, continue with Step 3: Select the installation directories on page 58.
Silent (all components)	`./se8300_install.sh -install -silent [-all]`	Silently installs the default Solutions Enabler components, or all Solutions Enabler components when the `-all` option is specified. When using this method, continue with Step 5: Complete the installation on page 63.
	`./se8300_install.sh -install -silent -nocert [-all]`	Silently installs the default Solutions Enabler components, or all Solutions Enabler components when the -all option is specified, but without the default SSL certificate files. When using this method, continue with Step 5: Complete the installation on page 63.
Silent (specific components)	`./se8300_install.sh -install -silent [-nocert] [-jni] [-srm] [-all] [-symrec] [-smis] [-lockboxpassword] [-force] [-daemonuid] [-permission] [-`	Silently installs only the specified components. When using this method, continue with Step 5: Complete the installation on page 63.

Table 14 Installation method (continued)

Method	Command	Comments
	`homedir] [-datadir] [-nodeps] [-copy_lic] [-tc] [-nocert]`	
Incremental (specific components)	`./se8300_install.sh -increment [-cert][-jni] [-srm] [-symrec]`	Incrementally adds the specified component to an existing installation. When using this method, continue with Step 5: Complete the installation on page 63. To use this method, you must have already installed the DATA, THINCORE, BASE, and SYMCLI components. **Note** This method is not supported on Solaris.
Response file	`./se8300_install.sh -file` *`Response_File_Name`*	Runs the installation script according to the contents of your response file. To use this method, create a response file containing the relevant command line options (refer to the examples on the next page), and then run the command, specifying the name of your text file. Response file entries can be

Table 14 Installation method (continued)

Method	Command	Comments

		separated by a space or on separate lines and options must not have leading hyphens.
		
		Using this method, you can specify the argument INCREMENT to perform an incremental installation or SILENT to perform a silent installation.
		
		For example, to incrementally install the SYMRECOVER component:
		
		a. Create the following response file:
		
		`# cat responsefile.txt increment symrec #`
		
		b. Run the command:
		
		`./se8300_install.sh -file responsefile.txt`
		
		For example, to silently install Solutions Enabler with the Java Interface and SRM components:

Table 14 Installation method (continued)

Method	Command	Comments

		a. Create the following response file:
		`# cat responsefile.txt`
		`install silent jni srm`
		`#`
		b. Run the command:
		`./se8300_install.sh -file responsefile.txt`
		When using this method, continue with Step 5: Complete the installation on page 63.

Table 15 on page 56 defines the various options used when running the installation commands detailed in Table 14 on page 52.

Table 15 UNIX installation options

Option	Description
`-all`	Installs all of the optional Solutions Enabler components, including the Java Interface; the Oracle, UDB, and Sybase daemons; and the SYMRECOVER component. Used with the `-silent` option.
`-cert`	Install SSL certificate files.
`-copy_lic=directory`	Copies the user-supplied `symapi_licenses.dat` file to `/var/symapi/config` during installation. Used with the `-silent` option. For example, the following

Table 15 UNIX installation options (continued)

Option	Description
	command will copy the `symapi_licenses.dat` file from `/tmp` to `/var/symapi/config`: `bash-3.00# ./se8300_install.sh -install -copy_lic=/tmp -silent`
`-daemonuid=Name`	Changes ownership of some daemons to non root user. Used with the `-silent` option. For information on which daemons are affected by this option, refer to the stordaemon man page in the EMC Solutions Enabler SYMCLI Command Reference Guide.
`-datadir=directory`	Sets the working root directory [**`/usr/emc`**]. Used with the `-silent` option.
`-decrement`	Uninstall of cert, jni, srm, smis (Linux only), symrec. This option is not valid for Solaris hosts
`-file`	Specifies to install Solutions Enabler with a response file.
`-force`	Kills all processes using the SYMAPI libraries. Used with the `-silent` option.
`-homedir=directory`	Sets the install root directory [**`/opt/emc`**]. Used with the `-silent` option.
`-increment`	Incremental installation of the `cert`, `jni`, `srm`, `smis` (Linux only), and `symrec` options. This option is not valid for Solaris hosts!
`-jni`	Installs the Solutions Enabler Java Interface component.
`-nocert`	Do not install SSL certificate files.

Table 15 UNIX installation options (continued)

Option	Description
`-permission=level`	Sets permission on **`/var/symapi`** directory. Used with the `-silent` option.
`-silent`	Specifies to perform a silent installation.
`-smis`	Installs the SMISPROVIDER component.
`-srm`	Installs all of the optional database components, including the Oracle, UDB, and Sybase daemons.
`-symrec`	Installs the SYMRECOVER component.
`-tc`	Installs THINCORE components (data and thin core).
`-lockboxpassword=password`	Sets the password for the lockbox. The password must be at least eight characters long, containing at least one uppercase letter, one lowercase letter, one number, and one special character. Allowed special characters are !@#%&. Used with the `-silent` option. For detailed information about the lockbox, please refer to the *EMC VMAX Family Security Configuration Guide*.

Note

For help running the installation script, run `./se8300_install.sh -help`

Note

The installation script creates log files in the directory `/opt/emc/logs`. For more information, refer to UNIX Installation Log Files on page 271.

Step 3: Select the installation directories

Procedure

1. To select the installation directories, do one of the following:
   - If you are installing Solutions Enabler on a host for the first time, complete Step 3A: Installing for the first time on page 59.
   - If you are upgrading or reinstalling Solutions Enabler, complete Step 3B: Upgrading /reinstalling on page 60.

Note

It is recommended that you install Solutions Enabler on your host's internal disks and not on a network device.

Step 3A: Installing for the first time

If you are installing Solutions Enabler on a Linux host for the first time, the following prompt displays:

```
Do you want to import public key for verifying Digital Signatures ?
[Y]:
```

A [Y]es response imports the public key for verifying Digital Signatures.

A [N]o response does not import the public key.

If you are installing Solutions Enabler on a host for the first time, the following prompt displays:

```
Install Root Directory [/opt/emc]:
```

Procedure

1. Press **Enter** to accept the default installation directory `/opt/emc`, or enter another root directory.

   If you enter a root directory (absolute directory) other than the default, you will be prompted to confirm the directory.

2. At the following prompt, press **Enter** to accept the default working directory /usr/emc, or enter another working directory. This directory is where the data and log files will be written:

   ```
   Working root directory [/usr/emc]:
   ```

   If you enter a working directory (absolute path) other than the default, you will be prompted to confirm the directory.

3. At the following prompt, specify whether to run the SYMAPI Server daemon, event daemon, Group Name Services daemon, and Watchdog daemon without

root privileges. A [Y]es response will enable you to specify a non-root user to run the daemons:

```
Following daemons can be set to run as a non-root user:
storevntd, storgnsd, storrdfd, storsrvd, storstpd, storwatchd
Do you want to run these daemons as a non-root user? [N]:
```

4. Continue with Step 4: Select installation options on page 60.

Step 3B: Upgrading /reinstalling

If you are upgrading or reinstalling Solutions Enabler, the following prompt displays:

```
Install root directory of previous installation: /opt/emc
Do you want to change Install root Directory ? [N]:
```

Procedure

1. Respond [N]o to install Solutions Enabler into the same root directories (install and working) as the previous installation, or respond [Y]es to display the following prompts in which you can enter other root directories:

   ```
   Install root directory [/opt/emc]:
   Working root directory [/usr/emc]:
   ```

   If you enter a root directory (absolute directory) other than the default, you will be prompted to confirm the directory.

2. If you are upgrading, the following prompt displays asking whether to backup the previous installation. A [Y]es response backs up the SYMCLI binaries in the install root directory under **symcli_old** :

   ```
   Do you want to save /opt/emc/SYMCLI/ ? [N]:
   ```

3. At the following prompt, specify whether to run the SYMAPI Server daemon, event daemon, Group Name Services daemon, and Watchdog daemon without root privileges. A [Y]es response will enable you to specify a non-root user to run the daemons:

   ```
   Following daemons can be set to run as a non-root user:
   storevntd, storgnsd, storrdfd, storsrvd, storstpd, storwatchd
   Do you want to run these daemons as a non-root user? [N]:
   ```

4. If the installation program detects that there are daemons currently running, the following prompt displays asking whether to shut them down or exit the installation. A [Y]es response shuts down the daemons. A [X] response exits the installation:

   ```
   Do you want to shutdown SYMCLI daemons [Y] or Exit setup [X]?
   [Y]:
   ```

5. Continue with Step 4: Select installation options on page 60.

Step 4: Select installation options

To select your installation options:

Procedure

1. At the following prompt, specify whether to install Solution Enabler SSL certificate files:

```
Install EMC Solutions Enabler Certificates for secure Client/
Server operation? [Y]:
```

   - A [Y]es response installs `ssl.rnd`, `symapisrv_install.cnf`, `symapisrv_trust.pem`, `symapisrv_trust_v8.3.pem` in `/var/symapi/config/cert`. The subject certificate and key files `symapisrv_cert.pem`, `symapisrv_key.pem`, `symapisrv_cert_v8.3.pem`, `symapisrv_key_v8.3.pem` will also be generated.
   - A [N]o response does not install CERT component.

   Note

   If you do not install SSL certificate files at this time but intent to use secure client/server communication with Solutions Enabler, you must install your own certificate files after the installation is completed. For detailed information on how to do that, please refer to the *EMC VMAX Family Security Configuration Guide*

2. At the following prompt, specify whether to install all of the Solutions Enabler libraries:

```
Install All EMC Solutions Enabler Shared Libraries and Run
Time Environment? [Y]:
```

   - A [Y]es response installs all the libraries, including persistent data, Thin Core, and Base (which includes the StorBase, StorCtrl, and StorMap library components).
   - A [N]o response installs only persistent data and Thin Core.

3. At the following prompt, specify whether to install the collection of binaries known as SYMCLI. A [Y]es response installs the SYMCLI binaries:

```
Install Symmetrix Command Line Interface (SYMCLI)? [Y]:
```

4. At the following prompt, specify whether to install the Solutions Enabler Java interface component. You should install this component if your Solutions Enabler application uses a Java interface. A [Y]es response installs the JNI component:

```
Install Option to Enable JNI Interface for EMC Solutions
Enabler APIs? [N]:
```

5. If you are installing Solutions Enabler on a host with a Linux, HP-UX, SunOS, or AIX operating system, the following prompt displays, asking whether to install optional database components:

```
Install EMC Solutions Enabler SRM Components? [N]:
```

A [Y]es response installs the following SRM database subcomponents, depending on the operating system:

- SRM Oracle Database files
  Installs the optional Oracle daemon on operating systems where Solutions Enabler supports Oracle.
- SRM Sybase Database files
  Installs the optional Sybase daemon on operating systems where Solutions Enabler supports Sybase.
- IBM UDB Database files
  Installs the optional UDB daemon on operating systems where Solutions Enabler supports UDB.

6. At the following prompt, specify whether to install the Solutions Enabler SRDF session recovery component. A [Y]es response installs the SYMRECOVER component:

```
Install EMC Solutions Enabler SYMRECOVER Components ? [Y]:
```

7. At the following prompt, specify whether to install the Solutions Enabler SMI-S Provider component. A [Y]es response installs the SMISPROVIDER component:

```
Install EMC Solutions Enabler SMIS Component ? [N]:
```

8. At the following prompt, specify whether to change the default UNIX file permissions. [Y]es response displays another prompt in which you can specify a new value:

```
Do you want to change default permission on /var/symapi
directory from [755] ? [N]:
```

9. At the following prompt, specify whether you want to use the default lockbox password. A [N]o response leaves the default password unchanged and the installation continues:

```
Do you want to use the default Lockbox Password? [N]:
```

A [Y]es response results in a confirmation request to make sure you really intend to use the default password for the lockbox:

```
Please confirm that you want to use the default Lockbox
Password [N]:
```

A [N]o response results in a prompt for the new password:

```
Please enter the Lockbox Password:
```

If the password meets the recommended password complexity, the installation asks you to re-enter the same password for confirmation:

```
Please re-enter the Password for confirmation:
```

Note

If you choose to use the default lockbox password generated by the installation program, you will have to make a note of it for future use if you need to reset the lockbox Stable System Values or generate certificates for client/server operation. See the *EMC VMAX Family Security Configuration Guide* for a description of how the default lockbox password is generated.

Note

If you change the default lockbox password, the default ECOM password is also changed from `admin/#1Password` to `admin/<specified password during installation>`.

10. If you are upgrading, the following prompt displays, asking whether to move the data files of the previous installation to the `symapi_old` directory. A **[Y]**es response moves your persistent data from the `/usr/emc/API/symapi` directory to `/usr/emc/API/symapi_old`. A **[N]**o response retains your persistent data:

```
Do you want to move this data to /usr/emc/API/symapi_old ?
[N]:
```

11. At the following prompt, decide whether you want to use the default lockbox password. A **[N]**o response leaves the default password unchanged and the installation continues:

```
Do you want to use the default Lockbox Password? [N]:
```

Step 5: Complete the installation

This section explains how to complete your Solutions Enabler installation.

Verifying your installation

To verify your installation, run the following command:

```
./se8300_install.sh -check
```

The output of this command depends on the installation options selected during the installation steps. This command produces an output similar to the following example in a Linux environment:

```
-bash-2.05b# ./se8300_install.sh -check

#------------------------------------------------------------------
-
#                            EMC Installation Manager
#------------------------------------------------------------------
--
Copyright (c) [1997-2016] EMC Corporation. All Rights Reserved.
This software contains the intellectual property of EMC Corporation
or is licensed to EMC Corporation from third parties. Use of this
software and the intellectual property contained therein is
```

```
expressly limited to the terms and conditions of the License
Agreement under which it is provided by or on behalf of EMC.
Checking for Solutions Enabler Native Installer kit
Installation......
  Sl No RPM                              Version
  ----- ---                              --------
    1   symcli-base                      8.3.0.1707-0.3
    2   symcli-cert                      8.3.0.1707-0.3
    3   symcli-data                      8.3.0.1707-0.3
    4   symcli-symcli                    8.3.0.1707-0.3
    5   symcli-symrecover                8.3.0.1707-0.3
    6   symcli-thincore                  8.3.0.1707-0.3
```

Removing temporary file

During installation, the install script creates the temporary file `/tmp/emc_app_data_path`. This file holds the value that was entered for the install root directory from the previous installation. This value is used as the default install root directory in subsequent installations.

For example:

```
EMC_APPLICATION_PATH:/OPT/EMC
```

In some cases this file will be removed when you reboot your system. If not, you may want to manually remove it to conserve disk space.

Unmounting the installation disc

To unmount the installation disc, enter:

```
unmount mount_point
```

Enabling the Solutions Enabler components

Enable your Solutions Enabler features by entering the appropriate license keys.

Note

For instructions, refer to eLicensing on page 122.

Creating certificate files after initial installation

If the certificate component is not initially installed, and then added by running the installer again or by performing an incremental install, the SSL certificate is not created.

You can create the SSL certificate by entering the following:

```
cd /var/symapi/config/cert
/usr/symcli/bin/manage_server_cert create -pass <lockbox_pwd>
```

where *<lockbox_pwd>* is the lockbox password created during the installation process.

Installing Solutions Enabler on Windows

You can install/upgrade Solutions Enabler on a Windows host using the InstallShield wizard (described below), the command line (refer to Using the command line on page 70), or a response file (refer to Using a response file on page 74).

Note

Solutions Enabler V8.3 is fully upgradeable. That is, you do not have to remove the previous version before installing V8.3.

Note

Before starting this procedure, review the pre-install considerations in Installation prerequisites on page 19.

Using the InstallShield wizard

To install/upgrade Solutions Enabler using the InstallShield wizard:

Procedure

1. Open a browser and visit the EMC online support website at https://support.EMC.com.
2. Download the installation package for your platform and extract the content to a temporary directory.
3. Save all files and exit all Windows applications.
4. Change directory to the location of the Solutions Enabler kit by entering the following:

   ```
   cd \tmp_directory
   ```

5. Start the installation program by running the following `se8300-WINDOWS-x64.exe` file.

   Note

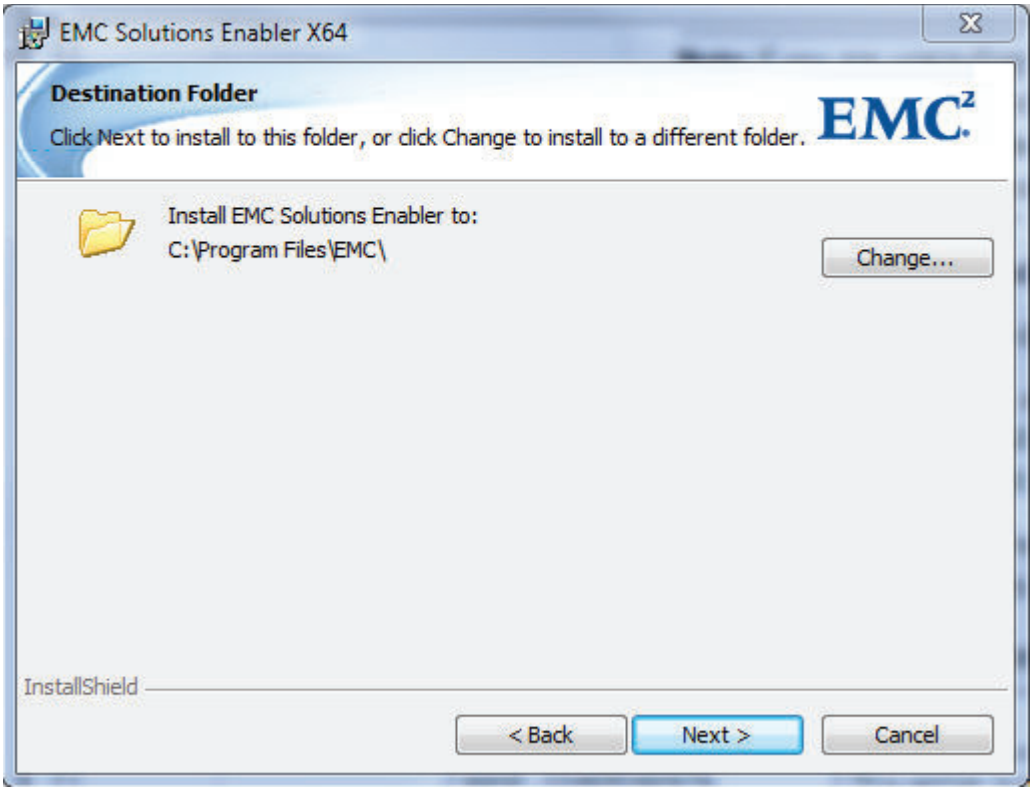
   If you do not have the required Visual C libraries installed on the host to run Solutions Enabler, you will be prompted to install them. If this is the case, click **Install** in the message dialog.

   Note

   If you are upgrading from a previous version of Solutions Enabler and the installation program detects that there are daemons running, you will be prompted to shut them down. Click **Yes** to shutdown the daemons and continue with the installation. Click **No** to leave the daemons running and exit the installation program.

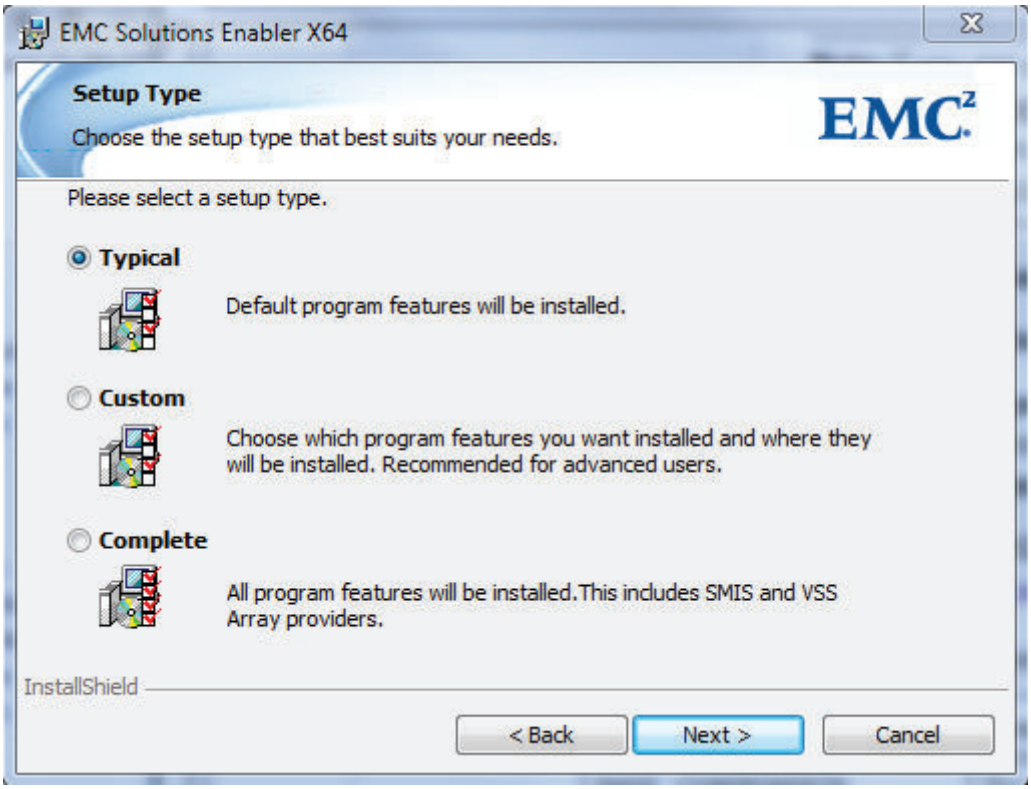
6. In the **Welcome to the Installation program for EMC Solutions Enabler** dialog box, click **Next**.
7. In the Destination Folder dialog box, select an installation directory and click **Next**.

Figure 2 Destination folder dialog box



8. In the **Setup Type** dialog, select **Typical** to install the default components, select **Complete** to install the full Solutions Enabler product set (along with SMI-S and VSS), or select **Custom** to install a subset of the options. Click **Next** when done.

Figure 3 Setup type dialog box



9. If you selected Custom, the Custom Setup dialog box opens. Select the options, listed in Table 16 on page 68, to install, where to install them, and then click **Next**.

Figure 4 Custom setup dialog box

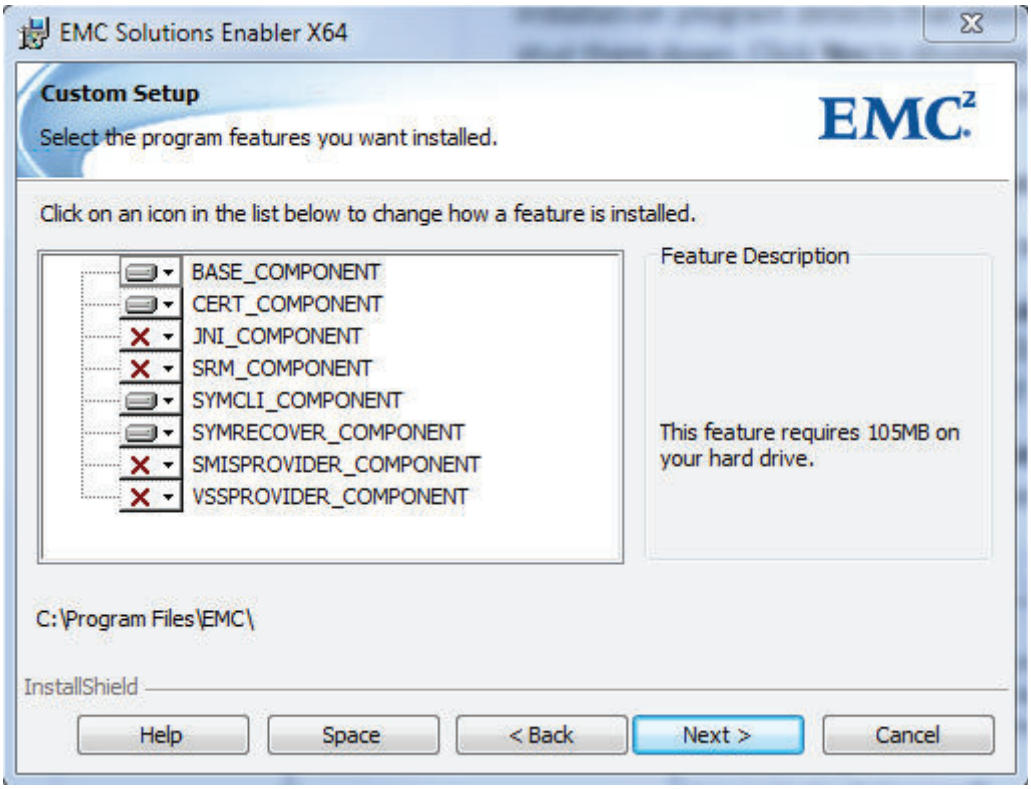


Table 16 Windows installation options

Option	Description
BASE_COMPONENT	This option is part of the shared library and runtime environment. It is a co-requisite for other options, and is therefore mandatory for a successful installation. It installs the following: • Solutions Enabler core functionality, including symapi, symlvm, storapi, storapid, storcore, stordaemon, and storpds. • The `storsil` and `storbase` libraries, which provide base storage and host-specific functionality, and an interface to storage arrays for features like I/O scan, device listings, statistics, and showings.

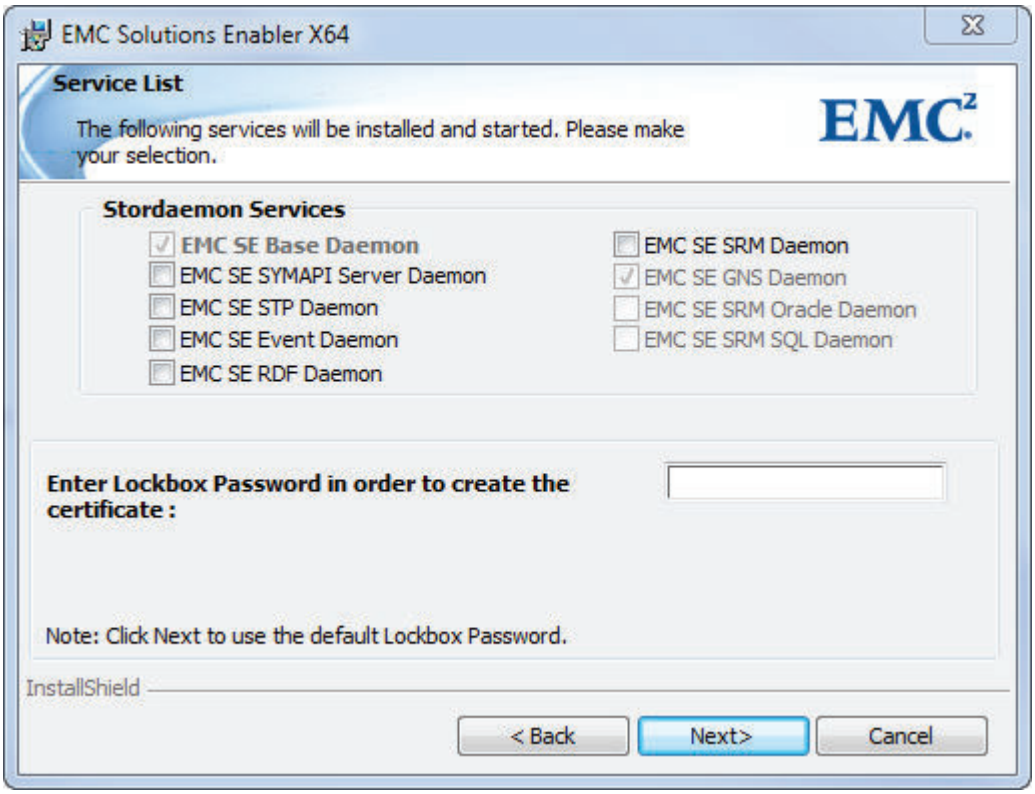
Table 16 Windows installation options (continued)

Option	Description
	• The control storage libraries, which include features like Snap, device masking, and device monitoring. • The Storage Resource Management base mapping library.
CERT_COMPONENT	Installs the `ssl.rnd`, `symapisrv_install.cnf`, `symapisrv_trust.pem`, `symapisrv_trust_v8.3.pem` in `C:\Program Files\EMC\SYMAPI\config\cert`. The subject certificate and key files `symapisrv_cert.pem`, `symapisrv_key.pem`, `symapisrv_cert_v8.3.pem`, `symapisrv_key_v8.3.pem` will also be generated.[a]
JNI_COMPONENT	Installs the Solutions Enabler Java Interface component. You should install this component if your Solutions Enabler application uses a Java interface.
SRM_COMPONENT	Installs the IBM UDB, SQLServer, and Oracle components (depending on the host platform).
SYMCLI_COMPONENT	Installs the collection of binaries known as SYMCLI.
SYMRECOVER_COMPONENT	Installs the SRDF session recovery component.
SMISPROVIDER_COMPONENT	Installs the SMI-S Provider component.
VSSPROVIDER_COMPONENT	Installs the VSS Provider component.

a. If you do not install SSL certificate files but intents to use secure client/server communication with Solutions Enabler, you must install your own certificate files after the installation is completed. For detailed information on how to do that, please refer to the EMC VMAX Family Security Configuration Guide.

10. In the **Service List** dialog, select the services to install/start. The services available in this dialog are based on the installation options you selected. Setting up daemons for distributed application support on page 139includes descriptions of the Solutions Enabler daemons.

Figure 5 Service list dialog box



11. Specify the lockbox password and confirm it. If you do not specify a password during installation, the installer will use the default password. For detailed information on the lockbox, please refer to the *EMC VMAX Family Security Configuration Guide*.

Note

If you change the default lockbox password, the default ECOM password is also changed from `admin/#1Password` to `admin/<specified password during installation>`.

---

12. In the **Ready to Install the Program** dialog, click **Install** .
13. In the **Installation Program Complete** dialog box, click **Finish** to complete the setup, and then go to eLicensing on page 122.

Using the command line

The `se8300-WINDOWS-x64.exe` is a wrapper for MSI installs. The MSI kit is embedded inside the executable and provides more flexibility.

In general, the `se8300-WINDOWS-x64.exe` is a two step process: first it extracts the MSI kit, and then MSI extracts all the files using `msiexec.exe`.

To install/upgrade Solutions Enabler using the command line:

Procedure

1. Open a browser and visit the EMC online support website at https://support.EMC.com .
2. Download the installation package for your platform and extract the content to a temporary directory.
3. Save all files and exit all Windows applications.
4. Select one of the MSI wrapper script installation options, detailed in the remainder of this section.

Note

By default, the installation program will generate a verbose log (`SE_RTinstall_Verbose.log`) for each install in the TEMP directory.

Silent mode

To install Solutions Enabler in silent mode, enter:

```
start /wait se8300-WINDOWS-x64.exe /s /v/qn
```

Where:

`/S` or `/s` is the silent option for the wrapper script. The `/s` option is used for silent extraction of MSI kit from the wrapper to a temp folder. The `/s` option is not related to the MSI kits.

`/V` or `/v` is the option used by the wrapper to parse the parameters to `msiexec.exe` when MSI kits are run after extraction. In other words, it is a gateway for the `msiexec.exe`. Whatever valid MSI parameters are passed after `/V` will be parsed to the `msiexec.exe`.

`/qn` is a regular msiexec option to install the MSI kits in silent mode.

Note

If the `/s` and `/v` options are entered as capital letters (`/S /V`), and a space is used to separate the `/v` and `/qn` options, the installation starts in Wizard mode.

Non-default location

To install Solutions Enabler in a non-default location, enter:

```
start /wait se8300-WINDOWS-x64.exe /s /V"INSTALLDIR=C:\EMC /qn"
```

Where:

`/V` or `/v` is the option used by the wrapper script to parse the parameters to `msiexec.exe` when MSI kits are run after extraction. In other words, it is a gateway for the `msiexec.exe`. Whatever valid MSI parameters passed after `/V` will be parsed to the `msiexec.exe`.

INSTALLDIR is a MSIEXEC public property. By using this as shown in the example, you can redirect your installation to a non default directory.

Space in directory name

To install in a non-default path with a space in the directory name or path, enter:

```
start /wait se8300-WINDOWS-x64.exe /S /V”INSTALLDIR=\”C:\Program
Files\ Non DefaultPath\” /qn”
```

Where:

\ is the escape character to insert the codes (“”) if there is a space in the directory path.

/qn is a regular MSIEXEC option to install the MSI kits in silent mode.

Adding non-default features

To perform a custom install (incremental) to add non-default Solutions Enabler features, enter:

```
start /wait se8300-WINDOWS-x64.exe /S /
V"ADDLOCAL=JNI_COMPONENT,SRM_COMPONENT
LOCKBOXPASSWORD=<PASSWORD> /qn"
```

Where:

ADDLOCAL is a MSIEXEC public property. By using this as shown in the example, you can install optional features.

/qn is a regular MSIEXEC option to install the MSI kits in silent mode.

ADDLOCAL=ALL will perform a complete installation.

Note

If the LOCKBOXPASSWORD argument is not passed, then the default lockbox password will be used.

Removing non-default features

To perform a custom install (decremental) to remove non-default Solutions Enabler features, enter:

```
start /wait se8300-WINDOWS-x64.exe /s/
V”REMOVE=JNI_COMPONENT,SRM_COMPONENT /qn”
```

Where:

REMOVE is a MSIEXEC public property. By using this as shown in the example, you can remove optional features.

/qn is a regular MSIEXEC option to remove the MSI kits in silent mode.

Note

REMOVE=ALL will uninstall completely.

Multiple commands

To have multiple commands passed:

```
start /wait se8300-WINDOWS-x64.exe /S /V"INSTALLDIR=\"C:\Program
Files\Some Folder\" ADDLOCAL=SRM_COMPONENT /qn"
```

Overwrite mode

To run installer in overwrite mode:

```
start /wait
             se8300-WINDOWS-x64.exe /S /V"REINSTALLMODE=VOMUS
REINSTALL=ALL /qn"
```

Where:

`REINSTALLMODE` & `REINSTALL` are MSIEXEC public property

`/qn` is a regular MSIEXEC option to install the MSI kits in silent mode.

Maintenance mode

To run the installer in Maintenance custom mode:

```
start /wait
             se8300-WINDOWS-x64.exe /S /V"REINSTALLMODE=VOMUS
ADDLOCAL=SRM_COMPONENT /qn"
```

Starting services

To start three Solutions Enabler services, use the silent install command:

```
start /wait
             se8300-WINDOWS-x64.exe /S /V"ADDLOCAL=ALL STORAPID=1
STOREVNTD=1 STORSRVD=1 /qn"
```

Where:

`ADDLOCAL=ALL` will install every Solutions Enabler feature, including SMI-S and VSS, `STORAPID=1 STOREVNTD=1 STORSRVD=1` will install, start, and set the `storapid`, `storevntd`, and `storsrvd` services to start automatically.

Starting the storstpd daemon

When installing Solutions Enabler on a Windows host, the option to install/start the performance collector service (`storstpd` daemon) in the Select Services dialog box will only install the daemon; it will not start it. To start the daemon after you have finished the installation, use the following command:

```
stordaemon start storstpd
```

Default Solutions Enabler components

With the exception of the CORE component, all the following can be blocked from installation using the `REMOVE` command:

`CERT_COMPONENT`

`SYMCLI_COMPONENT`

`SYMRECOVER_COMPONENT`

Non-default Solutions Enabler components

The non-default components can be installed using the `ADDLOCAL` command:

`JNI_COMPONENT`

`SRM_COMPONENT`

Using a response file

Solutions Enabler provides the option of using a response file for installing on Windows hosts.

To install Solutions Enabler using a response file:

```
start /wait
                se8300-WINDOWS-x64.exe /s /
V"WSC_CONFIG_FILE=path_to_response_file_with_the_filename /qn"
```

To use this method, create a response file similar to the following example, and then run the command, specifying the name of your file.

In the response file:

- Set the components you want to install to True and the components that you do not want to install to False.
- Set the daemons you want to automatically start to 1 and the daemons you do not want to automatically start to 0.

Sample response file and contents:

```
[COMPONENTSELECTION]

CERT_COMPONENT:TRUE
SYMRECOVER_COMPONENT:TRUE
JNI_COMPONENT:TRUE
SYMCLI_COMPONENT:TRUE
SRM_COMPONENT:TRUE

[PATHSELECTION]

EMC_ROOT_PATH="C:\Program Files\EMC\"
EMC_DATA_ROOT_PATH="C:\Program Files\EMC\SYMAPI\"
WIDESKY_SDK_KEY="xxxx-xxxx-xxxx-xxxx"

[DAEMONSSELECTION]

STORAPID=1
STOREVNTD=0
```

```
STORGNSD=0
STORORAD=0
STORRDFD=0
STORSQLD=0
STORSRMD=0
STORSRVD=1
STORSTPD=0
```

Installing Solutions Enabler on z/OS

This section describes how to install Solutions Enabler on a z/OS host to operate as a SYMAPI server.

The following procedure can be used for either a new installation, or to upgrade an existing installation.

Note

Before starting this procedure, be sure to review the pre-install considerations in Installation prerequisites on page 19.

Step 1: Copy the files

To copy files:

Procedure

1. Open a browser and visit the EMC online support website at https://support.EMC.com .
2. Download the installation package for z/OS `emc.ssem830.zip` and extract the content to a temporary directory.
3. In the temporary directory, extract the files from the `.zip` file, and then execute the command `uploadSE.bat`.
4. When prompted, provide the following information:
   - The name or IP address of the z/OS host on which you are installing.
   - The userid and password to login to the FTP server on the z/OS host, and other optional FTP information.
   - The high-level qualifier of the dataset name to use during allocation of the distribution file.
   - The name of a volume and esoteric unit name on which to allocate the distribution file.

   Once the upload completes, the distribution file will be ready for remaining installation steps.
5. Once the files are uploaded, login to the z/OS host and continue the installation.

Note

If you plan on running the Solutions Enabler server using secure (SSL) communications, you must create and install the certificates for z/OS before starting the server. To do this, you must run the Windows batch file `zoscert.bat` from the same location you ran the `uploadSE.bat` batch file. You cannot do this until after you have run job `#07DFLTS`, as this job creates some requisite directories in the UNIX System Services file system. SSL certificates on page 185 provides more information.

Step 2: Receive the transmit file

The file that you transferred to the host was created using the TSO `TRANSMIT` command. Therefore, you must use the TSO `RECEIVE` command to convert the file to a library of materials that you will use to complete the installation.

To receive the transmit file:

Procedure

1. Do one of the following:
   - From the TSO READY prompt, enter the following command: `RECEIVE INDS('high_level_qualifier.EMC.ssem830.XMITFILE')` Where *high_level_qualifier* is the same qualifier used during the CD-based batch upload procedure.
   - In the **Utilities.DSList (3.4)** of the main ISPF menu, type **`RECEIVE INDS(/)`** on the line where the uploaded transmit file is shown in the list.

   In either case, the following displays:

```
INMR901I Dataset EMC.ssem830.XMITLIB from
emcdist on NODENAME
INMR906A Enter restore parameters or 'DELETE' or
'END'
```

2. Press **Enter** to accept the allocation of the XMITLIB under your high-level qualifier, or respond with the following to change the allocated dataset name:

```
DSN('ds_prefix.xmitlb')
```

Note

The dataset name you specify must end in the XMITLIB extension.

Step 3: Extract the additional files from the XMITLIB

Edit the job `$EXTRACT` member of the XMITLIB and make the following changes:

Procedure

1. Add a JOB card to comply with your site's batch JCL standards.
2. Change all occurrences of *ds-prefix* to the desired prefix for your Solutions Enabler libraries.

3. Change all occurrences of `DVOL` to the volume on which you want to allocate the libraries.
4. Change all occurrences of `DISK-UNIT` to the disk unit name that includes the volume you specified in the `DVOL` change above.
5. Submit the job, and look for a zero return code. The `$EXTRACT` job creates some temporary data sets which will be deleted by the `#99ECLN` job after the installation is complete. It also creates some data sets for permanent use with Solutions Enabler.

Step 4: Customize the JCL

Solutions Enabler includes a REXX exec program, SEMJCL, to expedite the JCL customization process by allowing you to create a site-specific ISPF edit macro in your CLIST library and then running it against every member of the RIMLIB whose name starts with a pound sign (`#`).

Note

If you prefer to manually customize the JCL, customize the `#` prefixed members as necessary, and then continue with Step 5: Run the jobs on page 79.

To use SEMJCL:

Procedure

1. In the **Utilities.DSList (3.4)** of the main ISPF menu, type the first few qualifiers of your RIMLIB dataset name, and then press **Enter**.

   The RIMLIB displays as part of the DSLIST.

2. Scroll to the RIMLIB dataset and type **`M`** in the command field.

   The member list for the RIMLIB dataset displays.

3. Scroll to the SEMJCL member in the RIMLIB, and then type **`exec`** (or **`ex`**) in the input area to the left of the member name.

   This executes the SEMJCL exec, which displays the customization screen:

```
.----- Customize EMC Solutions Enabler 8.3.0 Electronic Kit Install JCL ------.
| Command ===> ____________________________________________                    |
| Press PF3 to Cancel or PF1 for Help                                          |
| Press ENTER to run edit macro SEMX830 which                                  |
| will customize the installation JCL                                          |
|                                                                              |
|     Data Set Name Prefix:  EMC.SSEM830                                       |
|    SMP/E Data Set prefix:  EMC.SMPE                                          |
|         SCF Subsystem Id:  EMC                                               |
|       SCF Linklib Prefix:  EMC.SSCF760                                       |
|           Disk Unit Name:  SYSDA      Disk Volume Serial:  SYM001           |
|                Time Zone:  EST5                                              |
|    SYMAPI Base Directory:  /var/symapi                                       |
|                                                                              |
| Enter JOB card below ('%MEMBER%' is replaced by the member name):            |
| //USERIDA JOB ACCT,'EMC SEM 8.3',                                            |
| // CLASS=A,            <-- CHANGE IF NEEDED                                  |
| // MSGCLASS=A,         <-- CHANGE IF NEEDED                                  |
| // NOTIFY=USERID      <-- CHANGE IF NEEDED                                   |
|                                                                              |
|                                                                              |
|                                                                              |
'------------------------------------------------------------------------------'
```

4. Enter your site-specific information according to the following:

   Note

   To cancel the SEMJCL, press **PF3** (that is, the **END** key).

   a. In the **Data set name Prefix** field, enter the high-level qualifier and any additional qualifiers to be used when allocating new Solutions Enabler datasets.

   b. In the **SMP/E Data set prefix** field, enter the prefix of the SPM/E datasets where ResourcePak Base is installed.

   c. In the **SCF Subsystem Id** field, enter the subsystem name of the SCF address space. The default is `EMC`.

   d. In the **SCF Linklib Prefix** field, enter the prefix of the SCF load module library corresponding to the subsystem you entered above.

   e. In the **Disk unit name** field, enter a valid unit name defined at your site to be used in the UNIT= operand when allocating new Solutions Enabler datasets. The default is `SYSDA`.

   f. In the **Disk Volume Serial** field, enter the volume serial number of the DASD volume where the new Solutions Enabler datasets will be allocated.

   g. In the **Time Zone** field, enter the appropriate setting for your time zone location. This setting must be a POSIX-compliant time zone value. This value is used to set the TZ environment variable of the Solutions Enabler task. If you do not supply a value, the time stamps of the Solutions Enabler internal messages written to the log files will default to UTC time.

      For example, entering a value of `EST5` will set the time stamp to the United States Eastern Standard Time, 5 hours earlier than UTC.

⚠ CAUTION

The default time zone value is UTC time.

h. In the **SYMAPI Base Directory** field, specify the location of the Unix System Services directory under which the SYMAPI runtime directories will be created.

Note

The userid used in the Solutions Enabler batch jobs must have write access to the entire SYMAPI base directory.

i. In the **Job Card Information** field, specify up to four statements for your job card.

A default job card is filled in, including a place holder for accounting field, programmer name value, CLASS=A, MSGCLASS=A, and NOTIFY operands. The JOBNAME and NOTIFY= operands use the TSO ID of the user running the SEMJCL process.

If you use `%member%` in the jobname field in the job card, the RIMLIB member name will be used as the job name.

Note

Statement syntax is not validated until jobs are submitted.

j. Press **Enter**.

SEMJCL generates an edit macro and uses the ISPF editor to apply the specified values to all the installation jobs. At this point in the procedure, all of the installation jobs have been edited with site-specific information and are ready to run.

Step 5: Run the jobs

Procedure

1. Run each of the following jobs:
   - `#01ALLOC`
     Creates all the datasets not allocated by the `$EXTRACT` job for installing the product, and copies sample configuration members from the RIMLIB into the Solutions Enabler PARMLIB.
   - `#04DDDEF`
     Creates the DD definitions for all three SMP/E global zones.
   - `#05RECEV`
     Gets the `SYSMODS` and `HOLDDATA`. It also gets the `FMID` function, FMID(SSEM830), which delivers the Solutions Enabler for z/OS software.

Note

If job `#05RECEV` fails with the message: `GIM23401T the program IEV90 was required for SMP/E but was not available`Run `#ASMHA` to define IEV90, and then re-run `#05RECEV`

- `#06APPLY`
  Selectively applies the function received in the previous job:

  ```
  apply select(SSEM830)
  ```

  At this point you have installed the load library members into the target load library. The next few jobs execute programs in the load library, which have additional requirements. Be sure to check each program's requirements before submitting each job.
- `#07DFLTS`

Note

Before running job `#07DFLTS`, decide first if you want to use a specific lockbox password as opposed to the default one. Setting up the lockbox password is mandatory and must be completed before running job `#10ECCIN`. Refer to Step 6: Manage z/OS Lockbox password on page 82 before proceeding.

This job assembles and links the assembler source in member `#SYMDFLT`. `#SYMDFLT` will have been updated when the exec SEMJCL was run. This job also creates the SYMAPI directory structure, based on your specification of the SYMAPI Base directory on the **SEMJCL Customization** panel.

- `#08SLMF`
  Runs the Solutions Enabler License Management Facility (`symlmf`) in batch mode. You must use an editor to customize the input, entering the license keys from the key cards that were received with your Solutions Enabler package.

  The `symlmf` program normally runs in batch in z/OS, and the input to the program is specified in the `SYSIN DD` statement. The statements there satisfy the dialog that `symlmf` would normally have with an interactive user on non-z/OS platforms.

  The dialog sequence is as follows:

  a. At the following prompt, enter **`Y`** to begin the registration process:
  `Do you want to enter a registration key?` **`Y`**

  b. At the following prompt, enter the 19-byte key value as specified on the key card:
  `Enter the license key:`

  c. At the following prompt, enter **`Y`** to register another key value, or **`N`** to complete the registration process:
  `Do you want to enter a registration key?` **`N`**

Entering `N` causes `symlmf` to finish updating the license file and end the job step. The sample input below shows the appearance of the `SYSIN DD` statement coded to enter two keys:

```
000045 //SYMLMFI  EXEC PGM=SYMLMF
000046 //STEPLIB   DD
DSN=EMC.SSEM830.LOADLIB,DISP=SHR
000047 //SYSPRINT  DD  SYSOUT=*
000048 //SYSOUT    DD  SYSOUT=*
000049 //SYSIN     DD  *
000050  Y
000051  0000-1111-2222-3333
000052  Y
000053  3333-2222-1111-0000
000054  N
000055  /*
```

Note

For more on the new licensing mechanism, refer to eLicensing on page 122. For alternative ways of installing licenses in z/OS, refer to Installing using alternative methods on page 126.

Note

From this point on, the Solutions Enabler load library must be APF-authorized. The EMCSCF linklib will have been APF-authorized for SCF to operate. Use the desired method at your site to authorize the Solutions Enabler load library.

Also, the user who runs jobs from this point must have an OMVS segment defined. For more information, refer to Before you begin on page 20.

The ResourcePak Base (EMCSCF) address space must be active and must specify the same subsystem identifier (SSID) as the one specified on the JCL Customization panel.

- `#10ECCIN`

Note

The Solutions Enabler Base Daemon (storapid) must be started before job `#10ECCIN` is run.

This job creates the SYMAPI database for SYMCLI clients. Job `#10ECCIN` attempts to discover every VMAX system connected to your Mainframe host. If there are many VMAX arrays connected, this job may run for a considerable period of time. If there are VMAX arrays that you do not want remote clients to view, you may exclude them from the discover process. See section "symavoid" on page 112 for details on excluding devices.

Note

If the configuration of any VMAX array attached to a host is changed, then you must re-run job `#10ECCIN` to correctly discover the changed VMAX array. Alternatively, run a SYMAPI discover from any client which provides this capability.

Note

All 12 digits of the serial number are required.

- `#16CFGCP`
  Copies the sample configuration files to the SYMAPI configuration directory.

Step 6: Manage z/OS Lockbox password

Solutions Enabler V8.3 on z/OS has an ISPF interface (SEMLB) for managing the lockbox password. During the z/OS installation phase, the lockbox password will be set to the default value when the job `#07DFLTS` is run, during this step:

```
//LOCKBOX EXEC PGM=LOCKBOX
```

To complete the lockbox installation, follow these steps:

Procedure

1. If you wish to have the default lockbox password set during the initial install phase, then continue to step 2. If you do not wish to have the default lockbox password set during the initial installation phase, then delete (or comment out) the lockbox step before the job `#07DFLTS` is run for the first time.

   Note

   The lockbox step may be deleted (or commented out) before or after the SEMJCL configuration.

2. Complete the SEMJCL setup.
3. Run the job `#07DFLTS`.
4. Once job `#07DFLTS` has run (with or without the lockbox step), the SEMLB interface can be used. For details, see The SEMLB interface on page 83.

   Note

   The lockbox setup process must be completed before any daemons are started and job `#10ECCIN` is run.

5. Start daemons.
6. Run the job `#10ECCIN`.

Note

For detailed information about lockbox, please see the *EMC VMAX Family Security Configuration Guide*.

The SEMLB interface

After the `#07DFLTS` job has run, the SEMLB interface can be used to set the lockbox password. To do this, follow these steps:

Procedure

1. Navigate using the ISPF option 3.4 to the installation RIMLIB, locate the member SEMLB, and then use `exec` to execute it. The following panel will be displayed:

```
+------------------------------------------------------------------------------+
|      EMC Solutions Enabler 8.3.0 Lockbox configuration                       |
|                                                                              |
| Command ===> ____________________________________________                    |
|                                                                              |
|                                                                              |
|                                                                              |
| Enter option 1 or 2 or press PF3 to Cancel                                   |
|                                                                              |
|     1 - Set or reset the Lockbox Stable System Values                        |
|     2 - Change the Lockbox password                                          |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
+------------------------------------------------------------------------------+
```

2. Select option `1`. The following panel will be displayed:

```
+------------------------------------------------------------------------------+
|      EMC Solutions Enabler 8.3.0 Stable System Values reset                  |
|                                                                              |
| Command ===> ____________________________________________                    |
|                                                                              |
| Reset the lockbox SSV values:                                                |
|                                                                              |
|    Press enter to use the default password.                                  |
|    Otherwise type the password and press enter                               |
|                                                                              |
|                                                                              |
| Password                                                                     |
|                                                                              |
| Confirm Password                                                             |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
+------------------------------------------------------------------------------+
```

3. Do one of the following:
   a. If you ran the lockbox step in `#07DFLTS`, then enter the default password and press **Enter**. The Stable System Values will be reset.
   b. If you did not run the lockbox step in `#07DFLTS`, then enter a new password and press **Enter**. The Stable System Values will be set and the new password will now be in effect.

Changing the lockbox password

To change the lockbox password, follow these steps:

Procedure

1. Select option `2` when the SEMLB exec is invoked. The following panel will be displayed.

```
+------------------------------------------------------------------------------+
|      EMC Solutions Enabler 8.3.0 Lockbox Password change                     |
|                                                                              |
| Command ===> ____________________________________________                    |
|                                                                              |
|                                                                              |
| To change the lockbox password, enter the required passwords.                |
|                                                                              |
|                                                                              |
| Current password                                                             |
|                                                                              |
|                                                                              |
| New password.                                                                |
|                                                                              |
|                                                                              |
| Confirm new password.                                                        |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
+------------------------------------------------------------------------------+
```

2. Enter the current lockbox password as well as the new password and press **Enter**. The lockbox password will be changed to the new password.

Note

If you change the default lockbox password, the default ECOM password is also changed from `admin/#1Password` to `admin/<specified password during installation>`.

Quick step summary of lockbox installation

Installation steps using the default lockbox password:

1. Configure using SEMJCL (refer to Step 4: Customize the JCL on page 77).
2. Run `#07DFLTS`.
3. Change the default lockbox password using SEMLB option 2 (refer to Changing the lockbox password on page 84).
4. Start the daemons.
5. Run `#10ECCIN`.

Installation using a specific lockbox password:

1. Configure using SEMJCL (refer to Step 4: Customize the JCL on page 77).
2. Delete (or comment out) the lockbox step `//LOCKBOX EXEC PGM=LOCKBOX`.
3. Run `#07DFLTS`.
4. Set the lockbox password using SEMLB option 1 (refer to The SEMLB interface on page 83).
5. Start the daemons.

6. Run #10ECCIN.

Step 7: Complete the installation

Do the following to complete the installation:

Procedure

1. Perform all other customizing and any testing as required. Sample startup jobs are provided in the RIMLIB for the SYMAPI daemons:
   - #STORAPI - Base Daemon
   - #STOREVT - Event Daemon
   - #STORGNS - GNS Daemon
   - #STORSRV - Server Daemon

   Note that you can either run STORSRV as a batch job or convert it to run as a started task.
2. Customize and run job #11ACCPT. This job accepts the FMID SSEM830 into the distribution zone.
3. By default, control functions such as authorization, SRDF or TimeFinder are allowed from hosts external to the z/OS host (via client/server). To disable this capability, an optional zap must be applied. This zap is located in the RIMLIB in member #12CNTRL. Refer to both that job and Remote control operations on page 197 for further details.

   Your Solutions Enabler installation is now complete. Next, you need to establish your server environment by performing the configuration and setup procedures explained in z/OS Post installation configuration on page 185.

   Note

   If you plan on using the optional Secure Socket Layer (SSL) encrypted communications between the SYMAPI server and its connecting clients, and you plan on running the server in SECURE or ANY modes, you must create and install the SSL certificates before starting the server. For more information, refer to SSL certificates on page 185.

Starting over

If, while installing the product, you decide that you want to back out and start the installation over, you can do so up until you run job #11ACCPT.

There are two utility jobs in the RIMLIB that allow you to back out of an installation. Both are customized by the SEMJCL process along with other installation JCL. The members are:

- #99RESTR — Executes the SMP/E RESTORE command, which reverses the effect of an APPLY function. Use this job if you have successfully run #06APPLY and want to back out of that step.
- #99REJCT — Executes the SMP/E REJECT command, which reverses the effect of a RECEIVE function. Use this job if you have successfully run #05RECEV and want to back out of that step. You cannot REJECT an FMID that has been applied. You must RESTORE it before REJECTing it.

Note

`#99RESTR` and `#99REJCT` are not normally used in the installation process. You should only use these jobs to redo your installation.

Restoring the RIMLIB

In the event that customization of the RIMLIB has rendered it difficult to work with, you can use job `#RIMREST` in the RIMLIB to re-create the RIMLIB. This job will create a new RIMLIB with the suffix `.REST` and will not alter the original RIMLIB. However, you should verify that the JCL in `#RIMREST` is appropriate before running the job.

Installing Solutions Enabler on OpenVMS

This section describes how to install/upgrade Solutions Enabler on an OpenVMS host.

Note

Before starting this procedure, review the pre-install considerations in Installation prerequisites on page 19.

Step 1: Accessing the software

Solutions Enabler is distributed as a platform-specific file download from EMC online help at:

https://support.EMC.com

Possible filenames are:

`SE830RIA.SAV`	HP Integrity hardware platform.

Note

Throughout the remainder of this installation procedure, substitute the appropriate filename for any occurrence of the variable *InstallKit*

To access the software from EMC online help:

Procedure

1. On EMC Online Support, click **Support by Product.** Type `Solutions Enabler` in the "**Find a Product:** " search field and press **`Enter`**. The Solutions Enabler product page appears.
2. Click **Download** and then the platform-specific installation kit.
3. Save the installation kit to the host's disk drive and run the following command against it:

```
set file/attr=(RFM:FIX,LRL:32256) InstallKit
```

Step 2: Install the software

To install the software:

Procedure

1. Extract the command procedure after setting `[set DEF SYS$SYSDEVICE:[EMC.KITS]` by entering:

   ```
   backup/select=instcli.com InstallKit /sav instcli.com;
   ```

2. With both files (`instcli.com` and *InstallKit*) in the same temporary directory, run the installation procedure by entering:

   ```
   @instcli.com
   ```

3. At the following prompt, specify whether to allow lower privileged users to execute `sym*` commands.

   ```
   Do you want to enable lower privilege user capability?
   ```

   A **[Y]**es response will enable lower privileged users to execute commands. Step 6 describes the privileges these users require.

4. At the following prompt, specify whether to use the default password for the lockbox. This prompt will not appear if the lockbox already exists. For detailed information on the lockbox, please refer to the *EMC VMAX Family Security Configuration Guide*.

   ```
   Do you want to use the default password for the lockbox?
   ```

   A **[Y]**es response will use the default password. A **[N]**o response will allow users to enter their own password.

   If **[N]**o response was entered, the following prompt will be displayed to allow the entry of a lockbox password:

   ```
   The Lockbox password must be at least 8 characters long,
   contain an uppercase character, contain an lowercase
   character, contain a numeric value and an special character (!
   @#%&). Enter lockbox password:
   ```

   The installation produces the following DCL command procedures:

   - `emc_cli.com` should be called by the system `login.com` or by each user's login procedure.
   - `emc_install_sys_specific.com` is generated to provide a way to install the data directories in the `sys$specific` directory on each node in a cluster. At this point in the installation, this DCL procedure has already been executed on the machine where Solutions Enabler was installed.

Note

After the installation, all the data files from the installation will be located in the `sys$specific:[emc.symapi]` directories. If there were data files located in a previous installation area, the following files will be copied from the previous installation area to the `sys$specific:[emc.symapi]` directories:

- The `config` directory files are copied from the previous installation area to the `sys$specific:[emc.symapi.config]` directory.
- The database file for the machine on which Solutions Enabler is being installed is copied from the previous installation area to the `sys $specific:[emc.symapi.db]` directory.
- The log directory files are copied from the previous installation area to the `sys$specific:[emc.symapi.log]` directory.

The previous installation area data files and directories will remain intact until all the nodes in a cluster have executed the `emc_install_sys_specific.com` at which time they could be deleted. Even though they remain intact they are not used by the just installed software.

5. Ensure that each SYMCLI user's login procedure calls the `emc_cli.com` procedure to establish their proper SYMCLI environment.
6. Each user must have the following privileges for the SYMCLI to properly function. Take care when granting these privileges.
   - NETMBX — Can create network device.
   - SYSLCK — Can lock system wide resources.
   - SYSNAM — Can insert in the system logical name table.
   - CMKRNL — Can change mode to kernel.

   In addition to the above privileges, users who will be installing and controlling the daemons, require the following privileges:

   - DIAGNOSE — Can diagnose devices.
   - PHY_IO — Can perform physical I/O.
   - SHMEM — Can create/delete objects in shared memory.
   - SYSPRV — Can access objects by way of system protection.
   - WORLD — Can affect other processes in the world.
   - Users with lower privileges require the EMCSERVERS right so they can run the `sym*` commands.
7. Set the following minimum process quotas for each user account:
   - FILLM:1000
   - BIOLM:300
   - DIOLM:300
   - ASTLM:500
   - ENQLM:4000
   - BYTLM:500000

- WSEXTENT:32768

8. You can use the following formulas to calculate an approximation of the WSdef and Pgflquo quotas you should use. Depending on the configuration, you may need to set these values higher. You should re-valuate these values if the configuration changes significantly.
   - For the WSdef quota, use the following formula:
     (B + ((S * SN) + (D * DN) + (V * VN) + (P * PN) + (H * HN) + (G * GN))
   - For the Pgflquo quota, use the following formula:
     (B + (S *SN) + (S * RN) + (D * DN) + (V * VN) + (P * PN) + (H * HN) + (G * GN))

   Where:

B	= Minimum base of 10000 pagelets.
S	= 14900 pagelets per array.
SN	= Number of locally attached arrays.
RN	= Number of remotely attached arrays.
D	= Two pagelets per disk.
DN	= Number of disks. This is the total number of devices when adding up single devices, RAID members, meta members, etc. that Solutions Enabler will see in all arrays attached to the host.
V	= One pagelet per volume.
VN	= Number of volumes. This is the number of OpenVMS volumes (1DGAxxxx as well as shadow volumes) that this host will see on all arrays visible to this host.
G	= 12 pagelets per group.
GN	= Number of groups. This is the total number of Solutions Enabler disk groups that Solutions Enabler will be able to see on all arrays connected to this host.
P	= One pagelet per physical disk.
PN	= Number of physical disks. This the total number of all

	devices on all the arrays attached to this host which Solutions Enabler will see.
H	= One pagelet per hyper volume.
HN	= Number of hyper volumes. This is the total number of hypers visible to Solutions Enabler on all arrays connected to this host.

9. The installation is complete. Go to eLicensing on page 122.

Installing Solutions Enabler on Solaris 11

Before you begin

For the Solaris installation methods provided below, a Solaris repository with the Solutions Enabler kit uploaded into it is required. To check the repository, use the following command:

```
#pkgrepo list -s /export/SolutionsEnabler
```

Example result:

```
PUBLISHER NAME                          O VERSION
emc.com application/EMC_SYMdse  8.3.0,5.11-2050.273:2015
emc.com application/EMC_SYMse   8.3.0,5.11-2050.273:2015
```

Oracle Solaris Zones have been integrated with the new IPS package management tools in Oracle Solaris 11. By default, commands such as `pkginfo` are not available in a local zone. Therefore, you have to install the `SUNWpkgcmds` package before installing Solutions Enabler on a non-global/local zone.

Install SUNWpkgcmds using the following command:

```
pkg install SUNWpkgcmds
```

To check the global and non-global zone configurations, use the `zoneadm list -icv` command.

- The status `installed` means the zone is created but not running.
- The status `running` means the zone is up and running.

Setup local repository

Procedure

1. Create a dedicated Oracle Solaris ZFS File System. It allows using technologies such as clones and snapshots, to easily manage data.

```
# zfs create rpool/export/SolutionsEnabler
```

```
root@speb204:~# zfs create rpool/export/SolutionsEnabler
```

2. Create the Oracle Solaris 11 Solutions Enabler repository.

```
# pkgrepo create /export/SolutionsEnabler
```

3. Populate the Oracle Solaris 11 Solutions Enabler repository with the contents of SolutionsEnabler kit .p5p format.

```
# pkgrecv -s se830_2026_15-SunOS11-ni.p5p -d \ /export/
SolutionsEnabler '*'
```

Example result:

```
Processing packages for publisher emc.com ...
Retrieving and evaluating 2 package(s)...
PROCESS                                 ITEMS    GET (MB)
SEND (MB)
Completed                                2/2 133.7/133.7
307.4/307.4
```

To list packages available in the repository, use the following command:

```
# pkgrepo list -s /export/SolutionsEnabler
```

Example result:

```
PUBLISHER NAME                            O VERSION
emc.com  application/EMC_SYMdse   8.3.0,5.11-2050.273:20150515
emc.com   application/EMC_SYMse   8.3.0,5.11-2050.273:20150515
```

4. As a result, Solutions Enabler IPS package is available in the repository. To verify this, use the following command:

```
#pkgrepo list -s /export/SolutionsEnabler
```

Example result:

```
PUBLISHER NAME                          O VERSION
emc.com   application/EMC_SYMdse
8.3.0,5.11-2050.273:20150515
emc.com   application/EMC_SYMse
8.3.0,5.11-2050.273:20150515
```

Setup the publisher

Procedure

1. Set the publisher with the following command:

```
# pkg set-publisher -p file:///export/SolutionsEnabler emc.com
```

Example result:

```
pkg set-publisher:
 Added publisher(s): emc.com
```

2. List publishers on host:

```
# pkg publisher
```

Example result:

```
PUBLISHER     TYPE     STATUS P LOCATION
solaris      origin   online F http://pkg.oracle.com/solaris/
release/
emc.com      origin   online F file:///export/
SolutionsEnabler/
#
```

3. The publisher is set successfully.

Installing Solutions Enabler IPS in Global Zone

Procedure

1. A Solaris repository with the Solutions Enabler kit uploaded into it is required. To check the repository, use the following command:

```
e.g: #pkgrepo list -s /export/SolutionsEnabler
```

Example result:

```
PUBLISHER NAME                        O VERSION
emc.com   application/EMC_SYMdse
8.3.0,5.11-2050.273:20150515
emc.com   application/EMC_SYMse
8.3.0,5.11-2050.273:20150515
```

2. To install the latest version kit from repository use the following command:

```
pkg install application/EMC_SYMse
```

To install a particular version from repository use:

```
pkg install application/EMC_SYMse@product_version application/
EMC_SYMdse@product_version
```

where *product_version* is the particular version that is to be installed from the repository, for example 8.3.0,5.11-2151.287.

3. To verify that Solutions Enabler kit is installed on Global Zone, use the following command:

```
#pkg list | grep EMC
```

Example result:

```
application/EMC_SYMdse (emc.com)   8.3.0-2026.23    i--
application/EMC_SYMse (emc.com)    8.3.0-2026.23    i--
```

Uninstalling Solutions Enabler IPS in Global Zone

Procedure

1. To uninstall Solaris11 SE kit on Global Zone, run the following command:

```
#pkg uninstall application/EMC_SYMse application/EMC_SYMdse
```

2. To verify the uninstallation, run the following command:

```
#pkg list | grep EMC
```

Installing Solutions Enabler IPS kit on Non-Global Zones

Procedure

1. Log in to Non Global Zone using one of the following ways:
   - If Non Global zone is configured with IP and network configuration, then login using putty session.
   - If Non Global Zone is not configured with IP and network configuration, then login to Global Zone first and then login to Non-Global Zone by using the command `zlogin <Non-Global Zone Name>`
2. To install the latest version kit from the repository on the Non-Global Zone, use the following command:

```
#pkg install application/EMC_SYMse
```

To install a particular version from repository, use:

```
# pkg install application/EMC_SYMse@product_version
application/EMC_SYMdse@product_version
```

where *product_version* is the particular version that is to be installed from the repository, for example 8.3.0,5.11-2151.287.

3. To verify that Solutions Enabler kit is installed on Non-Global Zone, use the following command:

```
#pkg list | grep EMC
```

Example result:

```
application/EMC_SYMdse (emc.com)   8.3.0-2026.23     i--
application/EMC_SYMse (emc.com)    8.3.0-2026.23     i--
```

Uninstalling Solutions Enabler on Solaris 11 in Non-Global Zone from Global Zone

Procedure

1. To uninstall Solaris 11 SE kit in Non-Global Zone from Global Zone, run the following command:

```
zlogin <Non-Global Zone Name>  pkg uninstall application/
EMC_SYMse application/EMC_SYMdse
```

2. To verify the uninstallation, run the following command:

```
zlogin <Non-Global Zone Name>  pkg list | grep EMC
```

Installing Solutions Enabler on Solaris 11 in Non-Global Zone from Global Zone

Procedure

1. To install the latest version kit from the repository on the Non-Global Zone, use the following command:

```
zlogin <Non Global Zone Name> pkg install application/
EMC_SYMse
```

To install a particular version from repository, use:

```
zlogin <Non-Global Zone Name> pkg install application/
EMC_SYMse@product_version application/
EMC_SYMdse@product_version
```

where *product_version* is the particular version that is to be installed from the repository, for example 8.3.0,5.11-2151.287.

2. To verify that Solutions Enabler kit is installed on Non-Global Zone, use the following command:

```
zlogin <Non Global Zone Name> pkg list | grep EMC
```

Example result:

```
application/EMC_SYMdse (emc.com)    8.3.0-2026.23    i--
application/EMC_SYMse (emc.com)     8.3.0-2026.23    i--
```

Installing Solutions Enabler on Solaris 11 in Non-Global Zone from Global Zone

Procedure

1. To install the latest version kit from the repository on the Non-Global Zone, use the following command:

```
zlogin <Non-Global Zone Name> pkg install application/
EMC_SYMse
```

   To install a particular version from repository, use:

```
zlogin <Non-Global Zone Name> pkg install application/
EMC_SYMse@product_version application/
EMC_SYMdse@product_version
```

   where *product_version* is the particular version that is to be installed from the repository, for example 8.3.0,5.11-2151.287.

2. To verify that Solutions Enabler kit is installed on Non-Global Zone, use the following command:

```
zlogin <Non-Global Zone Name> pkg list | grep EMC
```

   Example result:

```
application/EMC_SYMdse (emc.com)        8.3.0-2026.23
i--
application/EMC_SYMse (emc.com)         8.3.0-2026.23
i--
```

Upgrading SMI-S Provider

To upgrade SMI-S Provider:

Procedure

1. Stop **ECOM** service.
2. Make a backup of these folders:

   On Windows:

```
C:\Program Files\EMC\ECIM\ECOM\conf\cst
C:\Program Files\EMC\ECIM\ECOM\conf\ssl
```

   On Linux:

```
/opt/emc/ECIM/ECOM/conf/cst
/opt/emc/ECIM/ECOM/conf/ssl
```

3. Uninstall the existing version of SMI provider.
4. Install SMI Provider V8.3 with the Solutions Enabler V8.3 installer.

5. Replace the folders mentioned in Step 2 with the backup you made.
6. Start **ECOM** service.

Note

Affected platforms are: Windows 64-bit and Linux 64-bit.

Installing the Solutions Enabler Virtual Appliance

The Solutions Enabler Virtual Appliance is a VMware ESX server virtual machine that provides all the components you need to manage your storage environment using the storsrvd daemon and Solutions Enabler network client access.

For detailed installation steps on the Solutions Enabler Virtual Appliance, please refer to the *Solutions Enabler Virtual Appliance Installation Guide*.

CHAPTER 3

UNIX Native installation

This chapter describes how to install/upgrade Solutions Enabler using UNIX PureNative installation kits.

Before you begin

Before you begin to install/upgrade Solutions Enabler, be sure to complete the tasks listed in this section.

Procedure

1. Review the following best practices:
   - Backup persistent data and uninstall previous versions of Solutions Enabler before performing major upgrades.
   - Use the response file method for mass deployments.
   - The automated installers: Kickstart, Jumpstart, and Ignite are recommended.
   - To achieve full installation functionality, use the Solutions Enabler installation wrapper script.
2. For AIX and Solaris hosts with GPG installed, import the public key and verify the digital signature:
   a. Locate the public key (public_key) and the signature. For example, the digital signature for AIX is:

      `SYMCLI.8.3.0.0.bff.sig`

   b. Import the key, by entering:

      gpg --import public_key

   c. Verify the imported key using, by entering:

      -bash-3.00# gpg --list-key

   d. Edit the imported key and trust it ultimately, by entering:

      -bash-3.00# gpg --edit-key C4E34013

   e. Verify the digital signatures, by entering:

      `gpg --verify` *SigFile*

      Where *SigFile* is the name of the digital signature.

      For example, to verify the digital signature for AIX, enter:

      gpg --verify SYMCLI.8.3.0.0.bff.sig

   f. For Linux hosts, import the ascii public key, by entering:

      rpm --import sepubkey.asc

PureNative installation kits

Solutions Enabler PureNative kits are available for the following UNIX platforms:

- AIX
- HP-UX (PA/RISC and ia64)
- Linux (ia64, PPC64, and 390)

- Solaris (SunOS Sparc and SunOS x86)

The kits use the following naming convention:

```
seMmPp-OS-ARCH-ni.tar.gz
```

Where:

M = Major version

m = Minor version

P = Point

p = Patch

OS = Operating System

ARCH = Processor architecture

For example:

```
se8300-SunOS-sparc-ni.tar.gz
```

Table 17 on page 101 lists the kit components by operating system.

Note

N/A indicates that the component is not supported in the corresponding operating system. Components within shaded rows are required.

Table 17 Solutions Enabler PureNative kit contents

OS-specific component names				Description
AIX	**HP-UX**	**Linux**	**SunOS**	
SYMCLI.DATA.rte	SYMCLI.DATA	symcli-data	SYMdse	Installs persistent data files and SSL certificate files.
			SYMse	Installs Solutions Enabler program files for Solaris platforms (sparc and X86). This holds sub components like SRM, JNI, etc.
SYMCLI.THINCORE.rte	SYMCLI.THINCORE	symcli-thincore	N/A	Installs Solutions Enabler thin core functionality.
SYMCLI.BASE.rte	SYMCLI.BASE	symcli-base	N/A	Installs: • Solutions Enabler core functionality,

Table 17 Solutions Enabler PureNative kit contents (continued)

OS-specific component names				Description
AIX	**HP-UX**	**Linux**	**SunOS**	
				including `symapi`, `symlvm`, `storapi`, `storapid`, `storcore`, `stordaemon`, and `storpds` • Storage Resource Management base mapping library • Shared libraries and runtime environment, including Base Storage LIbrary component and Control Storage Library component This option is part of the shared library runtime environment. It is a core requisite for other options, and is therefore mandatory for a successful installation.
SYMCLI.CERT.rte	SYMCLI.CERT	symcli-cert	N/A	Installs SSL certificate files.
SYMCLI.SYMCLI.rte	SYMCLI.SYMCLI	symcli-symcli	N/A	Installs the collection of binaries known as Symmetrix Command Line

Table 17 Solutions Enabler PureNative kit contents (continued)

OS-specific component names				Description
AIX	**HP-UX**	**Linux**	**SunOS**	
				Interface (SYMCLI).
SYMCLI.SYMRECOVER.rte	SYMCLI.SYMRECOVER	symcli-symrecover	N/A	Installs the SRDF session recovery component.
N/A	N/A	symcli-smi	N/A	Installs the SMI Provider.
N/A	N/A	symcli-vss	N/A	Installs the VSS Provider.
SYMCLI.SRM.rte	SYMCLI.SRM	symcli-srm	N/A	Installs: • The shared libraries and runtime environment - base mapping component. • The Oracle daemon. • The SRM SYBASE database runtime component. • The SRM IBM UDB database runtime component.
SYMCLI.JNI.rte	SYMCLI.JNI	symcli-jni	N/A	Installs the Solutions Enabler Java interface component. You should install this component if your Solutions Enabler installation uses the Java interface.
SYMCLI.64BIT.rte	SYMCLI.64BIT	symcli-64bit[a]	N/A	Installs the 64-bit libraries.

a. Only for Linux X64.

Installing Solutions Enabler

This section describes how to install/upgrade Solutions Enabler using native installer commands.

Installing on AIX

To install on an AIX host:

Procedure

1. Uncompress and untar the installation kit.
2. Do either of the following depending on whether you want to perform a full or customized installation:
   - To perform a full installation, run the following command:

     ```
     installp -ac -d absolute_path_to_SYMCLI*.bff_file all
     ```

   - To perform a custom installation and install only specific components, run the following command:

     ```
     installp -a -d absolute_path_to_SYMCLI*.bff_file FileSetName
     ```

     Where *FileSetName* is a component name from Table 17 on page 101.
3. Run the following command to verify the component installation:

   ```
   lppchk -f FileSetName
   ```

   A `0` value is returned for a successful installation.
4. Repeat steps 2 and 3 for each component to install.

Installing on HP-UX

You can install Solutions Enabler on a HP-UX host using either a command line option or a response file.

Using the command line

To install on an HP-UX host using the command line:

Procedure

1. Uncompress and untar the installation kit.
2. From the local file system, run the following commands to start the installation:

   ```
   swreg -l depot AbsolutePathtoSYMCLI.depot
   ```

   ```
   swinstall -s AbsolutePathtoSYMCLI.depot
   FileSetName:InstallPath
   ```

   Where *FileSetName* is a component name from Table 17 on page 101.

3. Repeat step 2 for each component to install.

Using a response file

To install on an HP-UX host using a response file:

Procedure

1. Create a response file similar to the following:

```
#cat response_file_bin
SYMCLI.THINCORE:/opt/emc
SYMCLI.BASE:/opt/emc
SYMCLI.SRM:/opt/emc
SYMCLI.SYMCLI:/opt/emc
SYMCLI.SYMRECOVER:/opt/emc
SYMCLI.JNI:/opt/emc
SYMCLI.64BIT:/opt/emc

#cat response_file_data
SYMCLI.DATA:/usr/emc
SYMCLI.CERT:/usr/emc
```

2. Run the following command, specifying the location of the installation package and the name of your response file:

```
swinstall -s AbsolutePathtoSYMCLI.depot -f ResponseFile
```

Installing on Linux

You can install Solutions Enabler on a Linux host using either RPM, or a response file.

Using RPM

To install on a Linux host using the command line:

Procedure

1. Uncompress and untar the installation kit.
2. Run the following command to start the installation:

```
rpm -i symcli*8.3.0*.rpm
```

3. Run the following command to verify the component installation:

```
rpm -qa | grep symcli
```

4. Run the following command to verify the component installation:

```
rpm -i symcli*8.3.0*.rpm
```

5. Run the following command to set lockbox password:

```
/usr/symcli/install/set_lockbox.sh
```

Using a response file

To install on a Linux host using a response file:

Procedure

1. Create a response file similar to the following in `/usr/temp/emc_se_linux_response_file`:

```
-bash-2.05b# cat emc_se_linux_response_file
EMC_APPLICATION_PATH:/opt/emc
EMC_VAR_PATH:/usr/emc
ADDITIONAL_COMPONENTS:jni srm
```

2. Run the following command to start the installation:

```
rpm -i symcli*8.3.0*.rpm
```

3. Run the following command to verify the installation:

```
rpm -qa | grep symcli
```

Installing on Solaris

You can install/upgrade Solutions Enabler on a Solaris host using either a command line option, or a response file.

Using the command line

To install on a Solaris host using the command line:

Procedure

1. Uncompress and untar the installation kit.
2. Run the following command to view a list of packages:

```
pkgadd -d .
```

3. Run the following, depending on whether you want to start an interactive or silent installation:

Interactive:	pkgadd -d . *PkgName* pkgadd -G -d . *PkgName* (on Solaris 10 or higher)
Silent:	pkgadd -n -d . -a *Full_path_to_ADMINFile* -r *ResponseFile PkgName* pkgadd -G -n -d . -a *Full_path_to_ADMINFile* -r *ResponseFile PkgName* (on Solaris 10 or higher)

Where *ResponseFile* is the name of your response file and *PkgName* is a component name from Table 17 on page 101.

The Solutions Enabler Solaris installation kit consists of two components: SYMdse and SYMse. SYMdse contains persistent data files and SYMse contains program files. SYMse accommodates classes (sub components), which are used

to custom-install required Solutions Enabler features like SRM, JNI, etc., using a response file.

Install the components in the following order:

a. `SYMdse`

b. `SYMse`

4. Run the following command to verify the installation:

```
pkgchk -f PkgName
```

A `0` value is returned for a successful installation.

5. Repeat steps 3 and 4 for each component to install.

Using a response file

To install on Solaris host using a response file:

Procedure

1. Uncompress and untar the installation kit.
2. Create a response file similar to the following:

```
-bash-2.05b# cat response_file_bin
CLASSES=none thincore base symcli symrecover srm 64bit jni
BASEDIR=/opt/emc
```

```
-bash-2.05b# cat response_file_data
CLASSES=none data cert
BASEDIR=/usr/emc
```

3. Create the following admin file:

```
#cat admin_file
mail=
basedir=default
runlevel=quit
conflict=nocheck
setuid=nocheck
action=nocheck
partial=nocheck
instance=overwrite
idepend=quit
rdepend=quit
space=quit
```

4. Run the following command to start the installation:

```
pkgadd -n -d . -a Full_path_to_ADMINFile -r ResponseFile
PkgName
```

```
pkgadd -G -n -d . -a Full_path_to_ADMINFile -r ResponseFile
PkgName (on Solaris 10 or higher)
```

Where *ResponseFile* is the name of your response file and *PkgName* is a component name from Table 17 on page 101.

5. Install the components in the following order:
   a. data
   b. cert
   c. thincore
   d. base
   e. symcli
   f. symrecover
   g. srm
   h. 64bit
   i. jni

   Note

   For component descriptions, refer to Table 17 on page 101.

6. Run the following command to verify the installation:

   ```
   pkginfo
   ```

7. Repeat steps 2 through 6 for each component to install.

Uninstalling Solutions Enabler

This section describes how to uninstall Solutions Enabler using native installer commands.

Uninstalling from AIX

To uninstall from an AIX host, run the following command:

```
installp -u FileSetName
```

Where

FileSetName is a component name from Table 17 on page 101.

Uninstalling from HP-UX

To uninstall from an HP-UX host, run the following command:

```
swremove FileSetName
```

Where

FileSetName is a component name from Table 17 on page 101.

Uninstalling from Linux

To uninstall from a Linux host, run the following command:

```
rpm -e 'rpm -qa |grep -i symcli'
```

Uninstalling from Solaris

To uninstall from a Solaris host, run the following, depending on whether you want to start an interactive or silent uninstall:

Interactive:	pkgrm *PkgName*
Silent:	pkgrm -n -a *Full_path_to_ADMINFile PkgName*

Where

PkgName is a component name from Table 17 on page 101.

CHAPTER 4

Uninstalling Solutions Enabler

This chapter explains how to uninstall Solutions Enabler:

Overview

To uninstall Solutions Enabler from a UNIX host, you must first shutdown the application processes that use the Solutions Enabler libraries and binaries, and then uninstall the software.

Note

This is not necessary on Windows hosts since the uninstall program will prompt you to shut down the application processes. If you are uninstalling from a Windows host, skip this step and go to Uninstalling Solutions Enabler from Windows on page 116.

Stopping the application processes

To stop the application processes:

Procedure

1. For UNIX, issue the following command to identify any applications using the Solutions Enabler libraries:

   ```
   fuser /usr/symcli/shlib/libsym* /usr/symcli/shlib/libstor*
   ```

   For AIX, issue:

   ```
   fuser -x -f /usr/symcli/shlib/library_name
   ```

2. Issue the following command to stop the Solutions Enabler daemons:

   ```
   stordaemon shutdown all
   ```

   Note

   For more information on this command, refer to Stopping daemons on page 142.

3. Issue the following command to verify that the daemon(s) have stopped:

   ```
   stordaemon list -running
   ```

   Note

   For more information on this command, refer to Viewing daemons on page 142.

Uninstalling the software

To uninstall the Solutions Enabler software, refer to the following:

- For UNIX, refer to Uninstalling Solutions Enabler from UNIX on page 113.
- For Windows, refer to Uninstalling Solutions Enabler from Windows on page 116
- For OpenVMS, refer to Uninstalling Solutions Enabler from OpenVMS on page 118.

Uninstalling Solutions Enabler from UNIX

You can uninstall Solutions Enabler from a UNIX host using either the Solutions Enabler uninstall script or your native install tools (for example, `rpm --erase` on Linux).

⚠ CAUTION

Take care when removing Solutions Enabler, as it may be a prerequisite for other installed products.

Using the script

To use the script to uninstall Solutions Enabler from all supported UNIX hosts, change directory to `/usr/symcli/install` and run the following script:

```
./se8300_install.sh -uninstall
```

For help running the uninstall script, run the following script:

```
./se8300_install.sh -help
```

The uninstall script creates log files in the install root directory `/opt/emc/logs` in the format `SE_NI_`*`KitVersion_TimeStamp`*`.log`, where *TimeStamp* is in the form *YYMMDD_HHmmSS*.

Persistent data

The persistent data will remain under `/usr/emc/API/symapi` or in the data directory selected during installation.

The persistent data will remain accessible from the softlink `/var/symapi`.

Decremental method

To uninstall a single Solutions Enabler component you can use the `-decrement` option:

```
./se8300_install.sh -decrement [-cert][-jni] [-srm] [-symrec]
```

Note

This method is not supported on Solaris.

For example, to uninstall the Solutions Enabler SYMRECOVER component, enter:

```
./se8300_install.sh -decrement -symrec
```

Using native tools

When using your native tools to uninstall Solutions Enabler, you must uninstall the Solutions Enabler packages in the following order:

Table 18 Package order when uninstalling using UNIX native tools

Order	Solaris	For all other UNIX operating systems
1	SYMse	SMI
2	SYMdse	64BIT
3		SRM
4		JNI
5		SYMRECOVER
6		SYMCLI
7		BASE
8		THINCORE
9		DATA
10		CERT

In addition, you must also verify that all application processes using the Solutions Enabler libraries and binaries are stopped. For instructions, refer to Stopping the application processes on page 112.

Uninstalling from Linux

Use the following commands when uninstalling Solutions Enabler from a Linux host:

```
rpm -qa|grep symcli
```

Lists all of the installed RPMs.

```
rpm -ql <RPM entry from the installed list>
```

Lists all of the files in the specified RPM. For example, to list all of the files in the core component, enter:

```
rpm -ql symcli-thincore-8.3.0.1701-116.1
rpm -e <RPM entry from the installed list>
```

Uninstalls the specified RPM. For example, to uninstall the core component, enter:

```
rpm -e symcli-thincore-8.3.0.1701-116.1
```

Uninstalling from AIX

Use the following commands when uninstalling Solutions Enabler from an AIX host:

```
lslpp -L | grep SYMCLI
```

Lists all installed Solutions Enabler filesets.

```
installp -u FilesetName
```

Uninstalls a fileset. For example, to uninstall the core component, enter:

```
installp -u SYMCLI.THINCORE
```

Uninstalling from HPUX

Use the following commands when uninstalling Solutions Enabler from an HPUX host:

```
swlist -l  fileset | grep SYMCLI
```

Lists all of the installed Solutions Enabler filesets.

```
swremove FilesetName
```

Uninstalls a fileset. For example, to uninstall the Solutions Enabler core component, enter:

```
swremove SYMCLI.THINCORE
```

Uninstalling from Solaris

Use the following commands when uninstalling Solutions Enabler from a Solaris host:

```
pkginfo  | grep SYM
```

Lists all of the installed Solutions Enabler packages.

```
pkgrm PackageName
```

Uninstalls a package. For example, to uninstall the Solutions Enabler SYMse component, enter:

```
pkgrm SYMse
```

Uninstalling Solutions Enabler from Windows

This section describes the various methods available for uninstalling Solutions Enabler from a Windows host.

⚠ CAUTION

Take care when removing Solutions Enabler, as it may be a prerequisite for other installed products.

Using the InstallShield wizard

To uninstall Solutions Enabler using the InstallShield wizard:

Procedure

1. Change the directory to the location of the Solutions Enabler kit by entering the following:

   ```
   cd \Install_disk_mount_point\Windows
   ```

2. Start the uninstall by running the following:

   ```
   se8300-WINDOWS-x64.exe
   ```

3. In the **InstallShield Wizard for Solutions Enabler Welcome** dialog box, click **Next**.
4. In the **Program Maintenance** dialog box, select **Remove** and click **Next**.
5. In the **Remove the Program** dialog box, click **Remove**.
6. In the **Installation Program Complete** dialog box, click **Finish** to complete the removal process.

Using the command line

To uninstall Solutions Enabler from the command line using the msi installer options, run the following command:

```
start /wait FullPathToInstallImage\
se8300-WINDOWS-x64.exe /s /x /v/qn
```

Where:

FullPathToInstallImage is the path to the executable.

`/s` is the command to run silently.

`/x` is the command to uninstall.

`/v` is the command gateway for `msiexec.exe`.

`/qn` is the silent option.

Note

If the `/s` and `/v` options are entered as capital letters (`/S /V`), and a space is used to separate the `/v` and `/qn` options, the uninstallation starts in Wizard mode.

Removing the msi image

You can use either of the following methods to uninstall the msi image:

Procedure

1. Enter the following command, specifying the GUID of the product to uninstall:

   ```
   start /wait msiexec.exe /x {GUID} /qn
   ```

   Possible values for *GUID* are:

   `{A2A6F36B-9F18-41fe-BCA1-FECF2DE9F5BC}` Solutions Enabler

   `{CA1446F4-FF86-46ff-9783-C9E1AF21FE5E}` STORBLK

   `{DFFEB2C8-5442-45e2-B2E1-9D90AF84BCF5}` SDK

   `{E4D9E227-D0F3-4666-8BEF-34C577CE562B}` TCLIENT

2. Use the Windows Installer Clean Up utility, `msicuu2.exe`:
   a. Download the `msicuu2.exe` from Microsoft and install it on the host.
   b. From the Windows **Start** menu, select **All Programs**.
   c. Select the application to remove and click **Remove**.
   d. Stop the following services in the order listed below. You can do this from either the cmd prompt or the **Services** dialog.

      Storsrvd

      Storgnsd

      Storrdfd

      Storevntd

      Storsrmd

      Storstpd

      Stororad

      Storsqld

      Storudbd

      Storapid

      ECOM

      slpd

   e. Remove the list of files from System32. The list of files is the same as those in *InstallDir*`\Symcli\shlib`.
   f. Remove the `Symcli` directory and all its subdirectories.

g. Remove the `ECOM` directory and all its subdirectories.

h. Remove the subdirectories from `Symapi` , except for the `Config` and `db` directories.

i. Remove the following registry entries:

HKEY_LOCAL_MACHINE\SOFTWARE\EMC\EMC Solutions Enabler

HKEY_LOCAL_MACHINE\SOFTWARE\EMC\SYMCLI

HKEY_LOCAL_MACHINE\SOFTWARE\EMC\WideSky

j. From under the following registry key, remove the entries that only point to the SYMAPI or SYMCLI:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows \CurrentVersion\SharedDlls

Using the Windows Add/Remove Programs dialog

To uninstall Solutions Enabler from the Windows **Add or Remove Programs** dialog:

Procedure

1. From the Windows **Start** menu, select **Settings | Control Panel | Add or Remove Programs**.
2. In the **Add or Remove Programs** dialog, select **EMC Solutions Enabler** and click **Uninstall**.

Using the Windows Programs and Features dialog

To uninstall Solutions Enabler from the Windows **Programs and Features** dialog:

Procedure

1. From the Windows **Start** menu, select **Control Panel**.
2. Click **Programs and Features**.
3. Under **Programs** , click **Uninstall a Program**.
4. Select **EMC Solutions Enabler** and click **Uninstall**.

Uninstalling Solutions Enabler from OpenVMS

To uninstall Solutions Enabler from an OpenVMS host:

⚠ CAUTION

Take care when removing Solutions Enabler, as it may be a prerequisite for other installed products.

Procedure

1. Verify that all application processes that use the Solutions Enabler libraries and binaries are stopped.
2. If file `emc$root:[-]emc_disable_autostart.com` exists then execute the following: `@emc$root:[-]emc_disable_autostart.com`

3. Delete all the files in the `sys$specific:[emc]` and `sys$specific:[000000]emc.dir` directories. If the environment is a cluster, delete these files from every node in the cluster where Solutions Enabler was running.
4. Delete all the files from the installation directory.

Uninstalling Solutions Enabler from z/OS

To uninstall Solutions Enabler from a z/OS host:

Procedure

1. Verify that all jobs or Started Tasks that use the Solutions Enabler datasets are stopped.
2. Delete the installation datasets as required.
3. Delete Unix System Services files (if required).

Note

This may be `/var/symapi` or a different directory, depending on the choices made for the install job #07dflts.

Rolling back an upgrade

To roll back your upgrade, you must have created copies of the host database and config directories, as explained in Before you begin on page 20:

Procedure

1. Verify that all application processes that use the Solutions Enabler libraries and binaries are stopped.

Note

For instructions, refer to Stopping the application processes on page 112.

2. Export all device groups from the current SYMAPI database:
   a. Issue a `symdg list` command to list all the device groups.
   b. Issue a `symdg export` command to export the device groups.
   c. Issue a `symcg list` command to list all the composite groups.
   d. Issue a `symcg export` command to export the composite groups.

Note

This export is necessary because older versions of Solutions Enabler may not be able to read a database once a newer version of Solutions Enabler has converted it.

Note

For more information on these commands, refer to the *EMC Solutions Enabler Array Management CLI User Guide*.

3. Uninstall your software according to the platform-specific procedures earlier in this chapter.
4. Install the desired version of Solutions Enabler.
5. Once the installation is complete, issue a `symcfg list` command to verify that the SYMAPI database can be used by the older version:
   a. If the database can be used, the rollback is done.
   b. If the database cannot be used, issue a `symcfg discover` command to create an array host database file, `symapi_db.bin`, and import all the exported device groups.

CHAPTER 5

Post-Installation configuration for UNIX, Windows, OpenVMS, and z/OS

After you have installed Solutions Enabler, you need to perform certain follow-up procedures to enable your software's features and to establish your command environment. This chapter provides the follow-up procedures for a Solutions Enabler installation in UNIX, Windows, OpenVMS, and z/OS environments:

Note

As an alternative to the in-depth UNIX and Windows procedures in this chapter, Installation checklist on page 44 provides operating-system-specific checklists with high-level installation and configuration steps that advanced users may find useful.

---

eLicensing

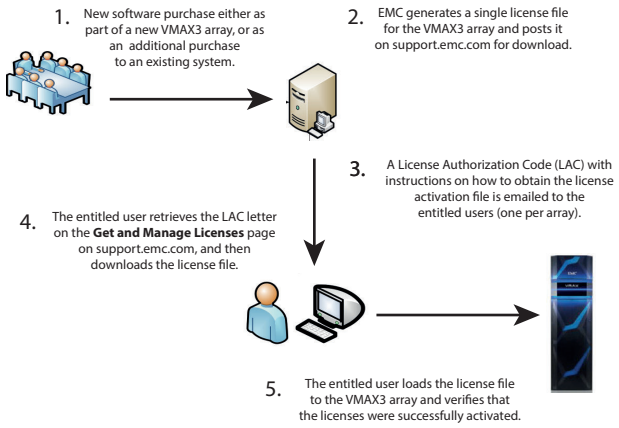
VMAX3 arrays use Electronic Licenses (eLicenses).

Note

For more information on eLicensing, refer to EMC Knowledgebase article 13866 on the EMC Online Support website.

You obtain license files from EMC Online Support, copy them to a Solutions Enabler or a Unisphere for VMAX host, and push them out to your arrays. The following figure illustrates the process of requesting and obtaining your eLicense.

Figure 6 Requesting and obtaining licenses



Each license file fully defines all of the entitlements for a specific system, including the license type and the licensed capacity. To add a feature or increase the licensed capacity, obtain and install a new license file.

Most VMAX3 array licenses are array-based, meaning that they are stored internally in the system feature registration database on the array. However, there are a number of licenses that are host-based.

Array-based eLicenses are available in the following forms:

- An individual license enables a single feature.
- A license suite is a single license that enables multiple features. License suites are available only if all features are enabled.
- A license pack is a collection of license suites that fit a particular purpose.

For details on the available license packages, their contents, and capacity measurement information, please refer to the *EMC VMAX3 Family Product Guide for VMAX 100K, VMAX 200K, VMAX 400K with HYPERMAX OS* and the *EMC Symmetrix VMAX Family with Enginuity Product Guide for VMAX 10K, VMAX 20K, VMAX 40K.*

Upgrade to an eLicensed array

When upgrading from a non-eLicensed array to an eLicensed array, the system is scanned for OS features currently in use that require eLicenses.

If OS features are found in use, and there are no registered and applied eLicenses, they are reported as “IN USE,” which allows continued access to the features while reporting that these features require proper licensing to ensure compliance. By only

reporting this information, it prevents disruption to normal operations of your system and business.

If your eLicensing report shows one or more OS features as “IN USE,” it is your responsibility to work with your EMC Sales team to obtain proper eLicensing for those features.

Host-based licenses

Most VMAX array licenses use the array-based model. However, there are still a number of licenses that remain host-based.

Note

The process for obtaining the remaining host-based licenses will remain the same as with previous versions of Solutions Enabler.

Note

Management of VMAX arrays requires Solution Enabler license keys but no license keys are required for using VSS Provider V8.3.

Table 19 on page 123 lists the host-based licenses that remain unchanged on Enginuity 5876 or lower.

Table 19 Host-based licenses unchanged, regardless of Enginuity level

License/Description	Commands included
FAST for DMX (full device only)	N/A. This feature is only available with Unisphere for VMAX.
TimeFinder (all, including TimeFinder/Mirror)	symioctl symmir symreturn

Table 20 on page 123 lists the host-based licenses required to perform operations on VMAX arrays running Enginuity versions lower than 5876 from a Solutions Enabler V8.3 host.

Table 20 Host-based licenses required for Enginuity versions lower than 5876

License	Commands included
Dynamic Cache Partitioning	symqos -cp
FAST	symfast symtier
Optimization	symmigrate symoptmz
Open Replicator/DM	symrcopy

Table 20 Host-based licenses required for Enginuity versions lower than 5876 (continued)

License	Commands included
SRDF	symrdf add RDF group symconfigure add RDF mirror symconfigure create SAVE devices symconfigure set dynamic RDF attribute
SRDF/Async	symrdf set mode async symconfigure SRDF/A settings and add RDF mirror symrdf create dynamic pair in asynchronous mode
SRDF/Star	symstar[a]
SRDF/Synchronous	symconfigure add rdf mirror symrdf create dynamic pair in synchronous mode
Symmetrirx Priority Control	symqos -pst
TimeFinder/Clone	symclone and symmir (using clone emulation)
TimeFinder/Snap	symsnap symconfigure create snap pool and SAVE devices

a. Also requires SRDF/A and SRDF/S licenses.

Managing arrays running different Enginuity versions

The operations that you can perform from a host are based on the host-based licenses in the host's `symapi_licenses.dat` file, if any, and the array-based licenses in the array's feature registration database (Enginuity 5876 or higher).

Note

The location of this `symapi_licenses.dat` file varies according to the operating system. For more information, refer to Solutions Enabler Directories on page 263.

The remainder of this section describes how the operations you can perform from a Solutions Enabler host are determined when accessing various Enginuity versions.

Solutions Enabler V7.6 (or higher) host

When accessing an array running Enginuity 5876 or higher from a host running Solutions Enabler V7.6 or higher, the operations you can perform on the array are based on:

- The licenses in the array's feature registration database.

- The licenses in the host's `symapi_licenses.dat` file, if using any of the host-based features listed in Table 19 on page 123.

When accessing an array running Enginuity version 5773 from the same host, the operations you can perform on the array are based on the licenses in the host's `symapi_licenses.dat` file, if using any of the host-based features listed in Table 19 on page 123 and Table 20 on page 123. If not, you can only perform operations that do not require a license.

When accessing an array upgraded to Enginuity 5876 or higher from a host upgraded to Solutions Enabler V7.6 or higher, any product title that you were currently using will still function (even if it does not have an entitlement). However, to use any of the new Enginuity 5876 product titles or any of the older product titles you were not using, you must obtain and install an array-based license file on the array. Installing array-based licenses on page 125 describes how to install license files.

Installing array-based licenses

This section explains how to use the `symlmf add` command to install array-based licenses.

Note

Installing licenses requires an authorization role of Storage Admin or higher.

You can only install array-based licenses from a host running one of the following operating systems:

- Windows: AMD64
- Linux: AMD64, ia64
- Solaris: 64 bit (Sparc)
- HP-UX 11.21: ia64
- AIX 5.3 and 6.1: PPC 64

For instructions on installing from a host running a supported operating system, refer to Installing from a supported host on page 125. For instructions on installing from a host running a non-supported operating system, refer to Installing using alternative methods on page 126.

Note

To obtain array-based licenses from EMC Online Support you will need the License Authorization Code (LAC) identification number from the LAC letter e-mailed to you.

Installing from a supported host

To install an array-based license file from a host running a supported operating system:

1. Obtain a license file from EMC Online Support and copy it to your host.
2. Use the following `symlmf` command to push the license file to the VMAX array:

```
symlmf add -type emclm -sid SymmID -file FileName -v
```

Where:

SymmID — Specifies the array on which you are installing the license file.

FileName — Specifies the name of the license file.

Output similar to the following appears:

```
License SYMM_VMAX_SPC 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_DCP 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_FAST_VP 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_FAST 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_OPTIMIZER 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_TF_SNAP 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_TF_CLONE 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_SRDF_STAR 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_SRDF_S 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_SRDF_A 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_SRDF 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_ENGINUITY 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_OR-DM 000000001234 15-Jan-2014: Processed successfully
License SYMM_VMAX_SMC 000000001234 15-Jan-2014: Processed successfully
Total Licenses Processed:         13
Total host-based eLicense ignored: 0
Total Licenses Not Processed:      0
```

Note

Issuing the `add` command without the `-v` option will eliminate all but the last three lines of the above output.

Installing using alternative methods

To install an array-based license file from a host running a non-supported operating system, use one of the following methods:

- Run `symlmf` directly on the VMAX service processor. This method requires that you contact EMC Customer Support.
- Run `symlmf` on one of the unsupported platforms via client/server to a SYMAPI server on one of the supported platforms.

Installing host-based licenses

Note

Installing licenses requires an authorization role of Storage Admin or higher.

To install a host-based license:

Procedure

1. Use the following `symlmf` command to install a license key on a host:

   ```
   symlmf add -type se -license LicenseNumber
   ```

2. Use the following command to list the licenses installed on the host:

   ```
   symlmf list -type se
   ```

Displaying licenses

The procedures in this section explain how to use the `symlmf list` command to display installed licenses.

Note

For field descriptions of the output examples in this section, refer to symlmf list output field descriptions on page 129.

Displaying array based licenses

To display the current array based licenses activated by a license file, use the following command:

```
symlmf list -type emclm -sid SymmID
```

Output similar to the following appears:

```
Symmetrix ID : 000000001234
Issue Date   : 03/22/2015
                              Activation        Capacity            Install
Name .......................Type      ID     Type          Licensed    Date
--------------------------- ----- ---------- ------------- -------- ----------
Foundation_Suite............ P-IND 111111111  Usable-TB..... 500 09/13/2014
Remote_Replication_Suite... P-IND 1234567    Usable-TB.....500 09/13/2014

Legend:
  Activation Type:
    E-IND = Evaluation Individual
    P-IND = Permanent Individual
    P-ENT = Permanent Enterprise Agreement
```

If individual licenses had been purchased, output similar to the following appears:

```
Symmetrix ID : 000194901138
Issue Date   : 03/22/2015
                              Activation        Capacity            Install
Feature Name                Type      ID     Type          Licensed    Date
--------------------------- ----- ---------- ------------- -------- ----------
SYMM_VMAX_ENGINUITY         P-IND 102938475  R-TB-Non-SATA      100 08/22/2014
                                             R-TB-SATA          500
SYMM_VMAX_FAST              P-IND 1234567    Reg-TB              60 08/22/2014
SYMM_VMAX_OR_DM             P-IND 1234567    Reg-TB              10 08/22/2014
SYMM_VMAX_PROSPHERE         P-IND 1234567    R-TB-Non-SATA      100 08/22/2014
                                             R-TB-SATA          500
SYMM_VMAX_SMC               P-IND 1234567    R-TB-Non-SATA      100 08/22/2014
                                             R-TB-SATA          500
SYMM_VMAX_SRDF              P-IND 1234567    Reg-TB              30 08/22/2014
SYMM_VMAX_SRDF_S            P-IND 1234567    Reg-TB              20 08/22/2014
SYMM_VMAX_SRDF_STAR         P-IND 1234567    Reg-TB              40 08/22/2014
SYMM_VMAX_TF_CLONE          P-IND 1234567    Reg-TB              50 08/22/2014

Legend:
  Activation Type:
    E-IND = Evaluation Individual
```

```
    P-IND = Permanent Individual
    P-ENT = Permanent Enterprise Agreement
```

In addition, you can also add the `-output xml_element` option to the above command to produce an XML report containing the same information. For example:

```
symlmf list -type emclm -sid SymmID -output xml_element
```

Displaying host and array-based licenses

To display the host-based and array-based licenses that apply to VMAX arrays, use the following command:

```
symlmf list -type sym -sid 1234
```

Output similar to the following appears:

```
Symmetrix ID: 000000001234
                                                Capacity
Feature Name                  Lic  Type                Units
---------------------------- ----- ------------------ --------
SYMM_UNPROT_SDR              SE    N/A                       -
SYMM_VMAX_ENGINUITY          EMCLM R-TB-Non-SATA           100
                                   R-TB-SATA               500
SYMM_VMAX_FAST_TIERING       EMCLM R-TB-Non-SATA           100
                                   R-TB-SATA               500
                                   R-TB-EXTERNAL           300
SYMM_VMAX_OR_DM              EMCLM R-TB-Non-SATA           100
                                   R-TB-SATA               500
                                   R-TB-EXTERNAL           300
SYMM_VMAX_PROSPHERE          EMCLM R-TB-Non-SATA           100
                                   R-TB-SATA               500
                                   R-TB-EXTERNAL           300
SYMM_VMAX_SMC                EMCLM R-TB-Non-SATA           100
                                   R-TB-SATA               500
                                   R-TB-EXTERNAL           300
SYMM_VMAX_SRDF_REPLICATION   EMCLM R-TB-Non-SATA           100
                                   R-TB-SATA               500
                                   R-TB-EXTERNAL           600
SYMM_VMAX_SRDF_STAR          EMCLM R-TB-Non-SATA           100
                                   R-TB-SATA               500
                                   R-TB-EXTERNAL           300
SYMM_VMAX_TIMEFINDER         EMCLM R-TB-Non-SATA           100
                                   R-TB-SATA               500
                                   R-TB-EXTERNAL           300

Legend:
  Lic(ense Type):
    EMCLM = emclm license
    SE    = se license
```

In addition, you can also add the `-output xml_element` option to the above command to produce an XML report containing the same information. For example:

```
symlmf list -type sym -sid SymmID -output xml_element
```

symlmf list output field descriptions

The following explains the output for the `symlmf list` command:

- **Activation ID**: Activation ID assigned to the license.
- **Activation Type**: The feature's license can be assigned to:
  - **Ind**(ividual) storage arrays,
  - Individual storage arrays but with a limited **Eval**(uation) time period, or to
  - All the storage arrays in the **Ent**(erprise).
- **Capacity Licensed**: The maximum quantity of data which the functionality of the software is licensed to use, in Terabytes. If the capacity type is **Engine** , this is the maximum quantity of engines which the functionality of the software is licensed to use.
- **Capacity Type**: Qualifies the capacity licensed. Possible values are:
  - **R-TB-Non-SATA**: Indicates that the capacity licensed applies to the raw capacity of all devices on the array, excluding SATA.
  - **R-TB-SATA**: Indicates that the capacity licensed applies to the raw capacity of all SATA devices on the array.
  - **REG-TB**: Indicates that the capacity licensed applies to the registered capacity of the VMAX array.
  - **Usable-TB**: Indicates that the capacity licensed applies to the usable capacity of the VMAX array.
  - **R-TB External**: Indicates that the capacity licensed applies to the raw capacity of the virtualized LUNs in external storage.
  - **Engine**: Indicates that the capacity licensed applies to the number of engines in the VMAX array.
- **Capacity Units**: The maximum quantity of data for which the functionality of the software is licensed to use, in Terabytes. If the capacity type is **Engine** , this is the maximum quantity of engines which the functionality of the software is licensed to use.
- **Days Until Expr**: Displays the number of days until expiration. For a Permanent license, this field displays a hyphen (-). This field only applies to Unisphere for VMAX.
- **Expiration Date**: Displays the expiration date. For a Permanent license, this field displays a hyphen (-).
- **Feature Name**: The name of the licensed feature.
- **Install Date**: The date the license was installed.
- **Lic**(ense Type): Whether the license is host-based (**SE**) or array-based (**EMCLM**).
- **SymmID**: The array to which the license is applied.

Querying licenses

The **`symlmf query`** command displays the current state and usage numbers for all licenses activated on a VMAX array.

For example, to display the state and usage number for all activated licenses on the VMAX3 array 1234, enter the following:

```
symlmf query -type emclm -sid 1234
```

Output similar to the following appears:

```
Symmetrix ID : 000000001234
Issue Date   : 03/22/2015
                                          Capacity
                                  -------------------------------
Feature Name                Act Type          Licensed Usage
--------------------------- --- ------------ -------- --------
Advanced_Suite              ENT Usable-TB          500    300.4
Foundation_Suite            ENT Usable-TB          500    300.4
Remote_Replication_Suite    ENT Usable-TB          500    300.4
DARE                        ENT Usable-TB        ARRAY    300.4


Legend:
  Act(ivation Type):
    ENT = Entitlement
    USE = In Use
```

If individual licenses had been purchased, output similar to the following appears:

```
Symmetrix ID : 000000001234
Issue Date   : 03/22/2015
                                          Capacity
                                  -------------------------------
Feature Name                Act Type          Licensed Usage
--------------------------- --- ------------ -------- --------
SYMM_VMAX_ENGINUITY         ENT R-TB-Non-SATA      100     19.2
                                R-TB-SATA          500    128.0
SYMM_VMAX_FAST_TIERING      ENT Reg-TB              60      0.0
SYMM_VMAX_OR_DM             ENT R-TB-Non-SATA      100     19.2
                                R-TB-SATA          500    128.0
                                R-TB-EXTERNAL      300      0.0
SYMM_VMAX_PROSPHERE         ENT R-TB-Non-SATA      100     19.2
                                R-TB-SATA          500    128.0
                                R-TB-EXTERNAL      300      0.0
SYMM_VMAX_SMC               ENT R-TB-Non-SATA      100     19.2
                                R-TB-SATA          500    128.0
                                R-TB-EXTERNAL      300      0.0
SYMM_VMAX_SRDF_REPLICATION  ENT Reg-TB              10      0.1
SYMM_VMAX_SRDF_STAR         ENT Reg-TB              20      0.0
SYMM_VMAX_TIMEFINDER        ENT Reg-TB              80      0.0


Legend:
  Act(ivation Type):
    ENT = Entitlement
    USE = In Use
```

Where:

- `Feature Name`: The name of the licensed feature.
- `Act`(ivation): How the product title was activated. Possible values are:
  - `ENT`: Indicates that the product title is activated through an entitlement.

  - `USE`: Indicates that the product title is activated because it was in use prior to upgrading to Enginuity 5876. In addition, this can also indicate that the product title was entitled in an earlier license file and not the current license file. Product titles in use (USE) are not considered properly entitled, in which case you should contact EMC for proper entitlement.
- `Capacity Type`: Qualifies the capacity licensed. Possible values:
  - `R-TB-Non-SATA`: Indicates that the capacity licensed applies to the raw capacity of all devices on the array, excluding SATA.
  - `R-TB-SATA`: Indicates that the capacity licensed applies to the raw capacity of all SATA devices on the array.
  - `REG-TB`: Indicates that the capacity licensed applies to the registered capacity of the VMAX array.
  - `Usable-TB`: Indicates that the capacity licensed applies to the usable capacity of the VMAX array.
  - `R-TB External`: Indicates that the capacity licensed applies to the raw capacity of the virtualized LUNs in external storage.
  - `Engine`: Indicates that the capacity licensed applies to the number of engines in the VMAX array.
- `Capacity Licensed`: The maximum quantity of data which the functionality of the software is licensed to use, in Terabytes. If the capacity type is `Engine` , this is the maximum quantity of engines which the functionality of the software is licensed to use
- `Capacity Usage`: The amount of Capacity Licensed currently being used.
  In addition, you can also add the `-output xml_element` option to the above command to produce an XML report containing the same information. For example:

```
symlmf query -type emclm -sid SymmID -output xml_element
```

Deleting licenses

Use the following command to delete a host-based license:

```
symlmf delete -type se -license LicenseName
```

Where *LicenseName* is one of the licenses in Table 19 on page 123 and Table 20 on page 123.

Note

You cannot delete array-based licenses.

Initial post-installation configuration of Solutions Enabler

This section describes the initial steps you must consider before you begin using Solutions Enabler SYMCLI commands.

Building the SYMAPI database

Before using the SYMCLI commands, you need to run the `symcfg discover` command to build your configuration (SYMAPI) database. This needs to be done once after installation, and after any changes are made to your VMAX array configuration.

Setting environment variables

After installing Solutions Enabler, you should set the environment variables or paths so you can directly access both the SYMCLI commands and the online help (man pages). The online help path allows you direct access to descriptions of the command set.

Note

For information on setting these variables, refer to Setting the CLI path on page 133 and Setting the online help path on page 134.

SYMCLI also provides additional environment variables that you can preset to streamline your command line session. These variables can be set to common argument values for a series of associated commands, which eliminates repeated key strokes for your session.

To view a list of environment variables that can be set for a given SYMCLI session, enter:

```
symcli -env
```

To view the environment variables that you currently have set, enter:

```
symcli -def
```

Note

For a complete list of the SYMCLI environment variables, refer to the *EMC Solutions Enabler SYMCLI Command Reference Guide*

Setting access permissions to directories

By default, the completed Solutions Enabler installation disables write access to other users beyond the owner. If you desire a different permission scheme, you can change it now. Refer to the *EMC VMAX Family Security Configuration Guide* for more information.

Starting the SCSI generic driver

Linux Kernel 2.4 requires that the SCSI generic driver be running. You can either compile it into the kernel or compile it as a loadable kernel module.

Note

For instructions, refer to the `README` file in the top level directory of your Linux source package.

Note

The SCSI generic driver is not required in Linux Kernel 2.6 or higher.

Verifying the existence of dedicated gatekeepers

To verify that there are dedicated gatekeepers available for use, run the following command:

```
stordaemon action storapid -cmd show -gk_stats
```

Note

For more information on this command, refer to Displaying gatekeeper statistics on page 254.

Setting the CLI path

Before using SYMCLI, append the SYMCLI binary directories to your PATH environment variable according to your operating system.

UNIX
For UNIX C shell, ensure the following SYMCLI directory is appended to variable PATH:

```
set path = ($path /usr/symcli/bin)
```

For UNIX Korn or Bourne shell, ensure the following SYMCLI directory is appended to variable PATH:

```
PATH=$PATH:/usr/symcli/bin
export PATH
```

Windows
For Windows, ensure the following SYMCLI directory is appended to the MS-DOS variable PATH:

```
C:\Program Files\EMC\SYMCLI\bin
```

OpenVMS
For OpenVMS, ensure the following SYMCLI directory has been defined for all users (use `emc_cli.com` in the system `login.com`):

```
SHOW LOGICAL SYMCLI$BIN
```

Setting the online help path

A complete set of online help (man pages) is provided for SYMCLI. To access these man pages in your environment, perform the following tasks according to your operating system.

UNIX
For UNIX C shell, ensure the following man page directories are added to variable `MANPATH`:

```
set MANPATH = ($MANPATH /usr/storapi/man /usr/storapi/storman)
```

For UNIX Korn and Bourne shell, ensure the following man page directories are added to variable `MANPATH`:

```
MANPATH=$MANPATH:/usr/storapi/man:/usr/storapi/storman
export MANPATH
```

Windows
For Windows, the manual pages are located, by default, in the following directories:

```
C:\Program Files\EMC\SYMCLI\man
C:\Program Files\EMC\SYMCLI\storman
```

To open a file, double-click it and select **NotePad** from the **Open With** dialog box.

Note

In Windows 2008 R2, double-clicking opens these files in WordPad by default.

OpenVMS
For OpenVMS, you can view help pages with the DCL utility SYMHELP.

Managing database and gatekeeper locking

Within a SYMCLI session, gatekeeper and database locks are used to avoid conflicts in accessing a VMAX array by way of gatekeepers or the configuration database.

Setting parallel SYMCLI access to the SYMAPI database

If an environment is configured to run many SYMAPI based applications on the same server, users might experience an issue when the SYMAPI_DB.bin is locked. This is caused by the `SYMCLI_CTL_ACCESS` environment variable that is set to `EXCLUSIVE` by default. This means only one command is allowed to execute at a time, and this command has an exclusive lock on the database.

To overcome the SYMAPI_DB.bin is locked issue, set the `SYMCLI_CTL_ACCESS` environment variable to `PARALLEL`. This mode enables multiple commands to have a read-only access to the SYMAPI database at the same time, while commands that need to modify the database would still have an exclusive lock on the database.

Semaphore requirements on UNIX

You do not need to modify semaphore settings on the host when using its default configuration (default options). However, some settings (for example, in the `daemon_options` file) will lead to semaphore allocation. In which case, you should configure the UNIX kernel to meet the SYMCLI semaphore requirements as follows:

- One semaphore ID for each VMAX gatekeeper device.
  The number of system-wide semaphores is specified by the UNIX kernel parameter `semmns`, or its equivalent.
- A minimum of three semaphores per semaphore set.
  The maximum number of semaphores per semaphore set is specified by the UNIX kernel parameter `semmsl`, or its equivalent.
- A minimum of three operations per `semop` call.
  The maximum number of operations per `semop` call is specified by the parameter `semopn`, or its equivalent.

See Setting the optional base daemon behavior parameters on page 146 for more information.

These requirements are usually within the bounds of the default semaphore parameter settings on a UNIX system. However, for information about maximizing these parameters on your specific platform, refer to Host specific behaviour running Solutions Enabler on page 257.

Meeting semaphore requirements

If the requirements are not within the bounds of the default semaphore parameter settings on a UNIX system, the UNIX kernel must be reconfigured. If the UNIX kernel is not reconfigured, the SYMCLI gatekeeper locking may fail. For more information about adjusting semaphore parameters for your operating system, refer to Host specific behaviour running Solutions Enabler on page 257.

Refreshing the semaphores

After you have reconfigured the UNIX kernel, you may need to reboot the UNIX system to refresh the kernel semaphore structures.

You can use the following UNIX command to view the currently allocated system semaphores:

```
ipcs -s
```

De-allocating semaphores

If you exceed the maximum number of semaphores allocated, you may need to de-allocate system semaphores in order to obtain more semaphores.

To de-allocate a system semaphore, use the following UNIX command:

```
ipcrm -s IpcID
```

Windows locking

On Windows, SYMCLI allocates named mutexes to accomplish locking. These mutexes are automatically de-allocated from the system when the last thread which has opened the mutex finishes accessing the mutex, or is terminated. There is no mutex kernel configuration requirement. The mutex name is derived from the gatekeeper pathname.

Avoidance and selection files

The following optional files can exist in the SYMAPI configuration directory[1], and limit the scope or change the performance of SYMCLI online commands, particularly, `symcfg discover` and `syminq`:

- gkavoid
- gkselect
- inqfile
- symavoid

Note

These files and the following text are for experienced SYMCLI or SYMAPI users and are not a prerequisite for normal use.

These files can be used to customize and streamline command line coding to your specific environment.

⚠ CAUTION

Be sure to delete these files when they are no longer needed as they can cause unexpected behavior and command limitations.

Editing and file format

These are editable files with device names or array IDs you can use to limit SYMCLI or SYMAPI from seeing certain VMAX arrays, devices, or gatekeepers which would otherwise be affected by various commands.

The files hold either physical device names (*PdevNames*) or array IDs (*Symmids*) with line entries having only one device name or ID per line. Lines beginning with a "*#*" (comment) are ignored by SYMCLI.

gkavoid and gkselect

The `gkavoid` and `gkselect` files affect calls to various online SYMCLI commands that use a gatekeeper to communicate with a VMAX array.

Note

For more information on using these files, refer to Using the gkavoid and gkselect files on page 251.

1. The location of this directory varies according to the operating system. For more information, refer to Solutions Enabler Directories on page 263.

inqfile

The `inqfile` file configures calls to `syminq` and `symcfg discover` to find only the *PdevNames* specified in this file. This can be useful if you want to limit the command(s) to view only certain devices from your host. The inquiry file is formatted with physical (host) device names with one *PdevName* per line.

Table 21 on page 137 provides platform specific *PdevName* examples.

Table 21 PdevName examples

Operating system	Example Pdevname
UNIX	`/dev/rdsk/c2t0d2s2`
Windows	`\\.\PHYSICALDRIVE1`
z/OS	`VOL001`

Note

For more information on *PdevNames*, refer to the *EMC Solutions Enabler Array Management CLI User Guide*.

symavoid

The `symavoid` file affects the operation of `symcfg discover` so that it does not look for devices that belong to the arrays specified in this file. This may be useful if there are multiple VMAX arrays connected to the host that you want SYMCLI to avoid. The array avoidance file is formatted with 12-character array IDs with one ID per line.

To obtain a list of array IDs, enter:

```
syminq -symmids
```

Changing the default behavior of SYMCLI

The `options` file (initially installed as `README.options`) in the SYMAPI configuration directory contains behavior parameters that can be set to critically change the default behavior of SYMCLI operations, SYMAPI calls, and their control actions. It can be used to impart certain global restrictions as well as customize and streamline command line coding to your specific environment.

⚠ CAUTION

This file and the text in this chapter are for experienced SYMCLI or SYMAPI users and are not a prerequisite for normal use. Improper adjustment of these parameters can impose unwanted restriction of features or possibly render your VMAX environment inoperative.

The `options` file must be created and placed in the SYMAPI configuration directory.[2]

Editing the options file

Once this file is created, you can edit it to change the default behavior of certain SYMCLI or SYMAPI command options. The file contains editable parameters to set certain optional defaults in the line entries. SYMAPI ignores lines beginning with a "#" (comment).

Removing default options

To remove a default option, remove the line entry, rename the file, or comment the line by adding a pound (#) sign at the beginning of the line entry.

Options file parameters

For `options` file parameter descriptions, refer to *EMC Solutions Enabler SYMCLI Command Reference Guide*.

Oracle multiple instances through a remote server

If you are using Storage Resource Management (SRM) and intend to perform database mapping calls from your host to a remote server that has more than one Oracle instance, you must complete the following procedure:

Procedure

1. With the remote SYMAPI service stopped, set the remote server UNIX environment variables *ORACLE_HOME* and *ORACLE_SID* for the system requirements. When set, re-start `storsrvd`.
2. Configure Oracle SQL*Net (V7) or Net8 to include other instance names (TNS names) in a network service. The TNS names are located in the `$ORACLE_HOME/network/admin/tnsnames.ora` file. The Oracle instance to which your *ORACLE_HOME* points is the only instance that must have the TNS names registered.
3. Configure the Oracle listener service for the other Oracle instances with which you need to work.
4. Test your Oracle environment for a valid configuration by running `$ORACLE_HOME/bin/sqlplus` as follows:

   ```
   sqlplus user/passwd@service
   ```

   where:

   user/passwd describes your Oracle username and password.

   service is the TNS name you registered for the Oracle instance.

   Note

   For more information about configuring SQL*Net or Net8, refer to the appropriate Oracle documentation.

2. The location of this directory varies according to the operating system. For more information, refer to Solutions Enabler Directories on page 263.

5. Set the EMC environment variable *SYMCLI_RDB_CONNECT* to describe your user name, password, and service name with the format `usr/passwd@service` to the instance of choice.

Client/server RDBMS environment variable behavior

The commands `symioctl` and `symrdb` scan the client's current environment variables and apply them across the client/server connection. For example, when the following is invoked from the client:

```
symrdb -type oracle list
```

`symrdb` will search for *ORACLE_HOME* and *ORACLE_SID* on the client side. If found, the variables are passed to the SYMAPI server and used with subsequent database mapping calls.

Set the*LD_LIBRARY_PATH* environment variable for all databases except Oracle and SQL Server.

Setting up daemons for distributed application support

To improve performance on a number of applications or scripts running at once, you can employ Solutions Enabler daemons (services) that run in the background with root privileges to a local storage resource. Applications do not have to run as a privileged user.

The base daemon (`storapid`) coordinates all VMAX array locks and parallel application syscalls to your operating system kernel, which optimizes their operations (such as TimeFinder-type actions).

For SRM applications, there are a number of vendor-specific database daemons available to improve the speed of database access or mapping operation. SRM database performance is improved by using a persistent database connection, a fast communication mechanism, and parallel operations. For SRM, a single database daemon can support connections to multiple instances/databases. In addition, there is also an SRM daemon (`storsrmd` and `storsrmd64`) that allows non-root users and non-administrators to perform certain SRM operations.

When your host is locally-connected to the VMAX array, applications and daemons must reside in that host. However, for client/server systems, the storage management applications reside in the client, and most of the daemons must reside in the SYMAPI server. The one exception to this is the event daemon, which runs on both the client and server.

Table 22 on page 140 lists the available daemons. Additional information is contained in the specific documentation for each. Note that on certain platforms, only some of these daemons are supported.

Table 22 Daemon support matrix

Daemon name	Platforms supported	Description	Daemon-specific parameter documentation
storapid	UNIX[a], Win64, z/OS, AS400	Base daemon	Refer to Managing the base daemon on page 145 in this guide.
storgnsd	UNIX, Win64, z/OS, AS400	Group Name Services (GNS) daemon	*EMC Solutions Enabler Array Management CLI User Guide*
storrdfd	UNIX, Win64	RDF daemon	*EMC Solutions Enabler SRDF Family CLI User Guide*
storevntd	UNIX, Win64, z/OS	Event daemon	Refer to Setting up the event daemon for monitoring on page 148 in this guide.
storsrvd	UNIX, Win64, z/OS, AS400	SYMAPI Server daemon (executes remote Solutions Enabler API functions)	Refer to Remote Operations on page 209 in this guide.
storwatchd	UNIX	UNIX only: Watchdog daemon	*EMC Solutions Enabler Array Management CLI User Guide*
storsrmd storsrmd64	Solaris, AIX, HP-UX, Windows	SRM daemon	*EMC Solutions Enabler Symmetrix Storage Resource Management CLI Product Guide*
storstpd	UNIX, Win64	Statistics (STP) daemon	
stororad		SRM daemon for Oracle DB	
storudbd		SRM daemon for UDB DB	
storsqld		SRM daemon for SQL DB	

Table 22 Daemon support matrix (continued)

Daemon name	Platforms supported	Description	Daemon-specific parameter documentation
storsybs12d		SRM daemon for Sybase DB - version 12	
storsybs12.5d		SRM daemon for Sybase DB - version 12.5	
storsybs12.5_64d		SRM daemon for Sybase DB - version 12.5 (64-bit)	
storvwmd	Linux	vWitness Manager Daemon that runs on the embedded VMAX Management Guests	vWitness Configuration Guide
storvwlsd	Linux	vWitness Lock Service Daemon that runs within a customer deployed management vApp	

a. UNIX represents Sun, AIX, HP-UX, and Linux systems.

For information on using daemons, refer to the remainder of this chapter.

Starting daemons

Most daemons are automatically started as their services are required. For example, `storgnsd` is automatically started the first time a group operation is performed.

However, in situations where you need to manually start a daemon, you can use the following command:

```
stordaemon start DaemonName [-wait Seconds]
```

By default, the `stordaemon` command waits 30 seconds to verify that the daemon is running. To override this, use the `-wait` option. For example, to start an SRM daemon for an Oracle database and wait five seconds for it to come up, enter:

```
stordaemon start stororad -wait 5
```

Stopping daemons

To stop a daemon, apply the following command:

```
stordaemon shutdown DaemonName |all [-wait Seconds] [-immediate] [-
abort]
```

By default, stopping a daemon causes it to no longer accept commands from client processes using its services; it does not actually exit until all client programs using its services exit first.

The `-immediate` option causes the daemon to exit regardless of whether there are still client programs connected to it.

The `-abort` option sends a KILL signal, instead of asking the specified daemon to shut itself down. Only privileged users (root) can use this option. (Supported on UNIX only.)

Viewing daemons

To view what daemons are present, enter either of the following:

```
stordaemon list [-running] [-all] [-v]
```

or

```
stordaemon show DaemonName
```

For the database daemons, an instance identifier is appended to the daemon name. For example, a `stororad` daemon started with the instance name `ords` would display as `stororadords`.

Setting daemons to auto-start on boot

To set a daemon to automatically start upon reboot of your system, enter the following:

```
stordaemon install DaemonName -autostart
```

Authorizing daemon connections

Typically, daemons run with root/administrator privileges, which enable them to handle the tasks required by SYMCLI commands (and any SYMAPI call) that require privileged access. This enables non-privileged users to run the SYMAPI application.

For example, when a SYMAPI call attempts to open a gatekeeper (which requires a privileged user), the request is actually passed to the base daemon process, which will open the gatekeeper device. If you were to run a process level debugger, such as `adb` on the Sun OS platform, and check the per-process file table, the open gatekeeper would appear in the base daemon process, not in the user process. From this point on, the transfer CDB requests are passed to the base daemon since it is the process that opened the gatekeeper.[3]

By default, the daemons only accept connection requests from users running with root or administrator privileges. For non-root users to use this feature, you need to create a `daemon_users` file (initially installed as `README.daemon_users`) with a list of allowed usernames.

The `daemon_users` file is an editable template file installed in the SYMAPI configuration directory.[4]

Using a text editor, a System Administrator can add entries to this file using the following formats:

`smith storapid`	Local user smith is authorized to use the `storapid` daemon.
`ENG/smith storapid`	Windows local user smith in the ENG domain is authorized to use the `storapid` daemon.
`smith storora*`	The `*` is a wildcard. Local user smith is authorized to use any daemon whose name begins with `storora`. For example, the SRM Oracle DB daemons.
`smith stororad freeze,...`	Local user smith is authorized to perform freeze and thaw operations via the `stororad` daemon. The third column consists of a comma separated list of operations that the user is authorized to perform. Valid values are: • `freeze`: The user is authorized to perform DB freeze and thaw operations. • `startup_instance`: The user is authorized to start a DB instance. • `shutdown_instance`: The user is authorized to shutdown a DB instance.

3. All daemons except for storapid the Base daemon may be configured to run as a non-root user in Unix. For details on considerations and configuration instructions, refer to the EMC VMAX Family Security Configuration Guide.
4. The location of this directory varies according to the operating system. For more information, refer to Appendix E.

Note

There is no reason to add privileged users to this file, as they are automatically authorized.

Note

For more information, refer to the `daemon_users` file.

Controlling daemon behavior

The `daemon_options` file (initially installed as `README.daemon_options`) contains parameters to control the behavior of the various Solutions Enabler daemons. As each daemon starts, it reads this file and applies all applicable settings.

⚠ CAUTION

These parameters are intended for experienced Solutions Enabler users. In most cases, the daemon default settings will be sufficient.

The `daemon_options` file is an editable template file located in the SYMAPI configuration directory.[5]

Using a text editor, a system administrator can add lines to this file using either of the following formats:

`NAME = VALUE`	Sets the parameter NAME for all daemons that understand this parameter.
`stororad:NAME = VALUE`	Sets the parameter NAME for only the `stororad` daemon.
`storora*:NAME = VALUE`	Sets the parameter NAME for all daemons whose name begins with `storora`. The `*` is a wildcard that can be used to match the remainder of a daemon's name.

Note

For more information, refer to the `daemon_options` file.

Controlling daemon logging

All Solutions Enabler daemons use a consistent infrastructure for logging events, which you can customize using the general logging options in the `daemon_options` file (Table 23 on page 145). In addition, the `daemon_options` file also includes daemon-specific options that allow you to further customize logging for a particular daemon (for example, `storevntd` and **`storsrvd`**).

5. The location of this directory varies according to the operating system. For more information, refer to Appendix E.

By default, each daemon records its log data in a pair of files (*daemon_name*`.log0` and *daemon_name*`.log1`) in the Solutions Enabler logging directory. Using this method, the daemons will alternate logging from one file to the other as they become full.

Optionally, you can configure each daemon to record its logs to a dated log file in the form *daemon_name*–*yyyymmdd*`.log`. Using this method, each daemon will begin recording to a newly dated log file on the first write after 12 A.M.

Table 23 on page 145 shows the general logging configuration options you can use to customize the Solutions Enabler daemon log files. For details on the syntax and values, refer to the `<SYMAPI_HOME>/config/daemon_options` file installed in the configuration directory.

Table 23 General logging configuration options in the daemon_options file

Option	Description
`logfile_type`	Controls file switching strategy. Possible values are WRAP or DATED.
`logfile_size`	Used for wrapping log files, this option specifies the maximum number of KBs to write before a switch to the other file of the pair.
`logfile_retention`	Used for dated log files, this option indicates how many days to retain old log files.
`logfile_perms`	Specifies the permissions on any newly created log files.

For logging configuration options specific to the event daemon, refer to Setting up the event daemon for monitoring on page 148, and for options specific to the SYMAPI server daemon, refer to Specifying server behavior on page 217.

Managing the base daemon

The base daemon (`storapid`) provides centralized gatekeeper device management for all Solutions Enabler applications requiring access to VMAX arrays, along with the GNS and RDF daemons. This alleviates contention when there are limited gatekeeper resources available and also eliminates the need for every client to constantly select, open, lock, and ping for an available gatekeeper device for every online function.

Additionally, the base daemon monitors Symmetrix External Locks (SEL) and Device External Locks (DEL), and automatically releases any SELs and DELs (except for persistent DELs) when an application (normally or abnormally) exits. The base daemon also eliminates the need for Solutions Enabler applications to run as root.

Note

For more on gatekeepers, refer to Gatekeeper Device Configuration on page 249.

Starting the base daemon

By default, the base daemon will automatically start the first time a Solutions Enabler application attempts to access a VMAX array. In addition, you can use either of the following methods to start the base daemon:

- Manually start the daemon via the `stordaemon` command line utility as follows:

```
stordaemon start storapid [-wait Seconds]
```

Note

For more information on this command, refer to Starting daemons on page 141.

- Set the base daemon to automatically start every time the local host is booted using the following command:

```
stordaemon install storapid -autostart
```

Note

`storapid` is installed with the `-autostart` option set by default.

Manually pre-starting the daemon will eliminate any performance delay incurred when the base daemon needs to be started by an application the first time it tries to connect.

If the base daemon abnormally terminates, the Solutions Enabler watchdog daemon (`storwatchd`) will automatically restart it. This ensures that the base daemon is always running.

Stopping the base daemon

To stop the base daemon, use the following command:

```
stordaemon shutdown storapid | all [-wait Seconds] [-immediate] [-
abort]
```

Specifying `all` as the *DaemonName* will stop all of the daemons currently running.

If there are applications with connections to the base daemon, you can use the `-immediate` option to shut it down immediately; otherwise, it will not shutdown until the applications are done using it.

The `-abort` option sends a KILL signal, instead of asking the base daemon to shut itself down. Only privileged users (root) can use this option. (Supported on UNIX only.)

Setting the optional base daemon behavior parameters

The `daemon_options` file contains a set of parameters that can be modified to affect base daemon behavior. The file contains editable behavior parameters set to

certain optional defaults in the line entries. Commented lines beginning with a pound sign (#) are ignored.

To remove any parameter option, remove the line entry, rename the file, or comment the line by adding a pound sign (#) at the beginning of the line entry.

Table 24 on page 147 lists some of the possible optional base daemon parameters.

Table 24 Base daemon optional behavior parameters[a]

Parameter	= <OptValue | defaultvalue>	Description
storapid:inquiry_timeout	`0` - `nn`, `-1` | `900`	Specifies how long (in seconds) inquiry results are to remain in cache before expiring, and new data retrieved from the host and array. A value of `-1` indicates the data never expires. A value of zero indicates the data always expires.
storapid:gk_use	`dedicated_only` | `legacy`	Specifies whether the base daemon is restricted to only using dedicated gatekeeper devices when making syscalls. `dedicated_only` restricts the base daemon to only dedicated gatekeepers. `legacy` allows the base daemon to use non-dedicated gatekeeper devices.
storapid:use_all_gks	`disabled` | `enabled`	Specifies whether the base daemon is free to use all available gatekeeper candidates. `disabled` restricts the base daemon to using only 75% of the available gatekeeper candidates. This option locks the gatekeeper with a host-based lock, such

Table 24 Base daemon optional behavior parameters[a] (continued)

Parameter	= <OptValue | defaultvalue>	Description
		as a semaphore or mutex. `enabled` allows the base daemon to use all available gatekeeper candidates. This option locks the gatekeeper with an internal locking mechanism. If you are running InfoMover, you must set this option to `disabled`.

[a]For more information on the available parameters, refer to the `daemon_options` file.

Setting up the event daemon for monitoring

The Solutions Enabler event daemon (`storevntd`) acts as a clearinghouse for events, also known as alerts, on a host. It supports two modes of operation. This section concentrates on the second mode of operation.

- Under the first mode, applications register for events (an event is defined by one or more conditions) in which they are interested through Solutions Enabler API calls. These requests are forwarded to the event daemon which then begins to watch for the conditions of interest. When an event is detected, it triggers an asynchronous callback to the application.
  Clients such as Unisphere for VMAX and SMI Provider all make use of this mechanism.
- Under the second mode, the event daemon actively watches for conditions of interest — independently of any applications. Options settings (described in Configuring event logging on page 152) specify the events for which the daemon should monitor and how it should log them when they occur. Possible logging options are:
  - `file`: record to a file on disk
  - `system`: record through the logging service provided by the host operating system. On UNIX-like systems, this is the local syslog service. On Windows, this is the Windows event log.
  - `syslog`: use the syslog wire protocol to forward event records to a remote syslog server, that is, an RSA enVision server.
  - `snmp`: forward event records to a remote SNMP listener. Solutions Enabler only supports SNMP version 1 traps.

Note

Only events for VMAX arrays are supported in this mode.

Event sources

The events daemon monitors for events from the following sources:

- Events that are directly generated by a storage array, and are merely routed by the event daemon to interested parties.
- Events manufactured by the event daemon by periodically polling the storage array and tracking various conditions. For example, an event tied to the overall utilization (as a percentage) of a Snap pool.
- Events that are generated by a different process entirely, and are forwarded to the event daemon to be routed to any interested parties. For example, the GNS (`storgnsd`) and Base (`storapid`) daemons both generate events that applications can register to receive
- The event daemon can also be directed to map records from the Audit log into events.
- Non-array events raised by applications such as Unisphere for VMAX.

Events, when delivered, contain a number of pieces of information including, but not limited to, the following:

- The entity to which the event relates. This will usually be an array ID.
- The sub-component to which the event relates, when there is one. The following is a list of the most relevant sub-components.
  - A device number as a 4-digit hexadecimal number, for example, 0007 or 0123.
  - A disk ID using the standard Solutions Enabler syntax, for example, 16B:C2.
  - A director ID using the standard Solutions Enabler syntax, for example, FA-3B.
  - A port on a director, for example, SA-03C:2.
  - A Snap, DSE, or thin pool using the pool name, for example, finance or cambridge.
- The identifier of the event corresponding to the SYMAPI_AEVENT2_UID_T enumeration found in the `symapi.h` header file that is shipped with the SDK.
- A severity level. Possible values are: NORMAL, INFO, WARNING, MINOR, MAJOR, FATAL, and CRITICAL. The NORMAL severity is relevant to threshold events described in the next section.
- The date/time that the event was generated.
- For certain events, a numerical value, which is used to determine the severity of the events. This concept is described in the following section.
- A description of the event along with some auxiliary textual data.

Threshold events

Certain events are associated with a numeric value. This value is compared with a set of threshold values, which determine whether the event is delivered and, if so, with what severity. These events are known as threshold events. Each threshold event has a set of default threshold filters defined for it.

For example, the SYMAPI_AEVENT2_UID_THRESH_POOL_FREESPACE event tracks as a percentage (0% - 100%) the space utilization within DSE, Snap and thin pools and has the following default threshold filters defined:

- If value is 100%, deliver event with FATAL severity
- If value is >= 80%, deliver event with CRITICAL severity
- If value is >= 70%, deliver event with MAJOR severity
- If value is >= 65%, deliver event with MINOR severity
- If value is >= 60%, deliver event with WARNING severity

When registering for events, you can specify a custom filter to replace the default one for that event. Each filter contains a set of rules composed of:

- A comparison function: either >= or <=.
- A number (integer) to compare the event value against.
- A severity to deliver the event with - if the comparison succeeds.

These threshold filters define bands of event value. Events are generated as the value crosses from one band to another. For the thresholds in the earlier example, a pool's utilization that rose gradually from 60% to 92% and then dropped back to 50% again would result in delivery of the following events:

WARNING — severity when the value passes 60%

MINOR — severity when the value passes 65%

MAJOR — severity when the value passes 70%

CRITICAL — severity when the value passes 80%

MAJOR — severity when the value drops below 80%

MINOR — severity when the value drops below 70%

WARNING — severity when the value drops below 65%

NORMAL — severity when the value drops below 60%

If an event's value crosses into a range that does not match any of the configured thresholds, the event daemon will automatically deliver an event with a severity of NORMAL to indicate that it no longer falls into one of the defined threshold bands. In essence, NORMAL should serve as an "all-OK" indicator.

There is never a reason to explicitly specify a threshold for the NORMAL severity. It should cover everything that is not explicitly matched.

Note

Many of the threshold events that indicate a percentage will only trigger at increments of 5%.

If the supplied threshold list has only a single filter that performs a comparison against zero, the event daemon will deliver an event every time the event value changes. For example, specifying the following filter:

```
"If value >= 0 : WARNING"
```

will deliver an event with WARNING severity every time the value changes.

Starting the event daemon

By default, the event daemon will automatically start the first time a Solutions Enabler application requires its services. However, you can also manually start the event daemon via the `stordaemon` command line utility as follows:

```
stordaemon start storevntd [-wait Seconds]
```

Note

For more information on this command, refer to Starting daemons on page 141.

In addition, you can also set the daemon to automatically start every time the local host is booted using the following command:

```
stordaemon install storevntd -autostart
```

Note

Configure the daemon to automatically start at system boot when you will be using it to log events to a Syslog, Event log, SNMP, or file on disk.

Reloading the daemon_options settings

To reload the event daemon settings, run the following command:

```
stordaemon action storevntd -cmd reload
```

Issuing the `reload` command causes the daemon to re-read the contents of the `daemon_options` file.

Listing supported event categories

To view a list of event categories currently supported by a running event daemon:

Procedure

1. Run the following command to load the array event module:

   ```
   stordaemon action storevntd -cmd load_plugin Symmetrix
   ```

2. Run the following command to list the supported event categories:

   ```
   stordaemon action storevntd -cmd list -categories
   ```

Stopping the event daemon

To stop the event daemon, run the following command:

```
stordaemon shutdown storevntd [-wait Seconds]
```

Note

For more information on using the `shutdown` command, refer to Stopping daemons on page 142.

Configuring event logging

The `daemon_options` file contains a set of parameters that can be modified to affect event daemon behavior. The file contains editable behavior parameters set to certain optional defaults in the line entries. Commented lines beginning with a pound sign (#) are ignored.

To remove any parameter option, remove the line entry, rename the file, or comment the line by adding a pound sign (#) at the beginning of the line entry.

Configuring event logging involves the following steps:

1. Specify logging targets.
2. Configure an event target.
3. Specify events to log.

The remainder of this section explains `daemon_options` file settings required to complete each of these steps.

Note

Changes made to the `daemon_options` file while the daemon is running will not take effect until you issue a `stordaemon reload` command, as described in Reloading the daemon_options settings on page 151.

Step 1: Specify logging targets

To specify a logging mechanism, define the following parameter in the `daemon_options` file:

```
storevntd:log_event_targets = snmp syslog system file
```

Note

You must set this parameter to one or more of the valid values; otherwise, event logging will not occur. When specifying multiple values, separate them with a space.

where:

`snmp` specifies to log events by way of SNMP traps. Solutions Enabler only supports SNMP version 1 traps.

`syslog` (supported on all platforms) specifies to log events to a Syslog server across the network, bypassing (if on UNIX) the local host's Syslog service and its configuration settings.

`system` does the following depending on the operating system:

- In UNIX, it specifies to log events to local host's Syslog services. The Syslog's configuration settings control where it directs the message.
- In Windows, it specifies to log events to the Windows Event Log.
  `file` specifies to log events to a file on disk.

  For example:

```
storevntd:log_event_targets = snmp system
```

Step 2: Configure an event target

To configure an event target, do the following based on the logging mechanism you specified in Step 1: Specify logging targets on page 152 above:

- If you specified to log events by way of SNMP (`snmp` option), complete Step 2A: Configure an SNMP event target on page 153.
- If you specified to log events in a log file (`file` option), continue with Step 2B: Configure a log file on page 155.
- If you specified to log events to the Syslog server across the network (`syslog` option), continue with Step 2C: Configure a Syslog target on page 158.
- If you specified to log events to Syslog or the Windows Event Log, (`system` option), you do not have to configure an event target. In this case, you should continue with Step 3: Specifying events to log on page 158.

Step 2A: Configure an SNMP event target

The event daemon provides the necessary SNMP MIB support and trap generation services required to monitor the status of VMAX storage environments from third-party enterprise management frameworks.

The event daemon includes a loadable SNMP library which, once enabled and configured in the `daemon_options` file, acts as a self contained SNMP agent. It is responsible for maintaining internal Fibre Alliance MIB (V3.0) tables, responding to SNMP browse requests, and generating traps in response to events.

For an application to receive SNMP trap information from the event daemon, you must specify it as a trap target by defining the following parameter in the `daemon_options` file:

```
storevntd:snmp_trap_client_registration = IP,Port,Filter,State
```

where:

IP is the application's IP address.

Port is the port on which the application will be listening for the trap. The default port is 162.

Filter is the trap filtering severity level as defined in the FC-management MIB. The application will only receive traps of the specified severity level (or lesser). The default value is 10 (Mark), which means that all events are delivered.

Table 25 on page 154 maps the event daemon severity level to the SNMP severity levels, as specified in the FC-management MIB.

Table 25 Event daemon severity level/SNMP severity level mappings

Event daemon severity	SNMP trap severity
	1 (Unknown)
fatal	2 (Emergency)
	3 (Alert)
critical	4 (Critical)
major	5 (Error)
minor	5 (Error)
warning	6 (Warning)
info	8 (Info)
normal	8 (Info)
	9 (Debug)
--	10 (Mark)

State is the start up row state in the `trap_client_registration` table in the FC-management MIB. Possible values are ACTIVE and INACTIVE.

Multiple entries can be on the same line, separated by a blank space. In addition, they can be on their own line, delineated with a backslash (\) character on the preceding line.

For example, the following registration file specifies that the daemon will only send SNMP traps to the indicated clients when it detects an event of a severity level less than or equal to 5 (that is, Error, Critical, Emergency). The daemon will ignore events with a severity level greater than 5:

```
storevntd:snmp_trap_client_registration = 10.2.12.30,162,5,ACTIVE \
                                          12.250.130.200,162,5,ACTIVE
```

Object IDs

Object Identifiers (OIDs) are entries in the Management Information Base (MIB).

Table 26 Event daemon severity level/SNMP severity level mappings

Object name	Object identifier	Description
connUnitEventId	1.3.6.1.3.94.1.11.1.3	This is the event index, mainly for internal use.
connUnitEventSeverity	1.3.6.1.3.94.1.11.1.6	The SNMP trap severity. The values are listed in Table 25 on page 154.
connUnitEventType	1.3.6.1.3.94.1.11.1.7	This is the event type. Possible values are:

Table 26 Event daemon severity level/SNMP severity level mappings (continued)

Object name	Object identifier	Description
		• UNKNOWN: 1 • OTHER: 2 • STATUS: 3 • CONFIGURATION: 4 • TOPOLOGY: 5
connUnitEventObject	1.3.6.1.3.94.1.11.1.8	This field is always NULL.
connUnitEventDescr	1.3.6.1.3.94.1.11.1.9	This is the description of the event. See sections #unique_247 and #unique_248 for detail.
connUnitName	1.3.6.1.3.94.1.6.1.20	This is the array ID.
connUnitType	1.3.6.1.3.94.1.6.1.3	This is the array type. Possible values are: • OTHER: 2 • STORAGE_SUBSYSTEM : 11
emcAsyncEventSource	1.3.6.1.4.1.1139.3.8888.1.0	The source of the events, for example Symmetrix or CLARiiON.
emcAsyncEventCode	1.3.6.1.4.1.1139.3.8888.2.0	This is the event ID. See sections #unique_247 and #unique_248 for detail.
emcAsyncEventComponentType	1.3.6.1.4.1.1139.3.8888.3.0	This is the component type, for example FastSRP.
emcAsyncEventComponentName	1.3.6.1.4.1.1139.3.8888.4.0	The component name, for example SRP on which the event was generated.

Step 2B: Configure a log file

The `daemon_options` file contains parameters (Table 27 on page 156) that allow you to configure the log file.

The target log file is not actually opened (or created, if necessary) until the event daemon actually has an event to log. Depending on the events it is monitoring, this may not be until long after it starts.

Table 27 Event log file configuration options

Parameter	= <OptValue | defaultvalue>	Description
storevntd:log_event_file_name	*LogEventFileName* | events	Specifies the base name of the event log files, which can also include the full pathname. This file is created in the standard Solutions Enabler log directory. For UNIX, the directory is: `/var/symapi/log` For Windows, the directory is: `c:\Program Files\EMC\SYMAPI\log`
storevntd:log_event_file_type	dated | wrap	Specifies the type of file to use. `dated` specifies that a new event log file should be created each day, with the name *xxxx*`-`*YYYYMMDD*`.log`. Where *xxxx* is the *LogEventFileName*. `wrap` specifies that event logging will alternate between two files (*xxxx*`.log0` and *xxxx*`.log1`) - switching from one to the other when it reaches its maximum size, as specified in the `log_event_file_size` parameter. By default, a single file will be used.
storevntd:log_event_file_size	> 0 - *nn* | 1	When used with the `log_event_file_type` parameter set to wrap, this parameter specifies the

Table 27 Event log file configuration options (continued)

Parameter	= <OptValue | defaultvalue>	Description
		maximum file size (in KB) allowed before wrapping to the alternate file. This value should be a decimal number greater than zero. Note: The maximum value for the `log_event_file_size` is 2097152 KB.
storevntd:log_event_file_retention	> 0 - *nn* | 3	When used with the `log_event_file_type` parameter set to dated, this parameter specifies the number of days to retain the log files. This value should be a decimal number greater than zero.
storevntd:log_event_file_perms	rw, n | r	Specifies the permissions for the event log files. `rw` specifies that anyone can read or write to the files. `r` specifies that anyone can read the files, but only the root/administrator (or whatever identity the event daemon is running as) can write to the files. `n` specifies that only the root/administrator (or whatever identity the event daemon is running as) can read and write to the files.

Step 2C: Configure a Syslog target

The `daemon_options` file contains parameters (Table 28 on page 158) that allow you to configure a Syslog target.

Table 28 Event log file configuration options

Parameter	= <OptValue | defaultvalue>	Description
storevntd:log_event_syslog_host	*SyslogHostName*	Specifies the name of the host on which the Syslog server is running. This value must be supplied.
storevntd:log_event_syslog_port	*nnn* | 514	Specifies the port on which the server is listening.

Step 3: Specifying events to log

Solutions Enabler provides the ability to capture both array events and non-array events from certain application to log files. This is accomplished by building event lists, which is a mechanism for specifying the types of events for which to generate traps. These event lists are defined in the `daemon_options` file.

Array events

To build an array event list, define the following parameter in the `daemon_options` file:

Note

Many array events are organized into categories. These categories are hierarchical in that a category can contain individual events, as well as other categories.

```
storevntd:log_symmetrix_events = [sid=SymmID,] UID|Category ...
[,sev=SEV] [,tgt=TGT] [,comp=COMP] [,comp_type=CPMP_TYPE]
[thresh_critical=Percent, thresh_maj=Percent, thresh_warn=Percent,
thresh_info=Percent, thresh=Percent] [,ignore]
```

where:

sid— Specifies the 12-digit ID of the VMAX array to which the record applies. You must specify the full SID (12 digits). If this field is missing, the registration applies to all local and remote VMAX arrays.

UID — The numerical event UID value.

Category — One or more of the following event categories, separated with a comma:

For events in the 1150 - 1199 range:

- events (all events in this category)
- array subsystem

- checksum
- diagnostic
- environmental
- device pool
- service processor
- srdf system
- srdf link
- srdf session
- srdf consistency group
- director
- device
- disk

For events in the 1200 - 1999 range:

- status (general component state change)
- optimizer (Optimizer/FAST related)
- groups (Group (DG/CG) related)

Note

Each of the event categories may contain numerous individual events, as described in the *VMAX Management Software Events and Alerts Guide*.

---

sev— Specifies the minimum severity level for which events should be logged. All events with a severity level at or above the specified severity will be logged. Take care when setting this option. Possible values are:

- normal
- info
- warning
- minor
- major
- critical
- fatal

tgt — Specifies the target to which the daemon should log the events. Possible values are: snmp, syslog, system, and file.

The value you specify for *TGT* must match one of the values you specified in the `log_event_targets` parameter; otherwise, the daemon will not log events for this record.

The target you specify here will override the global `log_event_targets` setting described in Step 1: Specify logging targets on page 152.

comp — Specifies the specific sub-component for which you want to log events. For example, a particular device, disk, pool, etc. When you specify a value for this field, the event daemon will only log events for the specified component. You can either specify a single component or a comma separated list of components. If the latter, you must enclose the list with double quotes.

For example:

`comp=0100`	a single device
`"comp=0100,0200,030"`	multiple devices
`"comp=finance,sales"`	multiple pools

cmpnt_type — Specifies a type of component. When present, only events for the specified component type are delivered. If omitted, events for any component type are delivered. This is most useful for events that can be delivered against multiple types of components.

An example is the Pool Status events, which can be generated for DSE, Thin or Snap Pools. Possible values are: `device`, `disk`, `director`, `port`, `dsepool`, `tpdatapool`, `snappool`, `dg`, `cg`, `sg`, `srdf-grp` and `migrsess`.

`thresh_critical=`*`Percent`* `thresh_maj=`*`Percent`* `thresh_warn=`*`Percent`* `thresh_inf=`*`Percent`* `thresh=`*`Percent`*	Specifies the threshold level at which the daemon delivers an event and at what severity it is delivered. This setting overrides the default threshold levels for an event. These parameters are only used when specifying threshold type events. Only a subset of the full threshold functionality described in Threshold events on page 149 is supported. The MINOR and FATAL severities cannot be specified and a >= comparison is assumed. The `thresh=`*`nnn`* setting is an alias for `thresh_maj`.

ignore — Indicates that events matched by this record are not to be delivered, even if they are matched by some other record. The order of records doesn't matter. If an event is matched by any record with the ignore parameter, it will be ignored.

Only a single `log_symmetrix_events` option can be present. Since this can become quite long, it can be spread across multiple lines in the file via the use of '\' continuation characters at the end of a line.

Note

The comment character (#) has no effect if it follows a line with the continuation character (/).

Non-array events

To build a non-array event list, define the following parameter in the **daemon_options** file:

```
storevntd:log_app_events = [appid=appid,] CAT[category,]
[comp=COMP,] [comp_type=COMP_TYPE,] [,tgt=TGT]
```

where:

appid— Specifies an application id. By default, all application events will be monitored.

CAT — Specifies event(s) to be monitored. This can be either the name of an event category or a numerical event ID. This is the only field that is required. One or more values (comma separated) may be present. The Supported categories are: SMC and SPA.

comp — Certain events apply to specific sub-components within the application. This field specifies that only events for the specified component (or components) should be delivered. If more than one component is present, the entire field must be enclosed in double quotes.

For example:

comp=name	a single component
"comp=name1,name2,name3"	multiple components

comp_type — Specifies events to be monitored. This must be one or more of the predefined types. The supported component types are: `univmax`, `univspa`, `univspv`, `jboss`, and `dbms`.

tgt — Specifies the target to which the daemon should log the events. Possible values are: `snmp`, `syslog`, `system`, and `file`.

The value you specify for *TGT* must match one of the values you specified in the **log_event_targets** parameter; otherwise, the daemon will not log events for this record.

The target you specify here will override the global `log_event_targets` setting described in Step 1: Specify logging targets on page 152.

An example with 4 records or separate registrations is as follows:

```
storevntd:log_event_targets = syslog file
storevntd:log_symmetrix_events = \
  sid=000192600356, 1200,1201,1202 ;\
  sid=000192600357, "comp=0001,0002,0003",1204,1205 ;\
  1212,1213, thresh_major=60, thresh_warning=50, thresh_info=30 ;\
  tgt=file, sid=000194900123, status
```

Event output examples

The following examples illustrate the format of the various event outputs. For a more detailed description of the event formats, refer to Event message formats on page 162.

In these examples:

- `symid:000194900123` is the event entity; normally a storage array.
- `date=xxx` corresponds to the date/time that the event was originally generated. If the date field contains a `Z` suffix, the date is in UTC time, otherwise, it is local time. If the example contains a second date field, it indicates when the logging service (for example, Syslog) posted the event.

Log file
The following example illustrates the format of an event as reported in a log file (target = file):

```
[evtid=1200] [date=2010-12-22T09:08:17] [symid=000194900123]
[Device=0010] [sev=normal]  = Device state has changed to Offline.
```

Syslog service (local UNIX host)
The following example illustrates the format of an event as reported by Syslog service on a local UNIX host (target = system).

Note that the italicized text was generated by local Syslog service. In this case, a Solaris host:

```
Dec 22 09:08:17 l82ab139 storevntd[14505]:
 [ID 989319 user.info][evtid=1200] [date=2010-12-22T09:08:17]
[symid=000194900123] [Device=0010] [sev=normal]  = Device state has
changed to Offline.
```

Syslog service (different system)
The following example illustrates the format of an event as reported to a Syslog service on a different host (target = syslog):

```
Dec 22 09:03:01 EMCstorevntd:
 [evtid=1200] [date=2010-12-22T04:08:17Z] [symid=000194900123]
[Device=0010] [sev=normal]  = Device state has changed to Offline.
```

Windows event log
The following example illustrates the format of an event as reported in a Windows event log (target = system):

```
[evtid=1200] [date=2010-12-22T09:08:17] [symid=000194900123]
[Device=0010] [sev=normal]  = Device state has changed to Offline.
```

SNMP trap
SNMP traps are formatted according to the Fibre Alliance MIB (V3.0). Messages contained in a trap are the same as used with the system and file logging.

Event message formats

As discussed in earlier, the Event Daemon can be configured to automatically log events to a number of different targets (also known as destinations):

- A disk file
- Syslog
- SNMP
- Windows Event Log or local syslog service on UNIX

These log messages consist of a destination specific portion (discussed later) and a common portion. The common portion has the following format:

```
{SDEs } = {Message
```

{*SDEs*} — A series of Structured Data Elements, each holding a '[Name=Value]' pair of tagged data.

{*Message*} — The text associated with the event.

The {*SDEs*} and {*Message*} are separated by space, equals, space (i.e.: ' = ').

In samples found below, line breaks have been added to improve readability.

For events derived from Audit log records, the event {*Message*} may itself contain multiple new lines spanning multiple lines. There will be no new lines in the {*SDEs*}.

The number of SDEs will in general be variable. Different SDEs may be present depending on the type of event - and optional ones may be omitted.

Likewise, the position (first, second, third, ...) of specific SDEs within a message cannot be relied on - except as noted below. The following common SDEs are used within all event messages:

[fmt=xxx]	The fmt SDE specifies the format of the message - its overall type. This will always be the first SDE in the message. Currently supported formats are: symaudit: Events that correspond directly to records from the Audit log. These are discussed in more detail further below. evt: All other events generated by the Event Daemon. Example: [fmt=evt]
[date=...]	The Date/Time. The format of the date adheres to the Syslog Protocol: *yyyy-mm-dd*T*hh:mm:ss*[Z] This contains a Date (*yyyy=mm=dd*) and Time (*hh:mm:ss*), separated by a 'T'. A trailing 'Z' signifies a UTC time ... otherwise, the time is Local. Events targeted to a Syslog server (target = syslog) will include a UTC ('Z') time.

	Other targets will include a Local time. Example: [date=2007-10-30T08:06:40]
[symid=....]	The ID of the array that the event relates to. This SDE is optional. Example: [symid=000192600386]

Note

Depending on the type of event, additional SDEs will be present as discussed in subsequent sections.

Format for simple events

In broad terms, there are two categories of events. Events derived from Audit log records are discussed in the next section. Other events generated by the event daemon are formatted with the following SDEs:

[fmt=evt]	Format. Always be the 1st SDE.
[evtid=1234]	Event UID. Always the 2nd SDE. This gives the type of event.
[date=2007-10-30T08:06:40]	Event time stamp. Always the 3rd SDE. See above.
[symid=000192600386]	Array ID. Optional. Identifies the VMAX array that the event relates to.
[{Comp}=name]	Component ID. Optional. Identifies, where it is known and meaningful, the sub-component within the array that the event relates to. The following are some of the component types that may be present: [Device=0030] Device [Disk=16B:C2] Disk [Director=FA-3B] Director [Port=SA-03C:2] Port on a Director [SRDF-grp=7] SRDF Group

	[SnapPool=sales] Snap Save Device Pool [DSEPool=mkt] DSE Device Pool [TPDataPool=eng] Virtual Provisioning Device Pool [SEL=nn] Symmetrix External Lock The following component types correspond to sub-modules (or enclosures) within a VMAX array. At this time, they occur with the array sub-component Environmental alert SYMAPI_AEVENT2_UID_ALERT_ARR_COMP_STATUS. The format of the component name can vary depending on the array model. As an example, one might encounter: "SB-1/Fan-A" or "SB-1/MIBE-L-2A/PS-A" or "DB-1/PS-A" [Power=xxxxx] Power sub-system [Fan=xxxxxx] Fan sub-system [LCC=xxxxx] Link Control Card [Enclosure=xxxxx] Enclosure [MM=xxxxx] Management Module [IOMC=xxxxx] IO Module [Dir=xxxxx] Director (for environmental alerts)
[sev=warning]	Event Severity. Optional. Supported values are: normal, info, warning, minor, major, critical, fatal

In the future, additional SDEs may be added (for example: Process ID).

Example:

```
[fmt=evt] [evtid=1201] [date=2006-12-17T10:33:05] [symid=000000006190]
   [sev=fatal] = Array state has changed to Unknown.

[fmt=evt] [evtid=1200] [date=2006-12-17T21:54:53] [symid=000000006190]
   [Device=0007] [sev=major] = Device state has changed to Offline.
```

Format for audit log records

Events derived from Audit log records are formatted differently—with an expanded set of SDEs.

Format	Description
`[fmt=symaud]`	Format. Always be the 1st SDE. See above.
`[date=2007-10-30T08:06:40]`	Event time stamp. Always the 2nd SDE. See above. This is the time that the Audit record was originally written.
`[symid=000000001234]`	Array ID. Always the 3rd SDE.
`[orig=SE]`	An indication of the originator of this audit message. Possible values are: SE Solutions Enabler (host based application) SW SymmWin (SP based) UC Array software (ucode) ' ' Empty string: Unknown
`[user=H:jupiter\jones]`	The user name field from an Audit record - if there is one.
`[host=saturn]`	The host_node name field from an Audit record - if there is one.
`[actid=SE12345678ab]`	The activity_id field from an Audit record - if there is one.
`[appid=InternalTest]`	The application_id field from an Audit record - if there is one.
`[aud-cls=Security]`	The audit_class field from an Audit record. This field will always be present and have a value of 'NA' if nothing better can be provided.
`[aud-act=Add]`	The action_code field from an Audit record. This value will

Format	Description
	always be present and have a value of '' (empty string) if nothing better can be provided. **Note** Parsing logic should treat this field as being optional.
`[aud-num=1234]`	The record_num field from an Audit record. Several formats are possible: 1234 Entire message fits in one audit record 1234,1/4 1st of 4 records in the message 1235,2/4 2nd of 4 records in the message 1236,3/4 3rd of 4 records in the message 1237,4/4 4th of 4 records in the message **Note** For a segmented (multiple audit record) message, each record is delivered with a different record number. These could end up interleaving with other audit messages - and appear with non-sequential record numbers.

Example:

```
[fmt=symaud] [date=2006-12-18T12:33:03] [symid=000000006190] [orig=SE]
    [user=jupiter\jones] [host=saturn] [actid=SEba8cde5711]     [appid=Internal_Test]
    [aud-cls=Security] [aud-act=Add] [aud-num=74]
    = The User Authorization set role operation SUCCEEDED
```

Notes

- This overall format is compatible with BSD Syslog (RFC 3164).
  Some extensions were motivated by the Syslog NG proposal: a simplified version of Structured Data, and the Date/Time format.

- The first step in parsing the text of an event is to search for the first ' = ' (<space>=<space>) in the string. Before this will be the SDEs added by the event daemon. After this will be whatever message (possibly multi-line) is associated with the event.
- We assume that SDE values cannot contain ']' characters - so these are not being escaped. To be safe, parsing logic should assume that SDEs end in a '] ' (right bracket, space). The last SDE will be followed by a ' = ' (space, equals, space) - with perhaps an extra space character.
- Parsers should tolerate additional white space between SDEs. Although there will be at least one space between SDEs, there may be more. Similarly, there may be additional white space before the ' = ' that terminates the SDEs.
- The order of SDEs shown above, some of which are optional, will be constant. In particular, the Component SDE (difficult because of the large and growing number of component types) will, if present, directly follow the symid one.
  If new SDEs are added in the future (for example: a process PID : [pid=nnn]) they will be added to the end of the list - before the " = " marker that begins the event message.

  To be safe, however, parsers should if possible not rely on the order of the SDEs.
- Parsers should treat SDEs that are marked optional above as such. They may or may not be present.
- The Component ID SDE is, in particular, optional. A given event may sometimes be delivered with a this SDE and sometimes not - depending on whether a component name is known.
  Similarly, a given event may be delivered with different component types. For example, the SYMAPI_AEVENT2_UID_ALERT_ARR_COMP_STATUS alert [event id 1244] may be raised against a component of FAN, MM, IO, POWER, etc.

Format for msgs written to Target = File

Event messages directed at a file on disk are written exactly as previously discussed.

Example 1 Examples:

```
[fmt=evt] [evtid=1200] [date=2006-12-17T21:54:53] [symid=000000006190]
     [Device=0007] [sev=major] = Device state has changed to Offline.

[fmt=symaud] [date=2006-12-18T12:33:03] [symid=000000006190] [orig=SE]
    [user=H:jupiter\jones] [host=saturn] [actid=SEba8cde5711] [appid=Internal_Test]
    [aud-cls=Security] [aud-act=Add] [aud-num=74]
    = The User Authorization set role operation SUCCEEDED
```

As noted above, the 'Message' portion of events derived from Audit Log records may contain new line characters - and span multiple lines.

One strategy for recognizing message boundaries in a log file are as follows:

- Any line that begins with a '[fmt=evt]' or '[fmt=symaud]' corresponds to a start of a new event.

Example 1 Examples: (continued)

- Any other lines correspond to continuations of the prior event - and should be appended to that, with a space replacing the new line that came between the two lines.

Format for messages written to Target = Syslog

A BSD-style prefix is included with the message before it is sent to a remote Syslog server. This prefix contains the following:

<PRI>	Priority (syslog_facility * 8 + syslog_severity)
Dec 17 10:33:20	Local Date/Time - without a Year. This is the time at which the event was sent to Syslog.
EMCstorevntd	Name of application (EMC Event Daemon)
:	The Header and Tag and terminated by a ':'

The date SDE (when the event was generated) will be UTC for a Syslog target - with a 'Z' suffix.

In the following examples, this prefix is shown in bold.

```
<11> Dec 17 10:33:20 EMCstorevntd: [fmt=evt] [evtid=1201]
    [date=2006-12-17T10:33:05Z] [symid=000000006190] [sev=fatal]
    = Array state has changed to Unknown.

<11>Jan  5 08:39:21 EMCstorevntd: [fmt=evt] [evtid=1200]
    [date=2007-01-05T08:39:05Z] [symid=000000006190]
    [Device=0007] [sev=major] = Device state has changed to Offline.
```

Notes:

- The Facility is LOG_USER (1).
  The Severity will be either LOG_CRIT (2), LOG_ERR (3), LOG_WARNING (4) or LOG_INFO (6).
- These messages contain two date/time fields.
  The first ('Dec 17 10:33:20') is called for by RFC 3164 (BSD Syslog): it is the local time that the event daemon sent the event to the remote Syslog server. As shown above, day numbers that are less than 10 (for example: Jan 5) are preceded by an extra space - as called for in RFC 3164.

  The second ('[date=2006-12-17T10:33:05]') is the time that the event was originally generated, in NG-Syslog format. In some cases, this will be in local time ... while in others (for example: events corresponding to the Audit log) these will be in UTC time ('Z' suffix). In most cases, this timestamps will be more meaningful than the BSD one at the front of the message.

- The application name 'EMCstorevntd' can serve an indicator that this originated from the EMC Event Daemon.
- In the sample event messages that are present in subsequent sections, new lines have been added to improve readability.

Format for messages written to Target = System (UNIX)

Messages sent to Syslog via the System Target have a prefix added by the platform syslog module - which may differ depending on the OS.

The following example was taken from a Solaris 2.8 desktop. The text in bold (before the fmt SDE) was added by the Solaris sylog logic.

```
Dec 17 10:33:20 l82ab139 storevntd[6881]: [ID 784156 user.error] [fmt=evt]
    [evtid=1201] [date=2006-12-17T10:33:05] [symid=000000006190]
    [sev=fatal] = Array state has changed to Unknown.
```

Notes:

- The facility is LOG_USER (1).
  The Severity will be either LOG_CRIT, LOG_ERR, LOG_WARNING or LOG_INFO.
- If syslog on the host is configured to forward across the network to a remote server (syslog.conf), the above will be prefixed by a "<PRI>" value.
- The '[6881]' field above is the process ID of the Event Daemon.
- The '[ID 784156 user.error]' field above is an extension added by Solaris. The '784156' serves as a message identifier - in this case, taken from some type of hash over the message.

Format for messages written to Target = System (Windows)

The message itself has the same format as what was shown above - no prefix is added.

Example:

```
[fmt=evt] [evtid=1201] [date=2006-12-17T10:33:05] [symid=000000006190]
    [sev=fatal] = Array state has changed to Unknown.
```

For the other attributes stored in the Windows event log:

- The Type will be ERROR, WARNING or INFORMATION.
- The Source will be storevntd.
- The Category will be Event.
- The Event ID will be 0.
- The User will be N/A.
- The Description is as shown above.

Format for messages written to Target = SNMP

The Event Daemon encodes SNMP traps according to the Fibre Channel Alliance MIB (version 3.0). These traps contain a number of fields (identified by OID) and values. The most relevant of these are the following - along with examples of values they might have.

SNMP trap ID (this is an integer)

This is the internal event ID. It is incremented for each event, ranging between 1 and *connUnitMaxEvents*. The default value for *connUnitMaxEvents* is 256. It is configurable by modifying the *snmp_event_table_size* value in the `daemon_options` file.

OID:	1.3.6.1.3.94.1.11.1.3
Name:	connUnitEventId
Value:	3

SNMP trap type (this is an integer)

OID:	1.3.6.1.3.94.1.11.1.7
Name:	connUnitEventType
Value:	1: unknown 2: other 3: status 4: configuration 5: topology

SNMP trap object (this is an OID)

OID:	1.3.6.1.3.94.1.11.1.8
Name:	connUnitEventObject
Value:	1.3.6.1.4.1.1139.1.3.5.4

Trap severity (this is an integer)

OID:	1.3.6.1.3.94.1.11.1.6
Name:	connUnitEventSeverity
Value:	8

Event Description (this is a string)

This description is a subset of the other formats shown above. One major difference is that the Entity and Component are formatted differently - not inside an SDE '[..]'.)

OID:	1.3.6.1.3.94.1.11.1.9
Name:	connUnitEventDescr
Value for Simple Event:	
	Symmetrix 000000006190 Device 0002 : Device state has changed to Online.
Value for an Audit Log Record Event:	

	Symmetrix 000000006190 : [orig=SE] [user=H:jupiter \jones] [host=saturn] [actid=SEb5d5129f28] [appid=Internal_Test] [aud-cls=Security] [aud-act=Add] [aud-num=40] = The User Authorization set role operation SUCCEEDED.

Event source

OID:	1.3.6.1.4.1.1139.3.8888.1.0
Name:	emcAsyncEventSource
Value:	1 = generated by the Event Daemon 2 = generated by the VMAX array

Event code

OID:	1.3.6.1.4.1.1139.3.8888.2.0
Name:	emcAsyncEventCode
Value:	These integers represent the event itself. For details on the events, refer to the VMAX Management Software Events and Alerts Guide.You can return a list of events and descriptionus using the command `stordaemon action storevntd -cmd list -events`.

Array component type to which the event corresponds

OID:	1.3.6.1.4.1.1139.3.8888.3.0
Name:	emcAsyncEventComponentType
Value:	Numeric value defined in Table 29 on page 173

Array component name to which the event corresponds to

OID:	1.3.6.1.4.1.1139.3.8888.4.0

Name:	emcAsyncEventComponentName
Value:	String value such as "0070", "SATAPool"

Table 29 on page 173 contains the possible values.

Table 29 Solutions Enabler event daemon event UID values

UID (integer value)	Component
1024	Symmetrix
1025	Service Processor
1026	Device
1027	Physical Disk
1028	Director
1029	Port
1030	SRDF sub-system
1031	SRDF group
1032	Snap Save Device Pool
1033	Cache / Memory
1034	Power or Battery subsystem
1035	Environmental (e.g.: Temperature, Smoke)
1036	Diagnostics
1037	Communications sub-system
1038	External Lock
1039	Fan
1040	Link Controller Card
1041	Enclosure, Enclosure-Slot or MIBE
1042	SRDF/A DSE Device Pool
1043	Thin Device Data Pool
1044	Solutions Enabler DG group
1045	Solutions Enabler CG group
1046	Management Module
1047	IO Module Carrier

Table 29 Solutions Enabler event daemon event UID values (continued)

UID (integer value)	Component
1048	Director - Environmental
1049	Storage Group
1050	Migration Session
1051	Symmetrix Disk Group

Event host

OID:	1.3.6.1.4.1.1139.3.8888.4.0
Value:	Actually name of the component effected, such as the disk ID or device name.

Miscellaneous options

The `daemon_options` file contains parameters (Table 30 on page 174) that allow you to configure a Syslog target.

Table 30 Event log file configuration options

Parameter	= <OptValue | defaultvalue>	Description
storevntd:log_event_network_pad	1 -10 | 0	Specifies the rate at which events are transmitted to the syslog or SNMP targets. Events are delivered to the targets using the UDP network protocol, for which certain recipient hosts (or network intermediaries) will drop messages if they arrive too quickly. This option defines how long to wait (in milliseconds) between event transmissions. Use this option carefully, as too large a value can result in an event delivery rate that cannot keep pace with the generation rate, which can lead to queue overflows (and even loss) within the event daemon. The default value of 0 means that there is no delay between transmissions.

Table 30 Event log file configuration options (continued)

Parameter	= <OptValue | defaultvalue>	Description
storevntd:symm_poll_interval	*nnn* | 60 (seconds)	Specifies how often the event daemon checks (polls) for events to transmit. Its value indicates how often the basic event polling loop runs, in seconds. The event daemon does not check for every type of event during every polling cycle. It checks for some events every 2 cycles, 3 cycles, 4 cycles, etc.
storevntd:symm_recovery_interval	*nn* | 30 (minutes)	Specifies the period of time until the recovery table becomes invalid. For events being automatically logged to syslog or SNMP by the event daemon, the event daemon loads a recovery table when it starts up in order to avoiding losing track of events when it was not running. This option defines how long the recovery table is considered valid for the event daemon to load on startup.

Test mode

Test mode is a convenient way for you to verify that the event daemon has been correctly configured. For example, if you wanted to see if you have configured the SNMP trap correctly, without the test mode, you would have to use `stordaemon setflt` to inject various events. However, such testing can also stress the VMAX array as event daemon will try to sync up the state from the array.

To test without stressing the array, test mode is provided in the event daemon. When test mode is enabled for the event daemon, it will not sync its state with the array.

This is accomplished by specifying a parameter in the `daemon_options` file:

```
storevntd:test_mode = ENABLE|DISABLE
```

The default value for this option is DISABLE. The option will not take effect on `stordaemon reload` command. The daemon needs to be restarted for any change to this option to take effect.

VSS Provider environment variables

Update the environment variable for path to include the Solutions Enabler installation directory, which by default is `C:\Program Files\EMC\SYMCLI\bin`, to run the command line utilities from any directory.

SMI-S Provider Windows authentication settings

To enable Windows authentication, you must modify default settings in the `security_settings.xml` file. On Windows platforms, this file resides in `c:\program files\emc\ecim\ecom\conf`.

To enable Windows authentication:

Procedure

1. If ECOM is running, stop it, as explained in Starting and stopping ECOM on page 179.
2. Modify the following default settings in `security_settings.xml`:

```
<ECOMSetting Name="NonCIMRequest_AuthenticationEnabled
"Type="boolean" Value="false"/>

<ECOMSetting Name="HTTPChallengeMechanism"
Type="string" Value="Basic"/>
```

   to:

```
<ECOMSetting Name="NonCIMRequest_AuthenticationEnabled
"Type="boolean" Value="true"/>

<ECOMSetting Name="HTTPChallengeMechanism"
Type="string" Value="Basic,WindowsAuth"/>
```

3. Restart ECOM.

VMAX arrays

When using the SMI-S Provider to manage VMAX arrays, it is recommended that you configure six gatekeepers for each array accessed by the provider. Only set up these gatekeepers for the host on which the SMI-S Provider is running. When started, the SMI-S Provider automatically discovers all arrays connected to the host on which the Array Provider is running. No other action is required, such as running the `symcfg discover` command.

When deploying the SMI-S Provider for VMAX arrays, ensure that only the arrays that will be managed by the provider are made visible to the SMI-S Provider.

As part of the Solutions Enabler discovery of VMAX arrays, those arrays that are SRDF connected to the local array being discovered will also be discovered.

If your client application only manages local arrays please symavoid these remote storage systems by creating a file called `symavoid` in `c:\program files\emc\symapi\config` on Windows or `/var/symapi/config` on Linux. In the file place the Symmetrix ID of the system to be avoided, one ID per line. The file should be named just `symavoid` - ensure it doesn't have any extension such as

`symavoid.txt`. Once the file is in place shut down ECOM and remove the file `symapi_db.bin` from `c:\program files\emc\symapi\db` on Windows or `/var/symapi/db` on Linux and the startup ECOM.

Doing this reduces unnecessary syscall traffic which would otherwise be consuming SRDF link resources.

ECC and Unisphere for VMAX 1.0 coexistence: symapi_db.bin database sharing

When the SMI-S Provider is installed on the same host as the ECC Symmetrix agent and/or the Unisphere for VMAX 1.0, you may see the following memory allocation errors in the `syampi` log file:

```
EMC:SMBASE __iload_db_osl pdsDbRecRead() failed : OSL:CONN_INFO ([PDS/DB] (Unable to
allocate memory)

EMC:SMBASE emcSymDBLoad Error encountered while reading from DB file [C:\Program Files\EMC
\SYMAPI\db\symapi_db.bin] (SYMAPI_C_MEMORY_ALLOC_ERROR)
```

The factors determining these memory allocation errors are governed by the amount of physical memory on the host as well as the number and size of the array configurations. Because it is difficult to predict how much memory is required for this type of installation scenario, perform the following steps to prevent the above errors from occurring:

1. Instruct SMI-S Provider to use its own `symapi` database by editing the `c:\program files\emc\ecim\ecom\providers\oslsprovider.conf` file.
2. Change the following line in `oslsprovider.conf`:

```
#OSLSProvider/com.emc.cmp.osls.se.array.StorApi.database.filename =
```

to:

```
OSLSProvider/com.emc.cmp.osls.se.array.StorApi.database.filename = c:/program files/emc/
symapi/db/symapi_smi_db.bin
```

3. Stop ECOM, the ECC Symmetrix agents, Unisphere for VMAX 1.0, and the Solutions Enabler daemons.
4. Remove the existing `symapi_db.bin` file, and save all device group information to be later restored to the new `symapi` database.
5. Restart ECOM, the ECC Symmetrix agents, Unisphere for VMAX 1.0, and the Solutions Enabler daemons.

ECOM

The ECOM post-installation tasks require that you set up an administrator role, supply certificates to both the ECOM server and its client, and then start ECOM.

Setting up administrator authentication

Authentication is required to query the EMC CIM Server. An initial setup is required on the EMC CIM Server to create a CIM user. This can be done as follows:

Procedure

1. Go to the URL `https://<ipaddress>:5989/ecomconfig`, and log in using the username `admin` and the password `#1Password`.
2. Click **Add User** and create a user with the role of **Administrator**. This newly created username can now be used to obtain access to the SMI-S Provider.

   Note

   For security reasons, change the default password of the user `admin`.

ECOM certificate management

In order for SSL communications between two peers to be authenticated, one of the following conditions must exist:

- If a peer presents a self-signed certificate, the host receiving the self-signed certificate must have its trust store seeded with that certificate.
- If a peer presents a CA-signed certificate, the host receiving the CA-signed certificate must have its trust store seeded with a chain of certificates starting from the issuer of the peer's certificate and ending with the root certificate.

Installing certificates in trust stores is performed at configuration time, not at runtime. The following sections describe how to supply certificates to both the ECOM server and its client.

Supplying a client with the ECOM server certificate

Procedure

1. Obtain the ECOM certificate (`ecomtls.crt`) from the directory `<ECOM_Home\conf\ssl`.
2. If `ecomtls.crt` does not exist, point your browser to the ECOM Admin page `https://<server>:<port>/ECOMConfig`. The connection fails as the trust store Is not yet set up but the certificate is generated.
3. Add the ECOM certificate (`ecomtls.crt`) to the client's trust store. The certificate is in PEM format.

Supplying ECOM with the client certificate

To authenticate the client certificate, you must import the client certificate into the ECOM trust store. To do this, you must append the certificate to the file `ecomtls.ca` found in the directory `<ECOM_HOME>\conf\ssl`.

Follow these steps:

Procedure

1. Obtain the client certificate from an SSL certificate provider.

   Note

   ECOM accepts certificates in PEM format only at this time.

2. Point your browser to the ECOM Administration Login page:

   `https://<ServerName>:5989/ECOMConfig`

3. Select the **SSL Certificate Management** submenu.
4. Select **Import CA certificate file** to import the certificate. You do this by cut/pasting the certificate to the end of the list of already existing certificates if any exist.
5. Re-start ECOM.

Starting and stopping ECOM

ECOM runs on both Windows and UNIX environments. After installation completion, ECOM automatically starts. You can use the following commands to manually stop and restart the service should the need arise.

ECOM failure to start

If ECOM does not start, review the problem resolutions in the following sections.

Security initialization failure
Red Hat and SuSE Linux platforms may generate the following set of errors when ECOM does not start:

```
02-Nov-2010 15:09:52.091 -3086366416-W- ECOM: CST Lockbox Initialization
Error:ERR_LIB_NOT_INIT

02-Nov-2010 15:09:52.091 -3086366416-C- ECOM: -E- Security manager initialization failed.
Check whether the security plugin exists and is set up properly.
```

If you receive the above errors, complete the following steps:

1. Change directory to `/opt/emc/ECIM/ECOM/thirdparty` and issue the following command:

```
[root@losaz134 thirdparty]# ./cstadmin initialize /opt/emc/ECIM/
ECOM/conf/cst
```

2. A request for a lockbox passphrase displays. Enter a text string for the passphrase:

```
Enter lockbox passphrase:
Confirm passphrase:
```

Unsupported SELinux setting is enabled

The following error indicates an unsupported SELinux setting is enabled, which is the default for Red Hat, and must be disabled:

```
cstadmin: Failure initializing lockbox
/opt/emc/ECIM/ECOM/conf/cst. [The cryptography library was not initialized.] [-48]
Failed to retrieve Log Service: The cryptography library was not initialized. [/opt/emc/
ECIM/ECOM/conf/cst/csp.clb]
```

To temporarily disable this SELinux setting, complete the following steps:

```
[root@losaz134 ~]# cat /selinux/enforce
1
[root@losaz134 ~]# echo 0 >/selinux/enforce
[root@losaz134 ~]# cat /selinux/enforce
0
[root@losaz134 ~]# cd /etc
[root@losaz134 etc]# cd selinux
```

To permanently disable this SELinux setting, follow the instructions at:

http://www.crypt.gen.nz/selinux/disable_selinux.html

Windows

On Windows, ECOM runs as a service and can be controlled through the Windows **Services** control panel. The service name is `ECOM.exe` and it displays as `ECOM` in the **Services** control panel.

As an alternative method for stopping and starting ECOM, the `ECOM.exe` file is located in the Solutions Enabler `C:/Program Files/EMC/ECIM/ECOM/bin` directory. Use the following command to start the EMC CIM Server:

```
sm_service start ecom.exe
```

Use the following command to stop ECOM:

```
sm_service stop ecom.exe
```

UNIX

On UNIX, ECOM runs as a daemon in the background. To stop ECOM, obtain the PID of the ECOM process and issue the `kill -SIGTERM` command for that PID. For example:

```
kill -SIGTERM [PID]
```

The ECOM executable file is located in the Solutions Enabler `/opt/emc/ECIM/ECOM/bin` directory. Use the following command from this directory to restart ECOM:

```
./ECOM -d
```

Disabling ports

After installation, ports 5985, 5988 and 5993 are not encrypted using SSL. These ports can be disabled by modifying the file `port_settings.xml` which is located in `C:\Program Files\emc\ecim\ecom\conf` on Windows, and in `/opt/emc/ECIM/ECOM/conf` on Linux.

By default, the following entry is shown in the file:

```
<ECOMSettings>
<ECOMSetting Name="Port0">
<!--
  <portRange>5988</portRange>
-->
  <port>5988</port>
  <secure>false</secure>
  <slp>true</slp>
</ECOMSetting>

<ECOMSetting Name="Port2">
<!--
  <portRange>5985</portRange>
-->
   <port>5985</port>
   <secure>false</secure>
   <slp>true</slp>
</ECOMSetting>

<ECOMSetting Name="Port4">
  <!--
     <portRange>5993</portRange>
  -->
  <port>5993</port>
  <secure>false</secure>
  <slp>true</slp>
</ECOMSetting>
```

To block these ports from being setup by ECOM, make the changes as shown below: (please note the characters in bold that were moved down completely blocking the associated ports from being setup by ECOM).

```
<ECOMSettings>
<ECOMSetting Name="Port0">
<!--
  <portRange>5988</portRange>
  <port>5988</port>
  <secure>false</secure>
  <slp>true</slp>
</ECOMSetting>
-->

<ECOMSetting Name="Port2">
<!--
  <portRange>5985</portRange>
   <port>5985</port>
   <secure>false</secure>
   <slp>true</slp>
</ECOMSetting>
-->

<ECOMSetting Name="Port4">
  <!--
     <portRange>5993</portRange>
   <port>5993</port>
   <secure>false</secure>
   <slp>true</slp>
</ECOMSetting>
-->
```

Once these changes are made save the file and restart ECOM. As a result, ports 5985, 5988 and 5993 are no longer started.

SMI-S Provider runtime settings

The `OSLSProvider.conf` file allows you to control the runtime behavior of the SMI-S Provider. You can find this file in the following directories of the Solutions Enabler:

- Windows platforms: `C:/Program Files/EMC/ECIM/ECOM/Providers`
- UNIX platforms: `/opt/emc/ECIM/ECOM/providers`

Table 31 on page 182 describes the SMI-S Provider runtime settings. In order for these runtime settings to take effect, you must stop and then restart ECOM.

Table 31 SMI-S Provider runtime settings

SMI-S Provider properties[a]	= <*OptVal*|*DefaultVal*>	Description
OSLSProvider/ com.emc.cmp.osls.se. symm.SymApiService .database.discover	true | false	Specifies whether to perform a one-time discover upon starting a CIM Server. This is done before processing the first request received by the CIM Server.
*/ com.emc.cmp.ofl.log. Control.severity.id	FATAL, ERROR, WARNING, NOTICE, INFO	Specifies the severity levels for the event logs: FATAL — Events leading to shutdown of the system ERROR — Internal or client error conditions WARNING — Potential errors NOTICE — Very important information (default if not present) INFO — Informational, non-error messages Each setting causes messages of the set severity and more severe to be appended to the log.
#OSLSProvider/ com.emc.cmp.osls.se. symm.Session.All.con trols.enable	false | true	If false, disables all controls. A false setting takes precedence over all

Table 31 SMI-S Provider runtime settings (continued)

SMI-S Provider properties[a]	= <*OptVal*|*DefaultVal*>	Description
		control settings previously explained in this table.

a. The path shown is a UNIX-specific default installation path. Your actual install path may differ.

RedHat Enterprise Linux 6.0/6.2 [GA] - x86_64 installation

Solutions Enabler V8.3 installation requires i686 version of `glibc` (GNU C Library) and `libgcc` (Library of GCC support routines) packages pre-installed.

Example 2 RHEL 6.0

If your RHEL 6.0 (x86_64) host does not have `glibc` and `libgcc`, use the following commands to install `glibc` and `libgcc`:

```
# cd media/<RHEL_6.0 x86_64 Disc mount point>/Packages
# rpm -ivh glibc-2.12-1.7.el6.i686.rpm glibc-devel-2.12-1.7.el6.i686.rpm
nss-softokn-freebl-3.12.7-1.1.el6.i686.rpm libgcc-4.4.4-13.el6.i686.rpm
Preparing... ########################################### [100%]
1:libgcc ########################################### [ 25%]
2:nss-softokn-freebl ########################################### [ 50%]
3:glibc ########################################### [ 75%]
4:glibc-devel ########################################### [100%]
```

After the installation, query the rpm as shown below:

```
# rpm -qa | grep i686 | grep lib
glibc-devel-2.12-1.7.el6.i686
libgcc-4.4.4-13.el6.i686
glibc-2.12-1.7.el6.i686
# rpm -qa | grep i686 | grep nss
nss-softokn-freebl-3.12.7-1.1.el6.i686
```

Example 3 RHEL 6.2

If your RHEL 6.2 (x86_64) host does not have `glibc` and `libgcc`, use the following commands to install `glibc` and `libgcc`:

```
# cd media/<RHEL_6.2 x86_64 Disc mount point>/Packages
# rpm -ivh glibc-2.12-1.47.el6.i686.rpm nss-softokn-freebl-3.12.9-11.el6.i686.rpm
Preparing... ########################################### [100%]
1:nss-softokn-freebl ########################################### [ 50%]
```

Example 3 RHEL 6.2 (continued)

```
2:glibc ########################################### [100%]
# rpm -ivh libgcc-4.4.6-3.el6.i686.rpm
Preparing... ########################################### [100%]
1:libgcc ########################################### [100%]
```

After the installation, query the rpm as shown below:

```
# rpm -qa | grep i686 | grep lib
libgcc-4.4.6-3.el6.i686
glibc-2.12-1.47.el6.i686
# rpm -qa | grep i686 | grep nss
nss-softokn-freebl-3.12.9-11.el6.i686
```

Adding the SSL certificate

If the "cert" component is not initially installed, and then added (by running the installer again) or by performing an incremental install, on AIX and Linux platforms, the SSL certificate is not created.

You can create the SSL certificate by entering the following:

```
# cd /var/symapi/config/cert
# /usr/symcli/bin/manage_server_cert create -pass <lockbox_pwd>
```

where

`<lockbox_pwd>` is the lockbox password that was used during the installation.

Vendor SNIA libraries needed for HBA information

There are certain SNIA libraries (Emulex or Qlogic) which need to be installed so that Solutions Enabler CLI can obtain host HBA information. By default, SNIA libraries are not pre-installed on the host. Follow these steps to install the SNIA libraries:

Procedure

1. Find the vendor information and model.

   ```
   ESI144:~ # cat /sys/class/fc_host/host1/symbolic_name
   Emulex LPe12002-M8 FV2.00A4 DV8.3.5.8.1p
   ESI144:~ #
   ```

2. Open the Emulex download page (http://www.emulex.com/downloads.html) and select **EMC**.
3. Select the specific version identified in step 1 (**LPe12002**) from **Fibre Channel Host Bus Adapters...** section. This opens the **EMC Qualified Downloads and Documentation** page.
4. Select the **Drivers** tab and select the **Operating System** and **version**. This selection opens the **Downloads** page.

5. Select the **Management and Utilities** tab and download the Application Kit **6.0.9.1-1 (CLI)** from the **UCNA and HBA Application Kit** section.
6. Install the application kit.

   Upon successful installation, `/etc/hba.conf` will be created (if the file doesn't exist) and will have the following entry:

```
ESI144:~ # cat /etc/hba.conf
com.emulex.emulexapilibrary /usr/lib64/libemulexhbaapi.so
com.emulex.emulexapilibrary /usr/lib/libemulexhbaapi.so
ESI144:~ #
```

Note

Repeat the same steps for each operating system type. If the host has Qlogic, follow similar steps from the https://support.qlogic.com/ **Downloads** page.

z/OS Post installation configuration

Once you have installed Solutions Enabler, you need to perform certain follow-up procedures to enable your software's features and to establish your command environment. This chapter provides the follow-up procedures for a Solutions Enabler installation in a z/OS mainframe environment.

SYMAPI server security preparation

This section explains how to control access to the SYMAPI server.

Started task user identity

The SYMAPI server is installed to be run as a batch job, but you can also customize it to run as a started task.

If you choose to run the server as a started task, you must associate a user identity with it. You can assign a user identity to the server using the `RDEFINE` command or the started task table ICHRIN03. An example of the **`RDEFINE`** command is shown below assigning the user `STORSRVD` to all started tasks whose names start with STORSRVD:

```
RDEFINE STARTED STORSRVD.* UACC(NONE) STDATA(USER(STORSRVD))
OWNER(SYS1)
```

If you use the ICHRIN03 table to associate started task names with user identities, refer to the IBM publication *Security Server RACF System Programmer's Guide* for details on preparing this table.

SSL certificates

Solutions Enabler optionally allows the use of SSL encrypted communications between the SYMAPI server and the clients connecting to it. You can configure the server to allow client sessions without SSL, or to require SSL sessions. Client configuration to use SSL or not must match the server configuration.

If you plan on using the optional SSL encrypted communications and you plan on running the server in SECURE or ANY modes, you must create and install the SSL certificates before starting the server.

Note

For information on configuring the security level on the server side, refer to the *EMC VMAX Family Security Configuration Guide*

Note

You must have run job `#07DFLTS` before the following steps can be taken. Job `#07DFLTS` creates requisite directories in the UNIX System Services filesystem.

Installing SSL certificates

To install SSL certificates into the certificate store created by the `#07DFLTS` job, you must visit the Windows machine where you initiated the z/OS installation, and then follow these steps:

Procedure

1. Change to the temporary directory where you ran the `uploadse.bat` command.
2. Run the batch file `zoscert.bat` with the `create` parameter in the temporary directory you created on the Windows host in Step 1: Copy the files on page 75.

   For example:

   ```
   zoscert create
   ```

   Note

   The `zoscert.bat` script requires that the Microsoft Visual Studio 2012 redistributable runtime library is installed. If this library is not installed, it will be automatically installed as part of the certificate generation process. The library will not be removed after the installation is complete. If you wish to remove the runtime library after successfully generating the certificate, you can do so by using the **Add or Remove Programs** function from the Windows Control Panel.

3. When prompted, provide the following information:
   - The fully qualified name of the z/OS host (hostname including the domain name). This is the same name as you specified when running the `uploadse.bat` command.

Note

In the case of multi-homed hosts, more than one fully qualified hostname may be specified, separated by spaces, in response to the prompt for the hostnames. If you enter more than one hostname at the host prompt, the first name will be used as the Common Name in the certificates, and all names after the first are used in the Subject Alternative Names. You may specify IP addresses in addition to host names for either the Common Name or Subject Alternative Names. The first name entered is also the target address of the FTP command used to send certificates to the mainframe.

- The FTP port number (default 21) of the z/OS host.
- The z/OS userid for sign in to the FTP service on the mainframe. The user must have write permission to the SYMAPI base directory and all subdirectories.
- The SYMAPI base directory (specified when running the SEMJCL exec on z/OS).
- The password for the z/OS userid.

Once generated, the certificates will be uploaded to the correct location inside the Unix System Services file system on the z/OS host. For example, if you specified the SYMAPI base directory as `/var/symapi`, the certificates will be uploaded to the directory `/var/symapi/config/cert`.

The certificate configuration is now complete and the server is capable of running in a secure mode.

Note

For more information on certificate management, refer to the *EMC VMAX Family Security Configuration Guide.*

Configuring Solutions Enabler

This section explains how to configure Solutions Enabler in a z/OS environment.

SYMAPI database support

Solutions Enabler for z/OS supports the SYMAPI database and all the associated access modes. Solutions Enabler will refer to the database (or create one if it doesn't exist) in the *symapi_installation_directory*`/db` directory in Unix System Services.

A SYMAPI application can specify the database by providing a name associated with the database using the following formats:

```
/path/to/db.file
```

where:

/path/to is a valid, existing, writable Unix System Services path and `db.file` is the name of the SYMAPI database.

Solutions Enabler uses the following conventions to identify the database that it will associate with a particular session. The SYMAPI application specifies the database name in the SymInit( ) function call:

- As the database default name (by specifying NULL in the database argument)
- With an explicit database name

Note

If an explicit location is specified for the database, SYMAPI will use it; otherwise, specifying just a filename will result in the file being stored in the *symapi_installation_directory*/db directory.

Server default database locking

The default database is described in the fully qualified Unix System Services path of the database. When a session requests the default database, SYMAPI attempts to use the fully qualified Unix System Services path, handling locking for read-only and read/write sessions appropriately. If the session obtains database locks successfully, SYMAPI loads the database for the session in the mode (read-only, or read/write) desired.

Multiple users can share a database file in a read-only and read/write mode. Write integrity to the database is guaranteed by internal locking mechanisms. No two sessions can request read/write mode concurrently.

Once a read/write session has been started, the SYMAPI server will prevent multiple read/write sessions by failing to initialize subsequent SymInit() requests, or by blocking them until the first read/write session releases the database.

Note that the locking behavior applies to the fully qualified path.

Gatekeeper devices

The use of gatekeeper-defined devices in a VMAX array configuration does not apply to z/OS installations. However, z/OS servers do communicate to the system using a UCB on the first device found in the storage array. The SYMAPI protocol selects the first on-line device as its gatekeeper. It is possible that this auto-select mechanism may not always be appropriate. For example, you may not want to have the system paging device or a JES SPOOL volume selected as the communication portal. The high I/O rate produced from the SYMAPI may adversely affect system performance. To control gatekeeper use by the SYMAPI server tasks, you can define specific devices to be used as gatekeepers, and also specify devices to be avoided as gatekeepers.

Note

For more information on gatekeepers, refer to Gatekeeper Device Configuration on page 249. For more information on specifying devices to use/avoid from using as gatekeepers, refer to Avoidance and selection files on page 190.

SYMAPI files

Table 32 on page 189 lists and maps the SYMAPI files to corresponding DD statements. It also shows which files can be defined in PARMLIB members or in datasets, and which files can optionally be defined in Unix System Services files.

Note

For Unix System Services supported files, SYMAPI will only use a Unix System Services location if the corresponding DD name is not specified in the SYMAPI server JCL (comment it out or delete it).

Table 32 SYMAPI files

DD name	File type	Description
SYM$LIC	Unix System Services	An input file for the Solutions Enabler license information. Unix System Services: *symapi_installation_directory*/config/symapi_licenses.dat
SYM$OPT	Unix System Services	The SYMAPI options file. For more information, refer to Changing the default behavior of SYMCLI on page 137. Unix System Services: *symapi_installation_directory*/config/options
SYM$ENV	PARMLIB, Dataset	Contains the C runtime environment variables. This file must be either a sequential dataset or a member of a partitioned dataset. This file must only be used with the direction of the EMC Customer Support Center. PARMLIB: *ds-prefix*.PARMLIB(symenv00)
SYM$NETH	Unix System Services	Defines a list of trusted hosts and users who are allowed to connect to the server. For more information, refer to the *EMC VMAX Family Security Configuration Guide*. Unix System Services:

Table 32 SYMAPI files (continued)

DD name	File type	Description
		symapi_installation_directory/config/nethost
SYSOUT	Spool	Contains IBM Language Environment runtime messages.
SYSPRINT	Spool	Contains summary log output and output produced by the use of debugging controls.

Avoidance and selection files

Table 33 on page 190 lists the these files in the UNIX file system.

Note

From V7.6, Solutions Enabler no longer supports avoidance and selection files in JCL. Non-configuration specific files (such as SYM$ENV) that are unique to z/OS, and have no Unix System Services equivalent are still supported via JCL.

Should an unsupported DD statement be used, Syminit will fail with the error SYMAPI_C_FILE_TYPE_NOT_SUPPORTED.

These files can be used to customize and streamline command line coding for your specific environment.

These are editable files with device names or array IDs that you use to limit the effect of commands to include or exclude the specified devices, gatekeepers, or VMAX arrays. The files hold either volume serial names (*volser*) or array IDs (*Symmids*) with line entries having only one device name or ID per line. Lines beginning with a # (comment) are ignored.

Table 33 Solutions Enabler avoidance and selection files

DD name	File type	Description
SYM$AVD	Unix System Services only	JCL DD statement is not supported. For example, to avoid discovery of the storage array with a serial number of 0000183600186, code the serial

Table 33 Solutions Enabler avoidance and selection files (continued)

DD name	File type	Description
		number in the following file: Unix System Services: *symapi_installation_directory/*`config/symavoid` This file affects the operation of the discovery process so that it skips devices that belong to the VMAX arrays identified in this file. This may be useful if there are multiple VMAX arrays connected to the host that you wish the discovery to avoid. The avoidance file is formatted with 12-character array IDs, with one ID per line.
SYM$INQ	Unix System Services only	JCL DD statement is not supported. For example, to include information on volume ABC123 (only) and the array to which it is attached, code the volume serial number in the following file: Unix System Services:

Table 33 Solutions Enabler avoidance and selection files (continued)

DD name	File type	Description
		*symapi_installation_directory/*config/inqfile This file affects the inquiry and discovery processes so that they find only the volume serial name (volser) specified in this file. This maybe useful if you want to limit the command(s) to affect only certain VMAX array devices from your host. The inquiry file is formatted with volume serial names (volser), with one volser per line.
SYM$GAVD	Unix System Services only	JCL DD statement is not supported. For example, to instruct Solutions Enabler for z/OS to avoid using volume DEF456 as a gatekeeper device, code its serial number in the following file: Unix System Services: *symapi_installation_directory/*

Table 33 Solutions Enabler avoidance and selection files (continued)

DD name	File type	Description
		`config/gkavoid` This file affects calls to commands that use a gatekeeper to communicate to a VMAX array. A gatekeeper whose volser matches any of the entries specified in the gkavoid file will not be chosen as a gatekeeper to communicate with the VMAX array. This could be useful to designate certain VMAX array devices that should not be used as gatekeepers. The gatekeeper avoidance file is formatted with volume serial names (volser), with one per line.
SYM$GSEL	Unix System Services only	JCL DD statement is not supported. In SYM$GSEL, specify serials for the volumes you prefer to be gatekeepers. Specify one volume serial per line, with no other text on the line.

Table 33 Solutions Enabler avoidance and selection files (continued)

DD name	File type	Description
		Note If a SYM$GSEL list is not defined for a particular VMAX array or if the specified volumes to do not exist at the time the file is read (every time a CLI command is run), then normal gatekeeper selection rules will apply for that storage array. If you specify a volume serial in both the SYM $GAVD and the SYM$GSEL, the entry in SYM $GAVD takes precedence. Thus, SYM $GSEL creates a limited list of candidate gatekeepers, and SYM$GAVD further restricts the list by removing volumes from the candidate list. If you specify a gatekeeper selection list in SYM$GSEL, be sure to specify at least one volume on each system you want to access

Table 33 Solutions Enabler avoidance and selection files (continued)

DD name	File type	Description
		through Solutions Enabler. For example, to instruct Solutions Enabler to give preference to volumes GHI123, JKL123 and MNO123, code their serial number in the following file: Unix System Services: *symapi_installation_directory/*`config/gkselect` **Note** If you specify a volume in BOTH the SYM$GSEL and SYM $GAVD, the entry in SYM $GAVD takes precedence, effectively removing the volume from the list of potential gatekeepers. Thus, if the volume DEF456 also appeared in SYM$GSEL, its entry in SYM $GAVD (see example above) cancels its participation in gatekeeper selection.

Configuring for local time zone

The SYMAPI server software uses IBM Language Environment runtime library, and must execute with the LE option POSIX(ON). One of the side effects of running with POSIX(ON) is that the local time displays are influenced by the POSIX time semantic definitions. The default behavior defined by POSIX for local time interpretation may not fit your operation.

You can use the TZ environment variable to cause LE to display local time properly. There are several places where time stamps are displayed — the `storsrvd` log files and SYMAPI log file are the most important places. Use the TZ environment variable to establish your local offset from Coordinated Universal Time (UTC). The valid settings for TZ are standardized by the POSIX standard and are described in many publications, including the IBM Language Environment books.

In the PARMLIB member `SYMENV00`, you can set TZ. The sample setting in the distributed member causes the local time zone to be set to United States Eastern Standard Time, offset five hours from UTC (also known as Greenwich Mean Time or GMT), and EDT time may apply. The following example shows the same specification using an Instream dataset set for SYM$ENV:

```
//SYM$ENV  DD *
TZ=EST5EDT
/*
```

In the **Time Zone** field of the SEMJCL panel (4. on page 70), you can enter the appropriate setting for your time zone. Installing Solutions Enabler on z/OS on page 75 includes more information.

Note

Due to the way Language Environment processes a TZ variable passed in by SYM $ENV, a TZ variable with no DST in the string results in exactly the same time as a TZ variable with DST. For example, the variable MST7 will be processed the same as MST7DST and will have the same resultant time zone.

To workaround this, for any of the z/OS daemons, the TZ variable should be specified as part of the PARM on the EXEC DD statement. For example:

```
//STORSRVD EXEC PGM=STORSRVD,REGION=0M,
//          PARM='ENVAR(TZ=MST7)/'
```

Modifying default behavior with the options file

The `options` file contains statements that can be modified to change the default behavior of SYMCLI operations, SYMAPI calls, and their control actions. It can be used to impart certain global restrictions as well as customize and streamline command line coding to your specific environment. Each sample statement is commented, and can be enabled by removing the # in the first column.

Note

For descriptions of the `options` file parameters, refer to *EMC Solutions Enabler SYMCLI Command Reference Guide*.

Remote control operations

Remote control operations can be executed by the SYMAPI server on behalf of remote clients such as SYMCLI, or Unisphere for VMAX.

Restricting remote control operations

Remote control operations are enabled by default. Proceed only if you want to restrict certain remote control operations.

Remote control operations brings convenience but at the same time may also impact user data or system operation negatively. For that reason, you may wish to restrict the use of certain remote operations.

Table 34 on page 197 lists some of the control operations that can be disabled in the z/OS server.

Table 34 Examples of z/OS control operations

Function	Action
SymAccessSessionStart	Starts an access control session.
SymAuthzRuleDelete	Maintains internal authorization rules.
SymAuthzRuleUpdate	Updates internal authorization rules.
SymCgControl	Controls Consistency Groups.
SymCgBcvControl	Invokes a BCV control operation affecting all standard devices in a composite group.
SymCgRdfControl	Invokes an RDF control operation affecting all remotely mirrored RDF standard and R1 BCV devices in a composite group.
SymConfigChangeSessionStart	Starts a configuration change session.
SymDevBcvControl	Invokes a BCV control operation on the specified standard device and the specified BCV device.
SymDevControl	Invokes a basic operation on one or all devices that meet a specified selection criteria.
SymDevListBcvControl	Invokes a BCV control operation on a specified list of standard and BCV devices.
SymDevListControl	Invokes a basic operation on a list of devices that meet a specified selection criteria.
SymDevListRdfControl	Invokes an RDF control action on a list of devices.

Table 34 Examples of z/OS control operations (continued)

Function	Action
SymDgBcvControl	Invokes a BCV control operation affecting all standard devices in a device group, which has one or more associated BCV device.
SymDgControl	Invokes a basic control operation affecting all standard, or optionally all BCV, devices in a device group.
SymDgRdfControl	Invokes an RDF control operation affecting all remotely mirrored standard or RDF R1 BCV devices in a device group.
SymDirControl	Invokes a director control operation on one or all SRDF RA directors.
SymDirPortControl	Invokes a port control operation on a front-end director.
SymLdevBcvControl	Invokes a BCV control operation affecting one standard device in a device group, which has one or more associated BCV devices.
SymLdevControl	Invokes a basic control operation on a device in a device group.
SymLdevListBcvControl	Performs a BCV control operation affecting a list of standard devices in a device group.
SymLdevListControl	Executes a basic operation affecting the specified list of standard devices or BCV devices of a group.
SymLdevListRdfControl	Invokes an RDF control operation affecting one remotely mirrored standard device, or one or more RDF R1 BCV devices in a device group.
SymListDevListBcvControl	Invokes a single BCV or Snap control operation on a structure or array.
SymNewCgControl	Invokes a basic control operation affecting devices of a specified type within a specific composite group.
SymNewOptmzrControl	Invokes control operations on the Optimizer.

The control operations can be disabled by executing the job in the `#12CNTRL` member in the RIMLIB dataset. That job executes the AMASPZAP utility to change entries in a control table. Each entry in the table corresponds to one of the control operations

listed above. The comments in the AMASPZAP input indicate the relationship of the zap to the operation.

Control statements

The entries in the control table are mostly VER statements and REP statements grouped together respectively. A VER or VERIFY statement is composed of the command phrase VER, a hexadecimal address and an eight-byte hexadecimal value. The following is an example:

```
VER  0001D8 0000,0000
```

The VER statement checks to see if the value at the address given is the same as the value provided in the statement. If true, then the following statement will be executed. If not, the following statements will be ignored and job `#12CNTRL` will quit.

A REP statement is composed of the command phrase REP, a hexadecimal address and an eight-byte hexadecimal value. The following is an example:

```
REP 0001D8 0000,0001
```

The REP statement replace the current value at the given address with the value provided in the statement.

Note

HINT: Make a copy of member `#12CNTRL` for backup purposes before making any changes.

Modifying the control table

Job `#12CNTRL` is customized during the SEMJCL process, but does require a manual edit by the submitter before it can be used because it contains an invalid VER statement to force failure. This VER statement should be commented out or removed:

```
VER  0001D8 READ,DOC   COMMENT OUT THIS LINE TO RUN THE JOB
```

This invalid VER statement provides additional protection against accidental disabling of control operations. No change will take place if the job is submitted without making any changes.

Once the invalid VER statement is removed, the first entry in the table provides the capability to enable or disable control operations listed in Table 34 on page 197 as a whole. The following is how the first VER entry in the control table is configured by default:

```
VER  0001D8 0000,0000   IF ALL 0, CONTROLS ARE ENABLED
```

This statement verifies the value at address `0001D8`. If it is 0, that means Solutions Enabler does not check individual control operations. It simply allows all remote control operations.

To enable checking of individual operations, simply find the REP statement with the same address, `0001D8`; remove the leading asterisk to uncomment the statement and change the value following the address to `0000,0001`.

This effectively disables all control operations because you have just enabled checking of individual operations and all of them are set to disable by default.

To enable selective operations, find the REP statement with the same address as the VER statement for the desired operations, remove the leading asterisk, and change the value of the REP statement to `0000,0000`.

Note

HINT: Use the backup copy of the job as a reference.

For example, if you want to enable remote director control:

1. Find the VER statement for director control using the comment:

```
VER  0001F8 0000,0870   DIRECTOR CONTROL
```

2. Find the REP statement with the address 0001F8:

```
*REP  0001F8 0000,0870
```

3. Remove the leading asterisk to uncomment the statement and change the value from `0000,0870` to `0000,0000`.
4. Save job `#12CNTRL`.

Repeat these steps for each control operation you want to enable.

⚠ WARNING

Running multiple iterations of `#12CNTRL` could get the table into state where there are VERs failing due to prior changes, so plan accordingly by keeping an pristine backup copy of `#12CNTRL`.

Additional Work

In addition to executing the `#12CNTRL` member, the `SYMAPI_CTRL_VIA_SERVER` option can be set to `ENABLE` or `DISABLE`. The default value of the option is `ENABLE`, which corresponds to the `#12CNTRL` setting.

If you want to enable or disable control operations, you must:

- Verify that the `SYMAPI_CTRL_VIA_SERVER` option is set to `ENABLE` or `DISABLE`.
  Or
- Edit the `#12CNTRL` member in the RIMLIB as previously discussed.

⚠ CAUTION

By leaving control operations enabled, you enable open systems users to make changes to the array configuration on your mainframe system.

You may undo the changes you made using `#12CNTRL` by reversing any VER and REP changes and resubmitting the job.

Note

The server will need to be restarted if any `#12CNTRL` changes are applied.

Controlling the server

You can inspect and control the behavior of the server using the `stordaemon` command or the system console. For information on the commands accepted by the SYMAPI server, refer to Controlling the server on page 219.

This section describes specific methods of entering the commands.

Starting the server

To start the SYMAPI server, you can submit the job stream contained in the `#STORSRV` member of the Solutions Enabler RIMLIB for batch execution.

Note

`#STORSRV` was customized when you used SEMJCL to specify configuration information appropriate for your site during the installation procedure.

You can execute the SYMAPI server program `storsrvd` as a started task. You can prepare a catalogued procedure for use as a started task. No such procedure is provided with the installation kit.

You cannot use `stordaemon start` in the z/OS environment to start the server.

Stopping the server

To stop the SYMAPI server, you can use the `stordaemon shutdown` command, or the equivalent command from the z/OS system console.

You can also use the z/OS `STOP` command regardless of whether the server is running as a started task or as a batch job. Using the `STOP` command (for example, `"P STORSRVD"`) starts a normal shutdown, waiting for all SYMAPI sessions to terminate normally.

Using the console

You can control the SYMAPI server while it is running by issuing operator commands using the the z/OS system command **`MODIFY`** (abbreviated `F`):

```
F jobname,command
```

where:

jobname is the name of the batch job or started task under which the SYMAPI server is running.

command is the text of the command passed to SYMAPI server.

Usage notes

When issuing commands from the system console, you should be aware of the following:

- While `stordaemon` commands are sent to the daemons without upper case conversion, text entered on the system console (and all virtualized consoles) is normally folded to uppercase by the operating system. Enclosing the text in apostrophes (not quotes) alters the behavior, resulting in the command text being sent as is to the application.

- Commands issued using the `stordaemon action` verb must be entered with apostrophes to preserve the case. Complete enclosure in apostrophes is not necessary; a leading apostrophe is sufficient to preserve case. A closing apostrophe will be accepted and ignored.
- Dashed options are not required. The SYMAPI server allows the specification or omission of the dash on the command options. The console command parsing logic will accept a dash if specified, but ignore it for the purposes of option identification.
- Commands entered from the console are directed to a specific running daemon. Thus the multi-daemon commands and operands are not supported when entered from the console. The `list` command and the `all` option of the `shutdown`, `setvar`, `getvar` commands are not supported when entered at the console.
- The daemon name must be omitted in the command text, since the `MODIFY` system command specifies the jobname which directs the command to the correct daemon. Thus, the command text will begin with the verb.
- The `action` verb can be omitted only if the `-cmd` verb and/or operands can unambiguously distinguish the command from all general commands. For example, in the case of `storsrvd` , the general `show` command will show basic status information. The action `-cmd show` command will show other detailed information specific to `storsrvd`.
- The `-cmd` option can be omitted also. If either `action` or `-cmd` are specified, the command text will be passed to the running daemon for execution. If the daemon application log parses the command text successfully, it may execute the command and produce the appropriate output. If the application logic does not recognize the command, an error message will be generated and written to the console.
- Commands that change the environment outside of the daemon will not be accepted from the console. These are `start`, `install`, and `uninstall`.
- The `-wait` option of the `stordaemon shutdown` command is not supported and will be ignored if entered from the console.
- The `showlog` command is not supported from the console.

Examples

Table 35 on page 202 compares the syntax of the `stordaemon` commands issued from a Unix System Services shell to the syntax of the same commands entered on the z/OS console. Assume that the jobname of the server is `STORSRVD`, and the daemon name is also `storsrvd`. Note that the z/OS system command `MODIFY` alias is `'F'`.

Table 35 stordaemon command syntax for the z/OS system console

Command	stordaemon syntax	Console syntax
Show daemon status long. Show daemon status (state).	`stordaemon show storsrvd` `stordaemon show storsrvd -brief`	`F STORSRVD,SHOW` `F STORSRVD,SHOW [-]BRIef`
Stop the daemon. Stop the daemon immediately.	`stordaemon shutdown storsrvd`	`F STORSRVD,SHUTDOWN` `F STORSRVD,SHUTDOWN [-]IMMediate`

Table 35 stordaemon command syntax for the z/OS system console (continued)

Command	stordaemon syntax	Console syntax
	`stordaemon shutdown storsrvd -immediate`	
Show the current value of an operational variable (port in this example).	`stordaemon getvar storsrvd -name port`	`F STORSRVD,'getvar [-]name port'`
Change the current value of an daemon option (takes effect immediately.	`stordaemon setvar storsrvd -name log_filter=SESSION,APIREQ`	`F STORSRVD, 'setvar [-[name log_filter=SESSION, APIREQ'` **Note** The `-name` option can be abbreviated to 3 chars and the dash can be omitted.
Store a new value of a daemon option for reload or subsequent execution. In this example, change the port to 2708.	`stordaemon setoption storsrvd -name port=2708`	`setoption` is not supported from the console in this release.
Issue a `storsrvd` extending action. In this example, show details for SYMAPI session number 4.	`stordaemon action storsrvd -cmd show -session -num 4`	`F STORSRVD,'action show -ses -num 4` **Note** In this example the `-cmd` keyword is omitted, and a closing quote is also omitted.
Show network information.	`stordaemon action storsrvd -cmd show -netinfo`	`F STORSRVD, 'action show -netinfo'`

In general, command-generated output shown on the z/OS console will suppress blank lines for the sake of brevity and to reduce messages rolling off the console screen.

Using stordaemon TSO commands

In the TSO command shell, the `stordaemon` command operates as it does on all platforms. If the Solutions Enabler load library is in the TSO STEPLIB or CMDLIB, you can issue the `stordaemon` command as shown in the following example:

```
IKJ56455I USER1 LOGON IN PROGRESS AT 13:33:01 ON APRIL 1, 2015,
IKJ56951I NO BROADCAST MESSAGES,
REXX/SOCKETS z/OS V1R6 January 5, 2007,
READY
STORDEMN show storsrvd
<output will show here>
```

```
CALL 'EMC.SSEM830.LOADLIB(STORDEMN)' 'show storsrvd'
<output will show here>
```

Optionally, you can trap all output of the `stordaemon` command with the REXX language function outtrap(). In which case, all output will be saved in a REXX variable array, where it can be processed programmatically.

Using stordaemon in a Unix System Services shell

The following example illustrates how you can configure `stordaemon` to run from Unix System Services. For the sake of this example, assume that you have already logged in to the z/OS Unix System Services shell either via rlogin or the TSO `OMVS` command:

```
$ cd /var/symapi
$ mkdir bin
$ cd bin
$ ln -e STORDEMN stordaemon
$ export STEPLIB=EMC.SSEM830.LOADLIB
$ stordaemon show storsrvd
$ stordaemon shutdown storsrvd
```

In the example, the user makes an external link from a Unix System Services file to the Solutions Enabler load library module. By setting the STEPLIB environment variable, the shell follows the link from the Unix System Services file to the load library, finding the member stored there. The load library member executes the `stordaemon` application. Any z/OS supported `stordaemon` functions can be used in this environment.

Running the base daemon on z/OS

The base daemon (`storapid`) is required for z/OS SYMAPI server services and should be running at all times. The base daemon provides numerous benefits for the z/OS environment, including improved performance and enhanced array lock management.

Most of the information in this section is similar to the daemon information described in Post-Installation configuration for UNIX, Windows, OpenVMS, and z/OS on page 121; however, this section describes it from the z/OS point of view.

Starting the base daemon

Once the SYMAPI server is running, start the base daemon by submitting the job `#STORAPI` in the RIMLIB. This job will have been correctly configured when the SEMJCL process was run. If necessary, you can modify this job and convert it to run as a started task. You cannot use the `stordaemon` command to start the base daemon.

Note

As there is no watchdog daemon in z/OS, the base daemon will not automatically start/restart.

Stopping the base daemon

Table 36 on page 205 lists the commands for stopping the base daemon.

Table 36 Commands for stopping the base daemon

From	Use the command
Console	`F STORAPID,SHUTDOWN`
TSO	`stordemn shutdown storapid`
Unix System Services shell	`stordaemon shutdown storapid`

For more information on using these methods, refer to Controlling the server on page 201.

Using and configuring the base daemon

The base daemon behavior is determined by parameters set in the configuration file `daemon_options`. This file is found in the *`symapi_installation_directory`*`/config` folder. It is a standard text file that you can edit by way of oedit or any other text editor. For detailed information on editing the parameters in this file, refer to Controlling daemon behavior on page 144.

Base daemon logging

Solutions Enabler daemons all use a common infrastructure mechanism for logging messages and events. For information on the options available to manage the way the base daemon uses its log files, refer to Controlling daemon logging on page 144.

Avoidance and selection files and the base daemon

The base daemon will not recognize or use JCL specified selection and avoidance files. It will only use the appropriate files in the *`symapi_installation_directory`*`/config` folder in Unix System Services.

You should not use both MVS datasets (for the server) and Unix System Services files (base daemon) for these selection and avoidance files. Doing so will likely result in inconsistent definitions and confusion. If you use the base daemon, you should place the avoidance and selection files for both the SYMAPI server and the base daemon in the relevant Unix System Services location. For the SYMAPI server, the relevant DDnames in the job should be removed or commented out, so that the server will refer to the correct files in Unix System Services.

For more information on the avoidance and selection files, refer to Avoidance and selection files on page 190.

Running the event daemon on z/OS

The use of the event daemon (`storevtd`) is optional for the z/OS SYMAPI server. For information regarding the event daemon, refer to Setting up the event daemon for monitoring on page 148.

In the z/OS context, the event daemon is primarily used to enable monitoring capabilities on behalf of other clients. The only client expected to use the event daemon is EMC Unisphere for VMAX.

Starting the event daemon

Once the SYMAPI server is running, start the event daemon by submitting the job `#STOREVT` in the RIMLIB. This job will have been correctly configured when you ran the SEMJCL process. If necessary, you can modify this job and convert it to run as a started task. You cannot use the `stordaemon` command to start the event daemon.

Note

As there is no watchdog daemon in z/OS, the event daemon will not automatically start/restart.

Stopping the event daemon

Table 37 on page 206 lists the commands for stopping the event daemon.

Table 37 Commands for stopping the event daemon

From	Use the command
Console	`F STOREVTD,SHUTDOWN`
TSO	`stordaemon shutdown storevntd`
Unix System Services shell	`stordaemon shutdown storevntd`

For more information on using these methods, refer to Controlling the server on page 201.

Using and configuring the event daemon

The event daemon behavior is determined by parameters set in the configuration file **`daemon_options`**. This file is found in the *`symapi_installation_directory/config folder`*. It is a standard text file that you can edit by way of oedit or any other text editor. For detailed information on editing the parameters in this file, refer to Controlling daemon behavior on page 144.

Event daemon logging

Solutions Enabler daemons use a common infrastructure mechanism for logging messages and events. For information on the options available to manage the way the event daemon uses its log files, refer to Controlling daemon logging on page 144.

The z/OS Event Daemon supports two logging targets, namely `syslog` and `system`.

syslog
The `syslog` target routes event messages to a UNIX style syslog daemon (`syslogd`).

Note

This is a syslog daemon supporting the protocols as defined by RFC 5424 - The Syslog Protocol.

The following are examples of messages logged from an Event daemon on a z/OS host to a Linux on System z syslog daemon:

```
Feb 13 10:58:04 sys1 EMCstorevntd: [fmt=evt] [evtid=1234] [date=2011-10-13T14:58:04Z]
[symid=000000000001] [sev=info]  = Snap session created, activated or deleted.
Feb 13 10:58:07 sys1 EMCstorevntd: [fmt=evt] [evtid=1201] [date=2011-10-13T14:58:07Z]
[symid=000000000001] [sev=normal]  = Array state has changed to Online.
Feb 13 11:01:07 sys1 EMCstorevntd: [fmt=evt] [evtid=1234] [date=2011-10-13T15:01:07Z]
[symid=000000000001] [sev=info]  = Snap session created, activated or deleted.
```

The message text is prefixed with the originating host name `sys1` as well as the string `"EMCstorevntd:"`.

system

The `system` target sends event messages to the z/OS system hardcopy log.

These event messages are routed to the hardcopy log only and not to operator consoles (i.e., they are suppressed). They can be routed to the hardcopy log only on the same z/OS system on which the Event Daemon is running.

The following messages are also seen in the Event Daemon job log. Messages written to the z/OS system log are generally in the format:

```
SYS1    11291 11:41:03.72 JOB06676 00000290  SEEVT00001201  <14>  <fmt=evt>
<evtid=1201> ...
```

Where the message ID has the prefix `SEEVT` followed by an eight-digit event ID suffix. These event IDs suffixes correspond to documented Event Daemon event IDs and they are the same number as seen in the `evtid=nnnn` keyword in the message text. However, they are prefixed with sufficient zeros so as to make the SEEVT message ID suitable for automation handling via MPF or a similar tool. The numeric portion of the SEEVT message id will always be eight digits long.

Note

Event message formats on page 162 describes the formats of event messages in detail.

CHAPTER 6

Remote Operations

This chapter provides information on configuring and operating Solutions Enabler in a client/server environment:

SYMCLI through a remote server

In the UNIX, Linux, and Windows environments, the SYMAPI server runs in a background process started by the `stordaemon start storsrvd` command. In the z/OS environment, it runs as a job step task specified on the EXEC PGM= statement in a job stream. The server reads its configuration from the `daemon_options` file, and records log information in its own log file set, which resides in the SYMAPI logging directory.

The server is a multi-threaded program that listens for SYMAPI sessions and management requests initiated by the `stordaemon` command. The server also listens for management requests from the system operator console.

While session threads come and go, the server continues to accept connection requests until an operator enters a command to initiate the server shutdown process. The operator has the choice to end the server safely, where the server will wait for all current sessions to terminate on their own, or to end the server immediately, in which case the server will simply terminate all current session threads without giving them a chance to end on their own. The former method is preferred, when there is time to let sessions continue until they are done. The latter method can be used in an emergency, especially when a catastrophic condition occurs that requires a restart of the entire system.

Each session has a sequentially assigned session number, and an associated thread number. The operator can use the session number when referring to a session in a command. For example:

```
stordaemon action storsrvd -cmd show -sessions -num session_number
```

You can use the thread name (`SESS nnnn`, where *nnnn* is the session number) to identify log message issued by session threads.

Client configuration

This section explains how to configure a Solutions Enabler client.

Editing the netcnfg file

The `netcnfg` file is a template and an editable file located in the SYMAPI configuration directory.[6]

There are two ways to configure services in the `netcnfg` file:

- Single entry Service Name (legacy method): individual service name entries are specified, one for each server. Specify a hyphen (-) or the reserved word `Single` to indicate a single entry service name.
- Paired entry Service Name: two entries use the same service name, with a special indicator that controls how the SYMAPI library will choose an entry to initiate a remote session. Specify the word `Ordered` or `Balanced` to indicate a paired entry service name.

6. The location of this directory varies according to the operating system. For more information, refer to Appendix E.

Using a text editor, a System Administrator must add the network services to the file in the format of the relevant entry configuration.

Single entry Service Name

In the case of Single entry Service Names, use the following syntax:

```
service_name pairing_method network_protocol server_node_name server_network_address
port_number security_level
```

where:

service_name is the name of the service.

pairing method the hyphen (-) or the `Single` entry specifies this as a Single entry (legacy method).

network_protocol must be TCPIP.

server_node_name is the name of the server host.

server_network_address is the network address of the server. If this is specified, this value overrides the entry specified in the *server_node_name*.

Note

You can substitute a hyphen (-) for an unspecified *server_node_name* or *server_network_address*, but at least one must be specified. For more information, refer to Considerations for specifying server_node_name and server_network_address on page 213.

---

port_number is the server port number.

security_level is the type of connection the client is expecting to negotiate. Possible values are SECURE, ANY, and NONSECURE. In addition, you can specify a hyphen (-) to use the platform's default setting. For more information, refer to the *EMC VMAX Family Security Configuration Guide*.

Example
In the following example, three site-specific service names (SYMAPI_SERVER, BACKUP_SERVER and SERVER_IP6) are specified as available by the administrator:

```
SYMAPI_SERVER -  TCPIP node001 12.345.67.89                              7777 ANY
BACKUP_SERVER -  TCPIP node002 -                                          6666 SECURE
SERVER_IP6    -  TCPIP node003 3FFE:80C0:22C:18:250:88FF:FEAD:F92F        6666 SECURE
```

Comment text can be entered by placing a pound sign (#) in the first character space of the comment line.

Paired entry Service Name

There are two options of Paired entries:

- Ordered pairing means that the SYMAPI client library will first attempt a client/server session with the server named as the first of the two entries. If that attempt fails, the library will try the second one.

- Balanced pairing means that the SYMAPI client library randomly choses the first server which will be used for a client/server session. If that attempt fails, the library will try the other entry.

In the case of Paired entries, use the following syntax:

```
service_name pairing_method network_protocol server_node_name server_network_address
port_number security_level
```

where:

service_name the same service name is specified in both entries.

pairing method the `Ordered` entry specifies an ordered pairing of two entries, while the *Balanced* entry specifies a random selection method.

network_protocol must be TCPIP.

server_node_name is the name of the server host.

server_network_address is the IP address of the server host. If this is specified, this value overrides the entry specified in the *server_node_name*.

port_number is the server port number.

security_level is the type of connection the client is expecting to negotiate. Possible values are SECURE, ANY, and NONSECURE. In addition, you can specify a hyphen (-) to use the platform's default setting. For more information, refer to the *EMC VMAX Family Security Configuration Guide*.

Example
In the following example, two site-specific service names (SYMAPI_SERVER, BACKUP_SERVER) are specified, as ordered and balanced respectively, as available by the administrator:

```
SYMAPI_SERVER Ordered  TCPIP node001 -                                          7777 ANY
SYMAPI_SERVER Ordered  TCPIP node002 -                                          7777 ANY
BACKUP_SERVER Balanced  TCPIP node003 -                                          6666 SECURE
BACKUP_SERVER Balanced  TCPIP node004 -                                          6666 SECURE
```

Comment text can be entered by placing a pound sign (#) in the first character space of the comment line.

NOTES

- To configure client access to the eManagement servers, the same rules apply as presented in the sections above. In order to reach services on a guest, requests must be directed to the IP address and/or the hostname of the associated NAT Gateway.
- Both balanced and ordered pairing methods require two entries with the same name and pairing method specified in the file. It is invalid to specify one entry without a second.
- The number of balanced and ordered entries for a given service name may exceed two, but only the first two will be used. If validation of the first two succeeds, the service name will be considered valid and the first two entries will be candidates for connection attempts.

- The *server_node_name* fields in both paired entries may be different, or one or both may be a hyphen indicating that the value is omitted.
- The *IP_address* fields in both paired entries may be different or may both be a hyphen indicating that the host name must be used. Whenever there is no IP address specified, the *server_node_name* must be specified.
- DNS queries may return more than one IP address for a given host name. If a host name is mapped to two different IP addresses, the SYMAPI client library will attempt to connect to the first one. If the connection fails, the client library will try the second one. If both addresses in the first entry fail, the client library will repeat the process with all IP addresses associated with the second host.
- The *port_number* fields in both paired entries may be different.
- The *security_level* must be the same for both paired entries.

Considerations for specifying server_node_name and server_network_address

Although the syntax of each service definition allows you to specify both the node name and the network address, only one is in fact required. Specifying both can serve as documentation for your expectation of the mapping between node and address, but it has no real effect on connections established between the client and the server.

Any unspecified tokens in the service definition must be replaced with a hyphen, so if either the *server_node_name* or *server_network_address* are to be omitted, be sure to place a hyphen character in its position.

Use the following general rules to decide whether to specify a real value for *server_node_name* or *server_network_address*:

- If you do not want to have to remember or look up IP addresses, or if your network administrator discourages routing by address, then specify a real value for *server_node_name* and place a hyphen in the *server_network_address* field. The SYMAPI client library will look up the node name in DNS, and will attempt to connect to the server using the list of known addresses for the node. If you specify *server_node_name*, however, you cannot predict the address that will be used to successfully connect.
  Note that the value specified in the *server_node_name* can generally be a local node without qualifying domain, or it can be a fully-qualified domain name (FQDN). Your results depend on the configuration of name resolution in your network.

  Another key reason for using node name is that the client will try all eligible network addresses for a given node to complete the connection. Even though you have no specific control over the protocol or address used, the server availability may be improved using node name.
- If you want more control over the network address chosen (including the protocol) for the connection, specify a real value for *server_network_address* and place a hyphen in the *server_node_name* field. In fact, if any value is specified in the address field, it will be used, regardless of the value specified in the *server_node_name* field.
  Note that specifying the address implies that you know the protocols that will be in use on the server host. For example, if you specify an IPv4 address for a server which is no longer using IPv4 (not likely for years to come), the connection will fail. If you specify an IPv6 address for a server host whose IPv6 link is inoperative, the connection will fail. A host in this state might still be reachable over IPv4; by using the node name instead, the connection might succeed.

You can specify an IPv4 address or an IPv6 address. You may be able to use an IPv4-mapped address, but a successful connection using the mapped address will depend

on the whether the operating system of the server host is one that uses V4-mapping. In general, using IPv4-mapped addresses is discouraged.

Setting environment variables for remote access

To use SYMCLI through a remote SYMAPI service, you should set environment variable `SYMCLI_CONNECT` to an available service name of the server connection (defined in `netcnfg`). For example, for service name SYMAPI_SERVER, set the environment variable as follows:

`setenv SYMCLI_CONNECT SYMAPI_SERVER`	for UNIX C shell
`define SYMCLI_CONNECT SYMAPI_SERVER`	for OpenVMS
`set SYMCLI_CONNECT=SYMAPI_SERVER`	for Windows

To determine what network services are configured, enter:

```
symcfg list -service
```

Connection variable `SYMCLI_CONNECT_TYPE` should define the local/remote mode of the local host (client). Possible values for the client are:

`REMOTE`

Defines a client operation in which all the remote SYMCLI commands are strictly executed on the server, and the VMAX array database is strictly read and updated remotely.

`LOCAL`

Defines a local connection to the VMAX array. (Not used for a client-server connection.)

Example
To set the connection environment variables for a locally-cached remote operation, enter:

```
setenv SYMCLI_CONNECT_TYPE  REMOTE
```

Client/server IP interoperability

In a UNIX, Linux, or Windows environment, the SYMAPI client and server are both capable of negotiating sessions over the traditional Internet Protocol Version 4 (IPv4) and the newer Internet Protocol Version 6 (IPv6).

The IPv6 designers expected migration from the old protocol to the new protocol to take years. They designed the new protocol for interoperation in networks where both are present. A network administrator can introduce the IPv6 protocol as a supplement to IPv4, where IPv4 hosts and IPv6-capable hosts can interoperate with minimal disruption. Over time, as network configuration is improved and problems are reduced and eliminated, IPv4 protocols can be dropped in favor of IPv6. Such a transition

scheme is essential in environments where continual operation is a key business success factor.

In the UNIX, Linux, and Microsoft Windows Server environments, Solutions Enabler also supports the transition from IPv4 to IPv6 in a seamless fashion. With proper configuration of host operating systems, routers, and DNS servers, Solutions Enabler supports concurrent connections from clients using both IPv4 and IPv6. The client and server software will choose either IPv4 or IPv6 to communicate, depending on specification in configuration files of the host operating system and Solutions Enabler.

IPv6 addresses

The IPv4 address is familiar to most computer users: a 32-bit unsigned integer is displayed in a dotted-decimal string. For example, 172.23.191.20 (0xAC17BF14).

The IPv6 address supports many addressing features, but the most obvious attribute is its much wider addressing space: a 128-bit code is displayed as a series 16-bit groupings (represented in hexadecimal) separated by colons. Shorthand notation rules improve the usability of the IPv6 display address; nonetheless, an IPv6 address is not a human-friendly object. For example, one machine might be represented with this address:

`3ffe:80c0:22c:18:250:8bff:fead:f92f`

(0x3FFE80C0022C001802508BFFFEADF92F)

IPv4 address mapping

The interoperation of IPv4 and IPv6 varies from one operating system to another, according to the specification of IPv6. On some host operating systems, IPv4 connections are made through the native IPv4 protocol, and IPv4 addresses are represented as the dotted-decimal addresses which are familiar.

Other OS vendors have chosen to complete client connections from an IPv4 machine over IPv6, where the IPv4 address is represented as an IPv4-mapped address. An IPv4-mapped address appears in colonated-hexadecimal form, where the last 32-bits of the address are shown as the dotted-decimal IPv4 address (they may also be shown as two pairs of hexadecimal bytes). Immediately preceding the IPv4 address is the string ::FFFF:. For example, a host whose IPv4 address is 172.23.191.20 can be represented as a IPv4-mapped address as follows:

::FFFF:AC17:BF14

or

::FFFF:172.23.191.20

(0x00000000000000000000FFFFAC17BF14)

IPv4-mappped addresses are used by operating systems that do not support concurrent binding to the same port over both IPv6 and IPv4. AIX, and Linux generally use IPv4-mapped addresses.

SunOS, HP-UX, and Microsoft Windows 2003 allow concurrent binding on both IPv6 and IPv4 protocols.

Server operation

The SYMAPI server listens for arrival of client connections on either IPv6 or IPv4 protocols, or on both where possible. The server begins by attempting to bind to the *unspecified address* using the IPv6 protocol. It then attempts to bind the unspecified address using the IPv4 protocol.

The *unspecified address* is a special-purpose internet address used primarily by server applications. It indicates that an application is ready to receive a connection on any internet address configured on the host with a matching protocol. For hosts that have multiple network interfaces, it increases the availability of the server application by not limiting connections to arrive by way of a specific address.

The server insists on at least one successful bind on either IPv6 or IPv4 protocols, and will use both if available to continue initializing. If both bind attempts fail, the server will terminate immediately, since no network is accessible or the port is in use.

When the server has finished initializing for network communication, it will write the following message to its SYMAPI log file and to the terminal device, if one is available:

`ANR0020I SYMAPI server listening on port` *`port`* `over` *`protocols`*

Where *port* is the decimal port number to which client connections should be directed, and *protocols* are the protocols the server is using to listen for client connections. Possible values are:

- `IPv6 and IPv4` — Indicates that the server will accept connections from clients running either IPv6 or IPv4.
- `IPv6 with IPv4 mapping` — Also indicates that the server will accept connections from clients running either IPv6 or IPv4. Connections from IPv4 clients will be represented on the server side as an IPv4-mapped address (refer to IPv4 address mapping on page 215).
- `IPv4 only` — Indicates that IPv6 bind failed. Connections can only be accepted from IPv4 clients.

Client operation

The SYMAPI client library will attempt to connect to the server either by node name or by internet address, depending on how the service name is specified in the `netcnfg` file.

If the internet address of the server is specified, the client makes a single attempt to connect to the server. The client chooses the protocol based on the nature of the address: if it is an IPv4 address, it will specify IPv4 as the protocol. Similarly, specifying an IPv6 address (including an IPv4-mapped address) will result in the client using the IPv6 protocol to connect to the server.

If the node name of the server is specified, the client will lookup the server host by name. Such a lookup operation can return a list of candidate addresses, potentially including both IPv4 and IPv6 addresses. The client library will try to connect to all eligible addresses until either a connection attempt succeeds, or the list is exhausted with no successes. The list of eligible server addresses depends on the static and dynamic name resolution configuration of the host on which the client is running.

Client/server security

By default, the SYMAPI client and server, on platforms that will support it, are initially configured to negotiate only secure sessions. To modify this default behavior, you can configure the security level at which the client and server are operating. You can also change many other aspects of secure client/server operation. Refer to the *EMC VMAX Family Security Configuration Guide* for more information on client/server security and how to configure related settings.

Specifying server behavior

Table 38 on page 217 describes the `daemon_options` file parameters that you can use to control the behavior of the SYMAPI server daemon `storsrvd`.

For information on editing these parameters, refer to Controlling daemon behavior on page 144.

Table 38 storsrvd options for the daemon_options file

Parameter	Possible values[a]	Reloadable
port Specifies the decimal port number.	= *nnnn*| **2707**	No
log_show_category Specifies whether the specific storsrvd log category value should be displayed when a log message is written.	= ENABLE | **DISABLE** ENABLE: The category associated with the log event is shown as part of the text message. DISABLE: The category is not shown as part of the message.	Yes
log_show_msgid Specifies whether the specific storsrvd message identifier should be displayed when a log message is written.	= ENABLE | **DISABLE** ENABLE: The message ID of a storsrvd application log message is shown as part of the text message. DISABLE: The message ID is not shown as part of the message.	Yes
log_level Specifies a severity-based control over logging volume. Messages that are issued with a severity equal to or exceeding the level specified will be recorded in the log file. Do not use debug or verbose without direction from EMC Customer Support.	= ERROR | INFO | DEBUG | VERBOSE | **WARNING**	Yes
log_filter Specifies the types of events to log.	= SERVER | SESSION | APIREQ | **CONTROLS** SERVER: Log high level events related to initialization, termination, and main thread. SESSION: Log logical session events (arrival, termination,	Yes

Table 38 storsrvd options for the daemon_options file (continued)

Parameter	Possible values[a]	Reloadable
	security level, authorization rejections). APIREQ: Log SYMAPI activity (request start and stop (with completion status)). CONTROLS: Log control session handling information (command parsing, execution). **Note** Leaving this parameter commented out will result in the SYMAPI server application-level messages not being logged.	
security_alt_cert_file Specifies an alternate certificate file to the certificate file provided at installation. The specified file should have a matching `security_alt_key_file` option set for the matching key file. A full path name must not be specified. Specify the name of a file that resides in the `<SYMAPI_HOME>/config/cert` directory.	= *Any valid simple file name* | **symapisrv_cert.pem**	No
security_alt_key_file Specifies an alternate key file to the key file provided at installation. The file specified should have a matching `security_alt_cert_file` option set for the matching certificate file. A full path name must not be specified. Specify the name of a file that resides in the `<SYMAPI_HOME>/config/cert` directory.	= *Any valid simple file name* | **symapisrv_key.pem**	No
security_clt_secure_lvl Controls the verification of the client certificate by the	= NOVERIFY | MUSTVERIFY | **VERIFY**	Yes

Table 38 storsrvd options for the daemon_options file (continued)

Parameter	Possible values[a]	Reloadable
server. This parameter is not supported in z/OS. This value is ignored if secure communications are not established.	NOVERIFY: Indicates that the server will not verify the client certificate. MUSTVERIFY: Indicates that the server will only accept communications from a version of the client that can send a certificate to be verified. VERIFY: Indicates that the server will verify a client certificate if the version of the client can send a certificate.	

a. Default values are bold.

Controlling the server

This section explains the commands used to control the SYMAPI server.

Starting the server

If you have not already configured your host to start the server automatically, then you must start the SYMAPI service using the following command executed from the server side:

```
stordaemon start storsrvd
```

Stopping the server

To stop the SYMAPI service from the server side, use the following command:

```
stordaemon shutdown storsrvd
```

Showing server details

The `stordaemon show storsrvd` command displays the following information regarding the SYMAPI server:

- SYMAPI version
- Total number of sessions since startup
- Current active sessions
- log_show_msgid setting
- log_show_category setting
- Enhanced authentication setting

In the z/OS environment:

- In the z/OS environment:

  cond_hdlr (condition handler)

- Version of the language environment library

The `stordaemon action storsrvd -cmd show server` command displays the same information as the `stordaemon show storsrvd` command with the addition of operating system information.

The following example shows the output of a `stordaemon show storsrvd` command:

```
stordaemon show  storsrvd


Daemon State                                    : Running
Daemon Start Time                               : Wed Apr 10 08:18:35 2015
Version                                         : V8.3-1900 (0.0)
Auto-Restart by Watchdog                        : Disabled

Total Number of Connections                     : 2
Number of Active Connections                    : 0
Total Number of Requests                        : 0

ANR0123I Show Server Details                    :

SYMAPI Version                                  : V8.3.0.0   (Edit Level: 1900)
SYMAPI Session Total/Active                     : 0/0
SYMAPI Session Port                             : 2707
Security Level                                  : ANY
Show ANR Category                               : Disabled
Show ANR Message Id                             : Enabled
Enhanced Authentication                         : Disabled
Client Verification Level                       : VERIFY
Transfer Protocol Version                       : 2
Maximum Sessions                                : 100
Maximum Sessions per Host                       : NOLIMIT
Maximum Sessions per User                       : NOLIMIT
Symapi Debug Permitted                          : SERVER
Allow Wildcarded Certificates                   : Enabled
```

In the above example:

- The first seven lines of the display are generated by common logic. All daemons display lines similar to these, with information that reflects the state of the daemon.
- The lines following the message ANR0123I are generated by storsrvd, and will not display for any other daemon.
- `Total Number of Connections` is the total connections handled during the life of the daemon process. For most daemons, this includes control sessions (those that execute commands to control the daemon) and application sessions (those that need application services provided by the daemon). This number does not include the dedicated session managed by the z/OS Console thread.
- `Number of Active Connections` is the number of currently executing control sessions and application sessions.
- `Total number of Requests` is the number of control commands and application requests (SYMAPI function calls received at the server).
- `SYMAPI Session Total/Active` is the number of SYMAPI sessions only; it does not include the number of control sessions.

The following example shows the output of a `stordaemon action storsrvd -cmd show server` command:

```
stordaemon action storsrvd -cmd show server

ANR0123I Show Server Details:

SYMAPI Version                : V8.3.0.0   (Edit Level: 1900)
SYMAPI Session Total/Active   : 0/0
SYMAPI Session Port           : 2707
Security Level                : ANY
Show ANR Category             : Disabled
Show ANR Message Id           : Enabled
Enhanced Authentication       : Disabled
Client Verification Level     : VERIFY
Transfer Protocol Version     : 2
Maximum Sessions              : 100
Maximum Sessions per Host     : NOLIMIT
Maximum Sessions per User     : NOLIMIT
Symapi Debug Permitted        : SERVER
Allow Wildcarded Certificates : Enabled

ANR0123I Show OS Information Details:

Process ID                    : 20576
Host OS Name/Version          : Linux/2.6.18-194.el5
Processor Model/CPUs          : x86_64/2

ANR0123I Show Symapi Debugging Details:

SYMAPI_DEBUG                  : 0x00000000
SYMAPI_DEBUG2                 : 0x00000000
SYMAPI_DEBUG_CONTROLS         : 0x00040100
SYMAPI_DEBUG_FILENAME         : /var/symapi/log/debug/storsrvd_debug.log
```

Displaying networking information

The `show -netinfo` command displays information about the storsrvd networking interfaces. For example:

```
stordaemon action storsrvd -cmd show -netinfo


 ANR0123I Show Network Details:
 SYMAPI Session Port           : 2707
 IP Protocols                  : IPv6 with IPv4 mapping
 Host Name                     : Host1051
 IP address                    : 172.23.193.51
```

The above example includes information on the following:

- The port on which the server is listening.
- The IP protocols accepted by the server.
- The node name without the domain.
- The IP address line will be repeated for as many IP addresses as are known by the resolver configuration (local host files or DNS) on the host. Multi-homed hosts may show multiple lines, and hosts known by both IPv4 and IPv6 addresses may show multiple lines.

Reloading the daemon_options file

The `reload` command re-reads the `daemon_options` file, and adjusts its behavior according to the specified options. For example:

```
stordaemon action storsrvd -cmd reload
```

Summarize active SYMAPI sessions

The `list -sessions` command shows a one line summary of each currently active SYMAPI session thread. The list includes the session number (ordered by connection arrival), the thread number processing the session, the client host userid, and the host name or IP address where the session originated. For example:

```
stordaemon action storsrvd -cmd list -sessions
```

Show session details

The `show -session` command displays details about active sessions. This command uses the following form:

```
stordaemon action storsrvd -cmd show -session [-num session_num] [-
hostinfo]
```

Where:

`-num` *`session_number`* shows details on a particular session. If this option is not specified, the command will show details for all active sessions. If this option is used and the session number does not exist, an error message will display. You can view a list of session numbers using the `list -sessions` command.

`-hostinfo` shows details about the client host.

The following example shows the output of a `show -session` command:

```
stordaemon action storsrvd -cmd show -session -hostinfo

storsrvd
ANR0124I ==== Show Session Details for Session 1 on Thread 2:
User/Host:       Joe/Host127.aaa.bbb.com
Authentication
SYMAPI Version:  8.3.0
Session Started: 2015/04/07 17:25:53   Seclevel: NONSECURE
Total Requests:  2
Last Request:    SymUserContextSet  (4190)
     Started:    2016/03/24 13:07:32
     Ended:      2016/03/24 13:07:32   Result:      0 (SYMAPI_C_SUCCESS)
Client host information:
    PID:         11992
    OS:          SunOS
    Addressing:  64-bit
    Charset:     ASCII
    Byte Order:  Big Endian
```

The previous example includes information on the following:

- Remote client user name and host name (if it can be resolved, IP address if it cannot be resolved)
- API library version in use by the client, and architecture (64-bit)
- Session start time and security level
- Start time of the last API request, and the numeric code of the API
- End time of the last API request and the completion code, as well as the SYMAPI return code name (as defined in efbcore.h)
- Process ID of the client

Controlling and using the storsrvd log files

The server writes data to its log files provided by the common daemon infrastructure. These log files are named and handled in a manner consistent with other daemon log files. For example, under the default log management behavior, the files `storsrvd.log0` and `storsrvd.log1` are created in `/var/symapi/log`.

The behavior of the log files is subject to the standard daemon options: `logfile_type`, `logfile_size`, `logfile_perms` and `logfile_retention`. Thus, you can configure the logs as dated files with retention controls instead of the common wrapping pair of log0 and log1. The same rules apply to `storsrvd` as to all other daemons.

You can control the volume of data written to the log files with the `daemon_options` file parameters `log_filter` and `log_level`. For a description of these options, refer to Specifying server behavior on page 217.

Numbered messages issued by storsrvd

The SYMAPI server application-level messages are distinguished from messages issued by the Solutions Enabler common daemon support by the use of a messages identifier. The complete set of `storsrvd` messages is documented in the *VMAX Management Software Events and Alerts Guide*.

The following `daemon_options` file keywords affect the appearance of the `storsrvd` messages:

- `log_show_category` displays or suppresses the category (also known as the filter) that applies to a message.
- `log_show_msgid` displays or suppresses the message identifier in the message.

For a description of these options, refer to Specifying server behavior on page 217.

CHAPTER 7

Technical Notes and Configuration

This chapter provides technical notes for advanced configuration of Solutions Enabler, VSS Provider, and SMI-S Provider.

Solutions Enabler technical notes

Changes to default port flag settings

The default port flag settings have been updated in Enginuity 5875 and higher. The new default values should simplify the array installation and minimize changes required at install time. Table 39 on page 226 lists host environments and identifies the environments that require changes to the default port flag settings. Port flag settings can be updated using the `symconfigure` command.

Table 39 Port settings by operating environment

Operating environment	Port settings
EMC Celerra	Enable ARB and D flags
IBM AIX	5875 defaults unchanged
IBM i	Enable AS4 flag
Linux	Enable D flag
Hewlett-Packard OpenVMS	Enable OVMS flag
Hewlett-Packard HP-UX 11i V1 and V2	Enable V flag, disable SC3 and OS07 flags
Hewlett Packard HP-UX 11i V3	Enable V flag, disable SC3 flag
Microsoft Windows Server 2003	5875 defaults unchanged
Microsoft Windows Server 2008	5875 defaults unchanged
Oracle Solaris	5875 defaults unchanged
VMware ESX	5875 defaults unchanged

In addition, the following special configurations require changes to the new default port flag settings:

- Environments that directly connect servers to the array—not using switches or other SAN components—using Fibre-Channel Arbitrated Loop (FC-AL) require the PP flag to be disabled and a Loop ID assigned to a particular number ranging from 0-126.
- Environments that contain Fujitsu (Formerly Siemens), Novell Netware, or Teradata systems, should refer to the EMC support matrix at www.emc.com for specific recommendations.
- The VMAX array is configured by default with the ACLX flag enabled. This requires the masking of specific devices to specific ports on the array. You must provision ACLX volumes to enable host management using Solutions Enabler.

Existing scripts should be tested to ensure compatibility with the new default port flag settings.

Note

Some of these new default values are different from earlier Enginuity settings. Connecting a host to arrays running Enginuity 5875 and arrays running an early Enginuity version may require changes to the 5875 defaults. Alternatively, you can set the flags at the initiator to match the previously installed array with its existing connection to the server. Changes to flag settings become effective after rebooting the host.

Parallel Access Volumes

From DMX-3, DMX-3 950, DMX-4, DMX-4 950 and higher, both Dynamic Parallel Access Volumes and Hyper Parallel Access Volumes use a pool of aliases for the Control Unit image. Aliases are not dedicated to specific devices and they do not appear in the device specific Aliases column of the `symcfg show -cuimage` command.

SIU support for ESX Server V4.0

The Symmetrix Integration Utilities (SIU) supports VMware ESX Server V4.1 Update 1.

Access Control setup

Use of the Symmetrix Access Control feature requires the help of EMC Customer Service to initially set up your array. For more information, contact your EMC Representative and refer to *EMC Solutions Enabler Array Management CLI User Guide* and *EMC VMAX Family Security Configuration Guide.*

Note

The Solutions Enabler (`symacl`) command provides full support for open systems. Access Control can be enforced for SYMAPI-based applications on z/OS platforms. However, z/OS and IBM i cannot be used as the Access Control administration node.

Note

CREATEDV access can only be granted to ALL devices or on !INPOOLS when there are no accpools defined.

When host access IDs are tied to the host hardware, (refer to Alternate access ID earlier) upgrading or changing hardware components might result in a change to the host access ID. For information about changing a host's Alternate access ID, see the *EMC VMAX Family Security Configuration Guide*.

Solutions Enabler access control requirements

If Solutions Enabler Access Control is enabled on the VMAX array, then the host on which the SMI-S Provider is running must have sufficient privileges to perform the necessary operations. At a minimum, the host on which the SMI-S Provider is running must be in a group that has access to ALL_DEVS with BASE and VLOGIX privileges.

Solutions Enabler Windows 2008 configuration requirements

Solutions Enabler supports Windows 2008 and requires some additional configuration due to changes in Windows permissions.

The Solutions Enabler SYMCLI binaries require that the user executing them have write access to the following folders:

- `C:\Program Files\EMC\SYMAPI\db`
- `C:\Program Files\EMC\SYMAPI\log`
- `C:\Program Files\EMC\SYMAPI\ldb`

These folders are created during the Solutions Enabler installation and inherit permissions from `C:\Program Files`. On most platforms, the resulting ACLs on these folders will grant write privileges to the Administrators Group.

Using Windows Server versions prior to Server 2008, any member of the Administrator's Group can execute the SYMCLI binaries. Using Windows Server 2008, and its User Access Control (UAC), only the built-in Administrator is by default granted Administrative privileges.

Other members of the Administrator's Group will run in a degraded mode—as an ordinary User—and therefore cannot execute the SYMCLI binaries. As a result, there are several options for running Solutions Enabler binaries on Windows Server 2008:

- Log in as the built-in Administrator. This account may have been disabled by a Systems Administrator when Windows Server 2008 was installed.
- Log in using a different account and temporarily elevate your privileges to run as a full Administrator. For example, right-mouse-click on the `CMD.EXE` command icon and select **Run as administrator** from the menu. This will open a command shell running with full administrative privileges.
- Change the protection on the folders listed above to grant write access to the users you want executing Solutions Enabler binaries.

CQL support statements

The SMI-S Provider supports CIM ExecQuery operations using CQL statements. The performance and scalability of these CQL operations can vary widely depending upon the type and scope of the query being used along with the number of objects that must be evaluated in order to return a result. If you intend to use CQL queries as part of your application, please contact EMC through normal customer support channels with a list of your CQL queries so that they can be evaluated.

In compliance with the CQL Specification, the syntax for the “not equal” operator is “<>”. If your client application uses “!=”, it must be modified to be in compliance with the CQL specification. For example, below is a compliant CQL query using the “not equal” operator:

```
select * from CIM_StorageVolume where CIM_StorageVolume.Usage <> 2
```

AIX Object Data Model Environment Variable

If the ODMDIR environment variable in the running shell is not configured properly, a `The host System Stable Values do not match the current system configuration` error message is displayed when the user tries to run Solutions Enabler commands from a script using third party applications when logged in with non-root user ID.

The ODMDIR environment variable in the running shell should point to the directory where the AIX Object Data Model DB is located.

Set the environment variable `ODMDIR=/etc/objrepos` from the UserId used to run the script, or export the Environment Variable from the 3rd party script/application by:

```
export ODMDIR=/etc/objrepos
```

VSS Provider technical notes

Enable debugging for VSS Provider

To enable debug logging for VSS Provider on a given host, perform the following steps:

Procedure

1. Select **Run** from the Windows **Start** menu, type `regedit` in the Open selection window, and click **OK**. This opens the Registry Editor.
2. Select the following registry key from those listed:

   ```
   HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy
   ```

   Note

   The EMCVssProvider service must have been previously started and a snapshot attempted for a key to exist in the list.

3. Change the *LogLevel* value from `Error` to `Debug`.
4. Close the `regedit.exe` program.
5. Stop and restart the **EMCVssProvider** and **VolumeShadowCopyService** services.

Log file

By default, VSS Provider writes all errors and notable information messages to a log file (`hwprov.log`) located in the Solutions Enabler log folder (`C:\Program Files\EMC\SYMAPI\log`). This file provides necessary information for troubleshooting operations of VSS Provider.

Note

To change the location of the VSS Provider log file, edit the Log file registry key located in the `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy` directory.

Registry keys

Table 40 on page 230 lists the VSS Provider registry key fields and the possible values.

Note

When VSS Provider is installed, only the `EnforceDefaultToClone` registry key is set to False by default. Users must set the correct registry keys based on the information provided below to use a particular snapshot technology (Mirror/Clone/VSnap/Snap/SnapVX). Registry keys for two or more snapshot technologies must not be mixed.

Table 40 VSS Provider registry key values

Name	Type	Value/location
RemoteSnapshotsOnly	`REG_SZ`	Possible values include: TRUE = Enables creation of remote snapshots only. FALSE = EMC VSS Provider defaults to local snapshots if both are available. Default value = FALSE Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`
EnforceStrictBCVPolicy	`REG_SZ`	Possible values include: TRUE = Indicates that EMC VSS Provider enforces a strict BCV rotation policy, where a BCV should only be used if it is not currently part of a snapshot. FALSE = Indicates that EMC VSS Provider does not enforce a BCV rotation policy, leaving enforcement to the VSS requestor. Default value = FALSE Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`

Table 40 VSS Provider registry key values (continued)

Name	Type	Value/location
EnforceMappedDevPolicy	REG_SZ	Possible values include: TRUE = Indicates that EMC VSS Provider selects a target device if it is mapped to any front-end director. FALSE = Indicates that EMC VSS Provider does not need to look for a mapped/unmapped device. Default value = FALSE Location = HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy
SymmetrixStaticMount	REG_SZ	Possible values include: TRUE = The provider does not remove the target device from the host while taking the snapshot. When deleting a snapshot, the target device is not removed from the host. FALSE = When creating or deleting a snapshot, the target device is removed from the host, that is, LUN masking is performed. Default value = FALSE Location = HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy
EnforceDefaultToClone	REG_SZ	Possible values include:

Table 40 VSS Provider registry key values (continued)

Name	Type	Value/location
		TRUE = The provider uses TimeFinder Clone as default plex snapshot. FALSE = The provider does not use TimeFinder Clone as default plex snapshot. Default value = FALSE Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`
RetainCloneSession	`REG_SZ`	Possible values include: TRUE = Indicates that EMC VSS Provider should enforce a clone retention policy, where a clone session is retained after snapshot deletion for later incremental backups. FALSE = Indicates that EMC VSS Provider does not enforce the clone retention policy, leaving enforcement to the VSS requestor. Default value = FALSE Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`
EnforceVPSnap	`REG_SZ`	Possible values include: TRUE= The provider will look for VP snap replicas as default differential snapshot.

Table 40 VSS Provider registry key values (continued)

Name	Type	Value/location
		FALSE= The provider will look for Snap replicas for differential snapshot. Default value = FALSE Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`
RetainVPSnapSession	`REG_SZ`	Possible values include: TRUE = Indicates that VSS Provider should enforce a VP Snap retention policy, where a VP Snap session is retained after snapshot deletion for later incremental backups. FALSE = Indicates that VSS Provider does not enforce the VP Snap retention policy, leaving enforcement to the VSS requestor. Default value = FALSE Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`
EnforceTimeFinderVX	`REG_SZ`	Possible values include: TRUE = indicates that VSS Provider will look for SnapVX replicas (for plex or differential snapshots). FALSE = indicates that VSS Provider will

Table 40 VSS Provider registry key values (continued)

Name	Type	Value/location
		not look for SnapVX replicas. Default value = FALSE Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`
SelectVXTarget	`REG_SZ`	Possible values include: TBCV = indicates that VSS Provider will select Thin BCV device as SnapVX snapshot target if a valid device is available in the device group. TDEV = indicates that VSS Provider will select Thin data device as SnapVX snapshot target if a valid device is available in the device group. ANY = indicates that VSS Provider will select Thin BCV device first (followed by Thin data device if required) as SnapVX snapshot target if a valid device is available in the device group. Default value = ANY Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`
RetainVXTarget	`REG_SZ`	Possible values include: TRUE = indicates that VSS Provider should enforce SnapVX

Table 40 VSS Provider registry key values (continued)

Name	Type	Value/location
		retention policy, where same VX snapshot target is retained for incremental backups later. FALSE = indicates that VSS Provider does not enforce SnapVX retention policy. Default value = FALSE Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`
VXTimeToLive	`REG_SZ`	Possible values are between 1 to 400 days (both 1 and 400 included). VXTimeToLive indicates that a SnapVX snapshot is retained for these many number of days when RetainVXTarget is set to TRUE. If RetainVXTarget is set to FALSE, VXTimeToLive is ignored. Default value=1 Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`
SymmetrixSnapPoolName	`REG_SZ`	SymmetrixSnapPoolName indicates the name of snap pool to be used for TimeFinder Snap (on Enginuity 5876). This name can be a maximum of 32 characters. If this key is not set, TimeFinder

Table 40 VSS Provider registry key values (continued)

Name	Type	Value/location
		Snap uses the default snap pool name. Location = `HKEY_LOCAL_MACHINE\Software\EMC\ShadowCopy`

Note

If changes are made to any of the registry key values listed in Table 40 on page 230, the `EMCVssProvider` service must be stopped and restarted for the changes to take effect.

Remote snapshots

VSS Provider supports both local and remote (SRDF) snapshots on VMAX arrays. If both local and remote target devices are available, VSS Provider defaults to local snapshots. To force VSS Provider to create a remote snapshot, set the `RemoteSnapshotsOnly` registry key as shown in Table 40 on page 230.

Enforcing a strict BCV rotation policy

As noted in Table 40 on page 230, if the `EnforceStrictBCVPolicy` is enabled, the policy has the following effects on the snapshot process:

- To support a snapshot for a given BCV, the BCV must be in one of the following states: Synchronized, SyncInProgress, or Not Ready.
- Once the snapshot is created (BCV has been split), the BCV returns to a Ready state.
- After the snapshot is deleted, the BCV returns to a Not Ready state.

Note

When Replication Manager is installed, it creates the `EnforceStrictBCVPolicy` parameter settings in the Registry. If Replication Manager is uninstalled, ensure that the parameter setting is removed, as it may interfere with the performance of other applications (such as the TimeFinder/Integration Module).

Enforcing a mapped device policy

As noted in Table 40 on page 230, if the `EnforceMappedDevPolicy` is enabled, the policy has the following effects:

- To support a snapshot for a TimeFinder Mirror, the provider chooses a mapped target BCV that is in one of the following states: Synchronized, SyncInProgress, or Split.
- To support a snapshot for a TimeFinder Snap, the provider chooses a mapped target VDEV that is in the Created state, or any mapped VDEV that is part of the same device group.

Note

In case of single source paired with multiple target devices, VSS Provider selects the first mapped target device if available.

Using SymmetrixStaticMount to disable LUN masking and unmasking

As noted in Table 40 on page 230, if `SymmetrixStaticMount` is enabled, this has following effects during snapshot creation and deletion:

- Provider will not remove target device from the host while creating the snapshot.
- During import of the snapshot, Provider will not attempt to add target device to host.
- Provider will not remove target device from the host while deleting the snapshot.

Note

To take snapshots with registry key `SymmetrixStaticMount` enabled, it is required that target devices are made visible to the VM or host before the snapshot creation. User should see target devices under Disk Management on Windows Server operating system.

Enforcing TimeFinder Clone as default plex snapshot technology

Installation of VSS Provider creates registry key `EnforceDefaultToClone` with a default value of **FALSE**.

When the registry key `EnforceDefaultToClone` is set to TRUE, the VSS Provider uses TimeFinder Clone as the default plex snapshot. In this case, snapshot creation with TimeFinder Mirror sessions is not supported.

When the registry key `EnforceDefaultToClone` is set to FALSE (Default), the VSS Provider does not use TimeFinder Clone as the default plex snapshot. TimeFinder Clone operations requires the use of EMC requestors.

Enforcing a clone retention policy

The clone retention policy is applicable to TimeFinder Clone operations. As noted in Table 40 on page 230, if `RetainCloneSession` is enabled, then the policy has the following effects on the snapshot process:

- To support a snapshot, the target device must be in one of the following states: Created, Recreated, or Not Ready.
- Once the snapshot is created, the target device returns to a Ready state.
- After the snapshot is deleted, the target returns to a Not Ready state.

Enforcing TimeFinder VP Snap as default differential snapshot technology

To create differential snapshots for VP Snap Sessions, use the `EnforceVPSnap` flag.

When the registry key `EnforceVPSnap` is set to TRUE, the VSS Provider uses TimeFinder VP Snap as the default differential snapshot.

When the registry key `EnforceVPSnap` is set to FALSE (default), and the `EnforceTimeFinderVX` is set to FALSE (default), the VSS Provider uses TimeFinder Snap as the default differential snapshot technology.

Enforcing a VP Snap retention policy

The VP Snap retention policy is applicable to TimeFinder VP Snap operations. As noted in Table 40 on page 230, if `RetainVPSnapSession` is enabled, then the policy has the following effects on the snapshot process:

- To support a snapshot, the target device must be in one of the following states: Created, Recreated, or Not Ready.
- Once the snapshot is created, the target device returns to a Ready state.
- After the snapshot is deleted, the target returns to a Not Ready state.

Enforcing SnapVX as default snapshot technology on HYPERMAX OS 5977

When the registry key `EnforceTimeFinderVX` is set to TRUE, VSS Provider V8.3 uses SnapVX as the default snapshot technology. Snapshot context - plex or differential - is specified by VSS requester during backup operation. If no context is specified by requestor for SnapVX, VSS Provider uses differential context as default.

When the registry key `EnforceTimeFinderVX` is set to FALSE (default), the VSS Provider does not use SnapVX as the default snapshot technology.

Note

If registry key `EnforceTimeFinderVX` is set to TRUE, then user must not set `EnforceDefaultToClone` to TRUE or `EnforceVPSnap` to TRUE. This is by design to avoid mixing of these registry keys. VSS Provider will return appropriate error message if these keys are mixed.

LUN resynchronization

VSS Provider supports the LUN Resynchronization (restore) feature for transportable shadow copies that is provided with Microsoft Volume Shadow Copy Service on Windows platforms for VMAX arrays. The LUN Resynchronization feature allows a source LUN to be restored from the destination LUN, in the event that there is data loss on the source LUN. The Diskshadow VSS requestor tool should be used to initiate and perform the resynchronization.

LUN resynchronization support on VMAX arrays

The following information applies to using the LUN resynchronization feature on VMAX arrays:

- LUN resynchronization is supported for TimeFinder Mirror, Clone, VP Snap, Snap and SnapVX.
- For TimeFinder Mirror and Clone, LUN resynchronization is supported to both existing and new LUNs. For this, the new LUN must be a DATA device that is online.
- For SnapVX, VP Snap and Snap LUN resynchronization is supported to existing LUNs only.
- On successful LUN resynchronization operations, the devices are in a restored state. The requesting application, or the user, is responsible for termination of the restored session.

VSF (Veritas Storage Foundation) 5.1 SP1 for Windows

From VSS Provider V8.0, Veritas Storage Foundation (VSF) is no longer supported.

Windows Server 2008 R2 CSV (Cluster Shared Volumes)

From VSS Provider V8.0, snapshots of CSV on VMAX arrays are supported. For CSV backup, the requestor used should include Hyper-V writers.

Windows Server 2012 or 2012 R2 CSV

From VSS Provider V8.0, snapshots of CSV on VMAX arrays are supported. For CSV backup, the requestor used should include Hyper-V writers.

Using DPM to back up virtual machines deployed on CSV

When using System Center Data Protection Manager as a requestor application, virtual machines deployed on CSV with VSS Provider can be backed up serially.

Note

VSS Provider does not support parallel backups.

SMI-S Provider technical notes

Global mode

These steps must be completed before any replication group operations are initiated.

Procedure

1. Shut down ECOM service.
2. Shut down Solutions Enabler daemons.
3. In the `SYMAPI/config/options` file add/enable this setting:

   ```
   SYMAPI_USE_GNS = ENABLE
   ```

4. Start ECOM service. ECOM service will automatically start the Solutions Enabler daemons.

Mirror replication in two-provider configurations

These steps must be completed to enable mirror replication for two-provider configurations:

Procedure

1. Shut down ECOM service.
2. Shut down Solutions Enabler daemons.
3. In the `SYMAPI/config/options` file add/enable these settings:

   ```
   SYMAPI_USE_GNS = ENABLE

   SYMAPI_USE_RDFD = ENABLE
   ```

4. In the `<SYMAPI_HOME>/config/daemon_options` file add/enable this setting:

   ```
   storgnsd:GNS_REMOTE_MIRROR = enable
   ```

5. Start ECOM service. ECOM service will automatically start the Solutions Enabler daemons.

To examine the running daemon, use the `stordaemon list` command.

To stop all of the daemons, use the `stordaemon shutdown all -immediate` command.

To start a daemon, use the `stordaemon start <daemon name>` command.

Object paths in SMI-S Provider V8.3

The key values of the object path in SMI-S Provider V8.3 are different than the key values of previous SMI-S provider versions. As a result, an object path in SMI-S Provider V8.3 is incompatible with the previous versions, and vice versa.

CIM interop namespace

The CIM interop namespace for the SMI-S Provider is:

```
interop
```

Unexpected termination: Windows dump file

SMI-S Provider produces a dump file on the Microsoft Windows platform should the provider terminate unexpectedly.

If an unexpected termination occurs on the Windows platform, a dump file is produced in the `Program Files\EMC\ECIM\ECOM\Providers` directory with the name `ECOM.exe.dmp`. If this occurs, send this file along with the remaining contents of the log directory to EMC Customer Support for analysis.

Statistics collection interval

By default the Block Server Performance Subprofile collects statistics from an array every 15 minutes once the collection of statistics for that array is initiated. EMC does not support changing that interval to anything less than 15 minutes.

Logging in with the LDAP user

Use the following format when logging into the ECOM server using LDAP:

```
<domain>\<username>@<ldapserver>
```

or

```
<username>@<ldapserver>
```

For example, User1 in the ABCDomain attempting to connect to LDAPserver1 should use the following credentials:

```
ABCDomain\User1@LDAPserver1
```

or

```
User1@LDAPserver1
```

SMI-S Provider user roles

A role is a predefined set of permissions, or access types, that determine the operations a user can perform. Roles are predefined in SMI-S Provider and cannot be modified. The following list details the user roles defined in SMI-S Provider along with the associated privileges and capabilities:

- Administrator - User will have access to all administrative and storage management interfaces and configuration data.
- Manager - User will have visibility of all storage system configuration data and will be able to perform all storage management operations.
- Monitor - User will have visibility of all storage system configuration data, but will not be able to perform any storage management operations.
- Security administrator - User will only be able to perform SMI-S security tasks and will not have visibility of any storage system configuration data.
- VM User - This role is deprecated.

Linux on System z technical note

The following technical notes are specific to the Linux on System z operating environment:

HBA libraries

To run commands, such as `syminq hba`, version 1.3 (not version 1.4) of the zfcp HBA library has to be built and installed on the guest.

The IBM developerWorks "Useful add-ons" website documents the building and installation of the zfcp HBA API library.

After the zfcp HBA API library is installed, the driver must be loaded into the kernel using the following command:

```
modprobe zfcp_hbaapi
```

Various releases of Linux on System z have different names for the HBA API library. By default, Solutions Enabler V8.3 uses **`/usr/lib64/libzfcphbaapi.so.0`**

If this is not the correct library, link **`/usr/lib64/libzfcphbaapi.so.0`** (using ln) to the correct library.

z/OS technical notes

Thread dumps in the zOS server

By default, the Solutions Enabler server on z/OS is configured to take thread dumps to spool via Language Environment dynamically allocated files. In the event of a thread crash, these are the issued messages:

```
ANR0222E ConditionHandler invoked on thread 13, writing dump to DD
DMP00013
ANR0223E Dump to DMP00013 is complete; thread 13 will be terminated
```

You might prefer that these thread dumps are not written to spool. To do this, use DD SYM$ENV in the server's JCL and add this environment variable:

```
SYMAPI_LE_DUMP_LOGDIR = 1
```

Using this variable redirects the dump output to files in the Solutions Enabler Installation log directory. These files will have the same name as seen in the ANR0222E message text. Restart the server after changing the environment variable file.

In the event of a thread crash, the following messages are issued:

```
ANR0222E ConditionHandler invoked on thread 3, writing dump to DMP00003 in SYMAPI log
directory
ANR0223E Dump to DMP00003 in SYMAPI log directory is complete; thread 3 will be terminated
```

The `log` directory contains the thread dump output.

For example:

```
# ls /var/symapi/
log
DMP00003
```

#04DDDEF

Since you will install Solutions Enabler into the same SMP/E zone as the SSCF720 FMID, `#04DDDEF` will replace temporary and permanent DD definitions in the target and distribution zones. If these DD definitions do not exist, the SMP/E DDDEF REPLACE statement will end with condition code 4, indicating that there was nothing to replace. This is generally expected, and is not an error.

#05RECEV

The `#05RECEV` job may end with a return code of 16 if your site default assembler version does not point to IEV90. The SMP/E message is:

```
GIM23401T ** PROGRAM IEV90 WAS REQUIRED FOR SMP/E PROCESSING BUT WAS NOT AVAILABLE.
GIM20501I      RECEIVE PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE WAS 16.
```

If this message appears, customize and run the ASMHA job provided in the RIMLIB and then resubmit `#05RECEV`.

#12CNTRL

The `#12CNTRL` job disables control operations which are now enabled by default in Solutions Enabler.

STEPLIB APF authorization

Since Solutions Enabler needs SCF to run, the SCF link lib must be included in the STEPLIB concatenation for the RMLIB member `#STORAPI`. The SCF link library must already be authorized for SCF to execute, so if SCF is active, and if the Solutions Enabler load library is APF authorized, the APF requirements for `#STORAPI` are satisfied.

Note that the SCF link library may also be specified in the system link list or LPA, in which case you may comment out the DD statement that points to the SCF link library.

Disabling control functions

⚠ CAUTION

The `#12CNTRL` job disables control operations which are now enabled by default in Solutions Enabler.

⚠ CAUTION

All control functions are enabled by default.

All control functions are now enabled when shipped. As control functions are enabled, they will execute in an unprotected state on the z/OS host. For example, control functions allow remote open system hosts/clients to establish and split BCV and SRDF pairs from outside the IBM host. A zap is provided in the RIMLIB (job `#12CNTRL`). This zap allows these functions to be disabled, should a site determine that it is necessary.

Security considerations if you do not disable control functions

Use caution when leaving these functions enabled, as security checks are not performed. If security is an issue at your installation, do not leave the control functions enabled. For complete information, refer to the *EMC Solutions Enabler Installation Guide*.

HP-UX technical note

The following technical note is specific to HP-UX operating environments:

HP applications link-edited with prior versions of Solutions Enabler

Applications link-edited with Solutions Enabler 7.2.1 or earlier on any HP platform including PA-RISC 64-bit, and HP Itanium, may experience problems. The problem will be seen during initialization with an error message indicating an unresolved symbol has been detected. Refer to Knowledgebase article EMC269976 available on EMC Support.

OpenVMS technical note

A CLI runtime problem occurs on OVMS 8.4 hosts running on Itanium hardware. This problem only occurs with some CLI commands, such as `symsnap`. The symptom is an error message like this:

```
%DCL-W-ACTIMAGE, error activating image EMC$LIBSTORPDS
-CLI-E-IMAGEFNF, image file not found
$1$DKA0:[SYS0.SYSCOMMON.][SYSLIB]EMC$LIBSTORPDS.EXE;
```

The resolution to this problem is documented in EMC Knowledgebase article EMC278037.

Hyper-V technical notes

By default, SCSI commands are filtered in Hyper-V in Windows Server 2008 R2. In order to use Solutions Enabler in a guest partition, this filtering must be bypassed as recommended in Planning for Disks and Storage article in the Microsoft TechNet Library.

The following PowerShell script, executed from the parent partition will disable filtering for each child partition listed as arguments to the script. The settings are persistent, but will require a restart of the virtual machine to take effect. The script is provided as an example as-is, and includes no validation or error checking functionality.

```
$Target = $args[0]

$VSManagementService = gwmi MSVM_VirtualSystemManagementService
   -namespace "root\virtualization"

foreach ($Child in Get-WmiObject -Namespace root\virtualization
   Msvm_ComputerSystem -Filter "ElementName='$Target'")

{
$VMData = Get-WmiObject -Namespace root\virtualization -Query
   "Associators of {$Child}
Where ResultClass=Msvm_VirtualSystemGlobalSettingData
AssocClass=Msvm_ElementSettingData"

$VMData.AllowFullSCSICommandSet=$true

$VSManagementService.ModifyVirtualSystem($Child,
   $VMData.PSBase.GetText(1)) |
out-null

}
```

The following PowerShell script, executed from the parent partition will display the current filtering status of each child partition listed as arguments to the script. The script is provided as an example as-is, and includes no validation or error checking functionality.

```
$Target = $args[0]
```

```
foreach ($Child in Get-WmiObject -Namespace root\virtualization
Msvm_ComputerSystem -Filter "ElementName='$Target'")

{
$VMData = Get-WmiObject -Namespace root\virtualization -Query
   "Associators of {$Child}
Where ResultClass=Msvm_VirtualSystemGlobalSettingData
AssocClass=Msvm_ElementSettingData"

Write-host "Virtual Machine:" $VMData.ElementName
Write-Host "Currently ByPassing SCSI Filtering:"
   $VMData.AllowFullSCSICommandSet
}
```

For more information, refer to the *EMC Symmetrix with Microsoft Hyper-V Virtualization* white paper available on EMC Support.

Hyper-V Server setup

In Hyper-V setups where Solutions Enabler is installed on VMs, the VM names must match the hostnames of the VMs. This ensures that the `syminq` commands on VMs work properly.

Hyper-V gatekeepers

At least three unique gatekeepers must be assigned to each virtual machine, as a pass-through disk, to provide Solutions Enabler capabilities to each virtual machine. Based on the number of applications running on the guest, more gatekeepers may be required.

For detailed information on gatekeeper management, refer to the *EMC Solutions Enabler Installation Guide*. For specific gatekeeper sizing recommendations for all array configurations, refer to Knowledgebase article EMC255976 available on EMC Support.

SIU support for Hyper-V guest OS

Symmetrix Integration Utilities now supports Hyper-V guest operating systems on Windows Server 2008 R2 (and above). Refer to Hyper-V technical notes on page 244 for details about configuring a Hyper-V environment.

Note

Only Windows Server editions are supported as Hyper-V guest operating systems.

SIU support for multiple log files

Symmetrix Integration Utilities (SIU) supports multiple log files for concurrent execution of `symntctl` commands. This can be achieved by setting the environment variable SYMNTCTL_LOGFILE_NAME to a custom log file name in each command prompt window. Alternatively, if it is set as a system environment variable, SIU will always log all entries into the custom log file specified.

Virtual Appliance technical notes

Linux only support when using ovftool

Virtual Appliance deployment using the ovftool is supported only on the Linux platform. It is not supported on the MicroSoft Windows platform.

Daemon behavior during import/export operations

To ensure the integrity of persistent data, all active Solutions Enabler daemons will be shutdown during any import/export operation of persistent data. This causes an interruption in the daemon service. The daemons will automatically restart at the end of an import/export operation.

Login page cursor not focused

After launching the Virtual Appliance from Firefox, the cursor does not default to the **User** field. If you click Alt+Tab, leaving the application and then returning to it, the cursor will be in the **User** field. You can also place the cursor in the **User** field manually. This issue applies to the Firefox browser only.

Server hostname requirement

For the Virtual Appliance to resolve a hostname, you should only use a fully qualified hostname (as entered in DNS server) while configuring nethosts and ESX Servers (for adding gatekeeper devices).

SSL certificate generation

The Virtual Appliance generates an SSL certificate (storsrvd - client/server setup) during the initial boot after the IP address is provided, and during every IP change or reboot.

Gatekeeper devices

The Virtual Appliance will not allow more than 14 gatekeeper devices to be added to the Virtual Appliance. Attempting to add more than 14 gatekeepers returns an error message.

Host ESX Server configuration

Host ESX Server authentication is validated each time the **GateKeeper Config** tab is selected. If the authentication fails, the Host ESX Server login credentials and hostname information will be removed from Virtual Appliance records and must be added again.

SMC daemon service

When SMC daemon service is shutdown from the vApp Manager, the user is logged out of the Virtual Appliance and the browser is closed.

Flash Player version

Adobe Flash Player version 11.2 or higher is required for running the Virtual Appliance on a web browser.

Changing the IP address

Stop all daemons with the vApp Manager before changing the IP address of the appliance.

SYMCLI commands executed/submitted as root

When using the vApp Manager with the seconfig account, SYMCLI commands are executed/submitted as root.

Least privileged permission requirements

Consult the appropriate VMware documentation for guidance on the least privileged permissions required to deploy a virtual appliance.

CHAPTER 8

Gatekeeper Device Configuration

This chapter describes the function of gatekeepers and how to create them.

Overview

Solutions Enabler is an EMC software component used to control the storage features of VMAX arrays. It receives user requests via CLI, GUI, or other means, and generates system commands that are transmitted to the VMAX array for action.

Gatekeeper devices are LUNs that act as the target of command requests to Enginuity-based functionality. These commands arrive in the form of disk I/O requests. As more commands are issued in parallel from the host, and as the commands grow in complexity, more gatekeepers will be required to handle the commands in a timely manner.

A gatekeeper is not intended to store data and is usually configured as a small device. Users are encouraged to not build gatekeepers in larger sizes as the small size can be used as a characteristic to locate gatekeepers. Gatekeeper devices should be mapped and masked to single hosts only and should not be shared across hosts.

Starting with Enginuity 5876, multipath gatekeeper support has been expanded beyond using PowerPath to include a limited set of third-party multipathing solutions on a limited set of platforms.

Note

For specific gatekeeper sizing recommendations for all configurations, refer to EMC Knowledgebase solution emc255976 available on EMC Online Support.

How SYMCLI uses gatekeepers

When selecting a gatekeeper to process system commands, Solutions Enabler starts with the highest priority gatekeeper candidate (Priority 1, as described in Gatekeeper candidates on page 250). If there are no gatekeeper candidates at that priority, or the device is not accessible or currently in use, then Solutions Enabler tries to use the remaining gatekeeper candidates, in priority order, until it successfully obtains a gatekeeper, or it has tried all gatekeeper candidates.

When Solutions Enabler successfully obtains a gatekeeper, it locks the device, and then processes the system commands. Once Solutions Enabler has processed the system commands, it closes and unlocks the device, freeing it for other processing.

If the base daemon is performing gatekeeper management, gatekeepers are opened and locked, then used repeatedly to process system commands. The base daemon closes and unlocks gatekeepers after they have not been used for at least 60 seconds.

Gatekeeper candidates

Solutions Enabler selects certain devices from the list of all PDEVs to be gatekeeper candidates and automatically excludes the following PDEVS from the candidate list:

- BCVs
- Meta devices
- Virtual devices (VDEVs)

Note

From HYPERMAX OS 5977, gatekeepers must always be thin devices.

Solutions Enabler selects a gatekeeper from the candidate list based on a pre-established priority scheme. The gatekeeper priority list includes all gatekeeper candidates prioritized from the highest to the lowest, as shown below:

1. Small (< 10 cylinders) devices, marked by the storage array with the inquiry gatekeeper flag.
2. Standard non-RDF and non-metadevices.
3. RDF R1 devices.
4. RDF R2 devices.
5. VCM/ACLX devices.

Using the gkavoid and gkselect files

The `gkavoid` file specifies the VMAX devices that should not be used as gatekeepers. The gatekeeper avoidance file contains physical device names with one PdevName (`/dev/rdsk/c2t0d1s2`) per line.

The `gkselect` file specifies only those VMAX devices to be used as gatekeepers. The file contains physical device names, with one PdevName (for example, `/dev/rdsk/c2t0d1s2`) per line.

When determining which of these files is appropriate for your environment, consider the following:

Note

In the following list, *data device* refers to a non-dedicated gatekeeper device.

- If too many gatekeepers are in the `gkavoid` file, Solutions Enabler may end up selecting a *data device* as a gatekeeper. This could potentially cause significant impact on host application performance.
- If there are not enough gatekeepers in the `gkselect` file, array control operations may time out. However, no extra maintenance is required when adding new *data devices*, as would be necessary when using only the `gkavoid` file.

Note

If there are no devices listed in the `gkselect` file for a particular VMAX array, or if all of the devices listed in the file are offline or do not exist at the time the file is read, then normal gatekeeper selection rules apply, as explained in Gatekeeper candidates on page 250. This may also result in Solutions Enabler choosing a data device as a gatekeeper and that could impact host application performance. (The base daemon picks up all changes to the `gkselect` and `gkavoid` files dynamically.)

Note

If a device is listed in both the `gkavoid` file and the `gkselect` file, the device will be avoided.

Sizing gatekeepers

When a VMAX array is installed, the EMC Customer Engineer selects and configures VMAX devices with less than 10 cylinders (less than 5 MB) for use as gatekeeper devices.

However, the gatekeeper device must be at least as large as the minimum volume size accessible by your host, which is usually, 6 cylinders, 2.8 MB. Consult your host documentation for the minimum device size accessible by your particular host to determine the minimum gatekeeper device size for your environment.

Note

For specific gatekeeper sizing recommendations for all array configurations, refer to EMC Knowledgebase article emc255976 available on EMC Online Support.

You can determine the storage size of a VMAX device using:

- The `sympd` command using the `list` and `show` arguments as follows:
  `list` — Displays a list of physical device names and storage size (in MBs) for a specific VMAX array.

  `show` — Displays the parameters of a specified physical device that includes the device capacity or size in blocks and megabytes.
- The `syminq` command and specifying the physical device name.

Note

Sometimes the EMC Customer Service Engineer configures a few VMAX devices for use as dedicated gatekeepers. You can distinguish these devices in the output of the `syminq` command by locating a symbol `GK` next to the PdevName (physical device name). Devices listed in the `gkselect` file are not required to have the GK attribute, though it is highly recommended. Listing non-dedicated gatekeeper devices in the file may cause significant impact on host application performance.

Note

For Windows platforms in a clustered environment, gatekeepers must be a minimum of 8 MB in size and have a signature. In a non-clustered environment, gatekeeper devices smaller than 8 MB will show up in the new Disk Manager as devices with no available information. (Disk Manager just displays the disk number and a blank bar.) The devices are still addressable at the SCSI level, and SYMCLI scripts continue to work. (There may be some implications for device naming, since the Windows Device Manager does not create some of the normal device objects for devices smaller than 8 MB).

Note

For specific gatekeeper sizing recommendations for all array configurations, refer to EMC Knowledgebase article emc255976 available on EMC Online Support.

VMware setup

Unique gatekeepers must be assigned to each virtual machine, as a raw device, to provide Solutions Enabler capabilities to each virtual machine. Individual applications may have specific requirements for gatekeepers.

For specific gatekeeper sizing recommendations for all array configurations, refer to Knowledgebase article EMC255976 available on EMC Support.

Creating gatekeeper devices

The `symconfigure` command automates the process of creating gatekeeper devices. These gatekeeper devices are sized as follows:

- Enginuity 5771 or higher — 3 cylinders
- Enginuity versions lower than 5771 — 6 cylinders

Both sizes of gatekeeper devices are protection type RAID1.

Use the following syntax in a command file to create gatekeeper devices:

```
create gatekeeper count=n,
       emulation=EmulationType,
       [, type=thin
         [, binding to pool=<PoolName>]]
       [, mvs_ssid=n]
       [, sg=<SgName>]
       [, [mapping to dir DirNum:PortNum
       [starting] target = scsi_target ,
       lun=scsi_lun, vbus=fibre_vbus
       [starting] base_address=cuu_address]...]
       [host_id=compatible|native];
```

Where:

`count` — Indicates the number of devices to create.

`emulation`— Specifies the device emulation type.

`type=thin` — Specifies that the gatekeeper is a thin gatekeeper.

`binding to pool` — Specifies the existing device pool to which the newly created thin GK should be bound.

`mvs_ssid` — Specifies the subsystem ID group value for the newly created device.

`sg= <SgName>` — Specifies the SG to which the gatekeeper is added upon creation.

`mapping to dir`— Specifies the director/port addresses to which the newly created gatekeeper should be mapped.

`target` — Indicates a hex value for the SCSI target ID.

`lun` — Indicates a hex value for the SCSI logical unit number.

`vbus` — Specifies the virtual bus address if mapping to an FA port using volume set addressing.

`base_address` — Indicates a base or alias address for a device being mapped to an EA or EF port.

`host_id` — Indicates the host ID format, that is either the new Federated ID format (NATIVE) or an ID compatible with the previous ID format (COMPATIBLE) that is a non-portable ID value only unique within the array. Additionally, you can change the device's host ID on an existing device to either a native ID or a compatible ID.

Restrictions
On Enginuity versions lower than 5874, this command only allows the creation of a gatekeeper device. It does not allow the mapping of the newly created device to be performed at the same time as the creation of the new device.

On HYPERMAX OS 5977, this command only allows the creation of thin gatekeeper devices.

Native ID is not supported for iSeries (D910_099) devices.

The following restrictions apply for SBC and VMAXe series platforms:

- You are not allowed to create any disk group provisioned devices using the `create dev` command, except for DATA devices. It is advised to use the `create gatekeeper` command introduced in Solutions Enabler V7.3 to create gatekeeper devices.
- A gatekeeper device created with the `create dev` command will have a fixed size of 6 cylinders for DMX 800/1000/2000/3000, and 3 cylinders for DMX-3, DMX-3 950, DMX-4, DMX-4 950 or higher. There are no options to specify other device sizes.
- The gatekeeper device created using the `create dev` command has a fixed protection type of RAID 1. There are no options to specify another device protection type.

Displaying gatekeeper information

The `stordaemon` commands in this section display information on gatekeeper usage.

Displaying gatekeeper statistics

To display information on the number of gatekeeper candidates, dedicated gatekeepers, unique gatekeepers, open gatekeepers, and gatekeeper utilization information, use the following command:

```
stordaemon action storapid -cmd show -gk_stats [-sid SymmID]
```

Where:

SymmID specifies the VMAX array for which you want to display information. Issuing this command without the `-sid` option will display information on all storage arrays.

For example:

```
stordaemon action storapid -cmd show -gk_stats -sid 343
```

And the above command produces output similar to the following:

```
G A T E K E E P E R   S T A T I S T I C S

Symmetrix ID: 000195700343

                             Total Paths      Unique Paths
                             -----------      ------------
  Pdevs                              232               232
  GK Candidates                      232               232
  Dedicated GKs                       40                40
  VCM/ACLX devs                        0                 0

  Pdevs in gkavoid                    32
  Pdevs in gkselect                    0

  Max Available GKs                    8
  Num Open GKs                         3
```

```
Gatekeeper Utilization
  Current                          0 %
  Past Minute                     10 %
  Past 5 Minutes                  11 %
  Past 15 Minutes                 11 %
  Since Midnight                   0 %
  Since Starting                   0 %

Highwater
  Open Gatekeepers                 4
  Time of Highwater               01/19/2014 10:57:03

  Gatekeeper Utilization          25 %
  Time of Highwater               01/19/2014 09:48:07

Gatekeeper Timeouts
  Since starting                   0
  Past Minute                      0
  Time of last timeout            N/A
```

Displaying gatekeeper candidates and gatekeeper states

To display which devices are gatekeeper candidates and the state of each gatekeeper (opened or closed), use the following command:

```
stordaemon action storapid -cmd show -gk_pdevs [-sid SymmID] [-v]
```

Where:

*SymmID*specifies the storage array for which you want to display information. Issuing this command without the `-sid` option will display information on all storage arrays. The `-v` option specifies to display a verbose listing.

For example:

```
stordaemon action storapid -cmd show -gk_pdevs -sid 343
```

APPENDIX A

Host specific behaviour running Solutions Enabler

This section describes the issues in running Solutions Enabler on various hardware platforms. You will find additional information in the Release Notes, which are distributed in hard copy with the Solutions Enabler kits.

The information in this section is organized by hardware platform and operating system:

General issues

This section describes issues that apply to all supported platforms.

Host system semaphores

Note

This section only applies if you manually changed the `storapid:use_all_gks` to `disabled` in the `daemon_options` file. Otherwise, this section may be skipped.

In UNIX and Linux environments, Solutions Enabler uses semaphores to serialize access to the gatekeeper devices. You or the System Administrator may need to optimize the host system semaphore parameter settings. When optimizing the semaphore parameters, the following values are recommended:

- `semmni` — Specifies the number of semaphore identifiers for the host. Solutions Enabler requires one identifier for each gatekeeper, and one for each SYMAPI database. The minimum recommended value for this parameter is 256.
- `semmns` — Specifies the number of semaphores for the host. Solutions Enabler requires one semaphore for each gatekeeper, and one for each SYMAPI database. The minimum recommended value for this parameter is 256.
- `semmnu` — Specifies the number of undo structures for the host. Solutions Enabler requires one undo structure for each gatekeeper, and one for each SYMAPI database. The minimum recommended value for this parameter is 256.
- `semume` — Specifies the number of undo structures per process. The minimum recommended value for this parameter is 256.

RDF daemon thread requirements

The RDF daemon allocates threads based on the number of locally attached Symmextrix arrays visible to its host. On some host operating system configurations the default number of threads allowed per process may not be enough to accommodate the RDF daemon's requirements. Although the exact number of threads needed for a given daemon cannot be exactly predicted, the recommended practice is to allow 16 threads per locally attached VMAX array.

HP-UX-specific issues

This section describes the HP-UX system issues concerned with compatibility with the SYMCLI/SYMAPI database file, gatekeeper, and BCV device requirements.

Creating pseudo-devices for gatekeepers and BCVs

If the device you want to use as a gatekeeper or BCV device is accessed through the HP-PB (NIO) SCSI bus controller and you want the device to be visible to your host, you must create a pseudo-device for that device. (A pseudo-device is necessary for every device you want visible to the host.)

Note

Your HP-UX operating system may require a patch to support the HP-PB (NIO) SCSI board. Patches for the HP-PB SCSI Pass-Thru driver (spt0) are available for HP-UX V11.20 and higher from HP on an Extension Media CD. Consult your HP representative about spt drivers for your specific system.

Note

If your HP system is configured with an HSC fast-wide differential SCSI interface board and a device accessed through the HSC SCSI bus is available, you can specify the gatekeeper devices through the procedure outlined in the *EMC Solutions Enabler Array Management CLI User Guide*.

To create pseudo-devices and specify devices as gatekeepers and BCV devices:

Procedure

1. Execute the `ioscan` command and find the full pathnames of the gatekeeper and BCV devices.

   For example, the full pathname of the array volume designated to be the gatekeeper is `/dev/rdsk/c1t2d1`.

2. Enter the `lsdev` command and note the output. For example:

   ```
   lsdev -d spt0
   Character     Block   Driver     Class
       80           -1     spt0       spt
   ```

   Note

   The wide SCSI Pass-Thru is identified as spt0. If there is no output response to this command, the spt0 driver is missing. Install the proper driver before proceeding.

   Note

   There is also an spt driver. The spt driver will not work in this environment.

3. Create the device node for the gatekeeper device.

   Note

   This step creates a pseudo-device that is incapable of functioning like a normal device. It can only be used as a gatekeeper device or to process TimeFinder control functions directed to a BCV device.

   For example, to create the device node:

   ```
   mknod /dev/rdsk/pseudo_c1t2d1 c 80 0x012100
   ```

   where:

`/dev/rdsk/pseudo_c1t2d1` is the full pathname of the pseudo-device associated with `/dev/rdsk/c1t2d1`.

`c` specifies character (raw) device node creation.

`80` is the character value from the output of the lsdev command. This is the major number of the device file.

`0x012100` is the minor number of the device file. The individual values of the minor number are:

- `0x` indicates that the number is hexadecimal.
- `01` is the hexadecimal number of the controller referenced by `/dev/rdsk/c1t2d1`
- `2` is the hexadecimal number of the target ID referenced by `/dev/rdsk/c1t2d1`
- `1` is the hexadecimal number of the LUN referenced by `/dev/rdsk/c1t2d1`
- `00` must be the last two digits of the minor number.

4. Repeat step 3 for all BCV devices and alternate gatekeeper devices.

   ⚠ CAUTION

   Do not perform I/O through the device (`/dev/rdsk/cxtxdx`) associated with the pseudo-device, nor use the pseudo-device as a normal device. If you do, you have two paths to the same device from two different device drivers. Unknown results may occur.

5. To create the mapping information of standard devices to pseudo-devices, create the file:

   ```
   /var/symapi/config/pseudo_devices
   ```

   For each gatekeeper and BCV device, add a mapping to a pseudo-device. For example, in the `pseudo_devices` file, add the following line to map the pseudo-device filename (in **bold**), to the array device file:

   ```
   /dev/rdsk/c1t0d0     /dev/rdsk/pseudo_c1t0d0
   ```

   SYMAPI will then use this pseudo-device instead of the physical device file name.

   When the `SymDiscover()` function is used, the pseudo-device mappings get posted in the log file (`/var/symapi/log/symapi*.log`).

swverify command not supported

The native UNIX command **`swverify`** is not supported from Solutions Enabler V7.6 and higher.

HP OpenVMS-specific issues

The default client/server communication security level is SECURE (on platforms that will support it). This can cause communication failures between OpenVMS hosts and non OpenVMS hosts since OpenVMS does not support secure communication. To workaround this, you must change the security level on the host which the OpenVMS CLI commands will connect (SYMCLI_CONNECT) to ANY. For instructions, refer to the *EMC VMAX Family Security Configuration Guide*.

IBM AIX-specific issues

This section describes the IBM AIX system issues concerned with Oracle database mapping and rebooting a system.

Oracle database mapping

Oracle 8 database mapping with SYMCLI is supported on 32-bit AIX V4.3 and above.

You may need to create the Oracle library, `libclntsh.so`.

To determine if the library exists for Oracle 8, execute the following:

```
ls $ORACLE_HOME/lib/libclntsh.so
```

If the library does not exist, execute the following command:

```
make -f $ORACLE_HOME/rdbms/lib/ins_rdbms.mk client_sharedlib
```

The Oracle 8 OCI executable is linked dynamically. You must set the following environment variable as follows:

```
setenv LIBPATH $ORACLE_HOME/lib
```

BCV devices lost after reboot

When a system comes back up after a reboot, it will not recognize your mapped BCVs. To work around this problem, you should run the following special BCV script (`mkbcv`):

```
cd /
./inq.AIX | more  (look for no gaps in the numbers, ie.. rhdisk0,
rhdisk1, rhdisk3... - rhdisk2 is missing)
cd /usr/lpp/Symmetrix/bin
./mkbcv -a ALL
cd /
./inq.AIX | more  (look for no gaps in the numbers, ie.. rhdisk0,
rhdisk1, rhdisk2... - rhdisk2 is not missing)
```

It is recommended to have `./mkbcv -a ALL` in your AIX boot procedures.

Note

`inq.AIX` can be found on the EMC FTP site.

APPENDIX B

Solutions Enabler Directories

This appendix contains the directory list for UNIX, Windows, OpenVMS and UNIX System Services directories for z/OS installations:

UNIX directories

Table 41 on page 264 lists the directories for UNIX platforms. Your directories may differ from this list since the location of these directories is configurable at installation.

Table 41 UNIX directories

Contents	Directories	Details
Binaries for executables	/usr/storapi/storbin /usr/storapi/bin	STORCLI binaries. SYMCLI binaries.
Shared libraries	/usr/storapi/shlib	All shared libraries.
Database engines	/usr/storapi/shlib/sql/IBMUDB/ /usr/storapi/shlib/sql/ORACLE/ /usr/storapi/shlib/sql/SYBASE/	IBM database engine. Oracle database engine. Sybase database engine.
Language interfaces	/usr/storapi/interfaces/java/ /usr/storapi/interfaces/xml/	Java language interface. XML examples.
SYMCLI manpages	/usr/symcli/storman/man3 /usr/symcli/man/man1 /usr/symcli/man/man3	STORCLI and STORAPI man pages. SYMCLI man pages. SYMAPI and CLARAPI man pages.
SYMAPI Message Catalogs	/usr/storapid/locales/en	SYMAPI Error Message Catalog for English.
Daemons	/usr/symcli/daemons/	Location of the daemon executables.
Configuration database file(s)	/var/symapi/db/	Contains the configuration database file(s) for SYMAPI, CLARAPI, and STORAPI.
SYMAPI environment and system files	/var/symapi/config	Includes licenses, avoidance, options, `daemon_options`, `daemon_users`, and `nethost` files.

Table 41 UNIX directories (continued)

Contents	Directories	Details
		It is recommended that you back up this directory frequently.
SYMAPI certificate files	/var/symapi/config/cert	Contains server and trusted certificate files and support files for certificate creation. Used for client/server security.
Security data	/var/symapi/authz_cache	Acts as a cache of authorization data from attached storage arrays.
Log files	/var/symapi/log	Contains SYMAPI logs and daemon logs.

Windows directories

Table 42 on page 265 lists the default directories for Windows. Your directories may differ from this list since the location of these directories is configurable at installation.

Table 42 Windows directories

Contents	Directories	Details
Binaries for executables	C:\Program Files\EMC\SYMCLI\storbin C:\Program Files\EMC\SYMCLI\bin	STORCLI binaries. SYMCLI binaries.
Shared libraries	C:\Program Files\EMC\SYMCLI\shlib	All shared libraries.
Database engines	C:\Program Files\EMC\SYMCLI\shlib\sql\Oracle C:\Program Files\EMC\SYMCLI\shlib\sql\SQLSERVER C:\Program Files\EMC\SYMCLI\shlib\sql\ASM	Oracle database engine. SQL server database engine. ASM database engine.

Table 42 Windows directories (continued)

Contents	Directories	Details
Language interfaces	C:\Program Files \EMC\SYMCLI \interfaces\java C:\Program Files \EMC\SYMCLI \interfaces\xml \examples C:\Program Files \EMC\SYMCLI \interfaces\xml\docs	Java language interface, JAVA and jar files. XML examples. XML docs.
SYMCLI manpages	C:\Program Files \EMC\SYMCLI \storman\man3 C:\Program Files \EMC\SYMCLI\man \man1 C:\Program Files \EMC\SYMCLI\man \man3	STORCLI and STORAPI man pages. SYMCLI man pages. SYMAPI and CLARAPI man pages.
Daemons	C:\Program Files \EMC\SYMCLI \daemons	Location of the daemon executables.
SYMAPI Message Catalogs	C:\Program Files \EMC\SYMCLI \locales\en	Location of the SYMAPI Error Message Catalog for English.
Configuration database file(s)	C:\Program Files \EMC\SYMAPI\db	Contains the configuration database file(s) for SYMAPI, CLARAPI, and STORAPI.
SYMAPI environment and system files	C:\Program Files \EMC\SYMAPI \config	Includes licenses, avoidance, options, and server network files. It is recommended that you back up this directory frequently.
SYMAPI certificate files	C:\Program Files \EMC\SYMAPI \config\cert	Contains server and trusted certificate files and support files for certificate creation. Used for client/server security.

Table 42 Windows directories (continued)

Contents	Directories	Details
Security data	C:\Program Files \EMC\SYMAPI \authz_cache	Acts as a cache of authorization data from attached storage arrays.
SYMAPI log files	C:\Program Files \EMC\SYMAPI\log	Contains SYMAPI logs and daemon logs.
Providers	C:\Program Files \EMC\SYMCLI\shlib	VSS Provider.
Installer logs files	C:\Program Files \EMC\SYMAPI \InstallerLogs %TEMP% \SE_RTinstall_Verbose.log	Contains all installation related files.
Provider SMI	C:\Program Files \EMC\ECIM	Contains all ECOM related files.
Debug log files	C:\Program Files \EMC\SYMAPI \Debug	Contains Debug log files.

OpenVMS directories

Table 43 on page 267 lists the default directories for OpenVMS. Your directories may differ from this list since the location of these directories is configurable at installation.

Table 43 OpenVMS directories

Contents	Directories	Details
Binaries for executables	SYMCLI$BIN	STORCLI binaries. SYMCLI binaries.
Shared libraries	SYMCLI$SHLIB	All shared libraries.
SYMCLI man pages	SYMCLI$HELP	STORCLI man pages. STORAPI man pages. SYMCLI man pages. SYMAPI and CLARAPI man pages.
SYMAPI Message Catalogs	EMC$ROOT: [emc.symcli.locales.en]	Location of the SYMAPI Error

Table 43 OpenVMS directories (continued)

Contents	Directories	Details
		Message Catalog for English.
Configuration database file(s)	SYMAPI$DB	Contains the configuration database file(s) for SYMAPI, CLARAPI, and STORAPI.
SYMAPI environment and system files	SYMAPI$CONFIG	Includes licenses, avoidance, options, `daemon_options`, and `netcnfg` files. It is recommended that you back up this directory frequently.
SYMAPI log files	SYMAPI$LOG	Contains SYMAPI logs and daemon logs.

z/OS Unix System Services directories

Table 44 on page 268 lists the Unix System Services directories for z/OS. Your directories may differ from this list since the location of these directories is configurable at installation.

Table 44 z/OS directories

Contents	Directories	Details
Configuration database file(s)	/var/symapi/db/	Contains the configuration database file(s) for SYMAPI, CLARAPI, and STORAPI.
SYMAPI environment and system files	/var/symapi/config	Includes licenses, avoidance, options, `daemon_options`, `daemon_users`, and `nethost` files. It is recommended that you back up this directory frequently.
SYMAPI certificate files	/var/symapi/config/cert	Contains server and trusted certificate files and support files for certificate

Table 44 z/OS directories (continued)

Contents	Directories	Details
		creation. Used for client/server security.
Security data	/var/symapi/authz_cache	Acts as a cache of authorization data from attached storage arrays.
Log files	/var/symapi/log	Contains SYMAPI logs and daemon logs.
SYMAPI Message Catalogs	/usr/storapi/locales/en	Contains the SYMAPI Error Message Catalog for English.

APPENDIX C

UNIX Installation Log Files

This appendix describes the UNIX log files created by the Solutions Enabler install script:

Understanding the UNIX installer log files

The Solutions Enabler installer script `se8300_install.sh` creates log files in install root directory `/opt/emc/logs`.

Format
The log files are named using the following convention:

```
SE_NI_<V M.m.P>_<TimeStamp>.log
```

For example:

```
SE_NI_V8.3.0.110525_175707.log
```

Where:

SE	Solutions Enabler
NI	Native installation
V	Letter portion of version
M	Version major
m	Version minor
P	Version point
TimeStamp	File creation time stamp in the format:*yymmdd_hhmmss*

Log file contents
The log files contain the following information:

- Date
- Script name
- User running the script
- Operating system and hardware type
- Script command line options
- Location of native install (NI) kit if the kit is found
- Previous Install root directory
- Previous working root directory
- Install root directory
- Minimum operating system version required
- Existing operating system version in system
- Installed product version
- Current product Version
- Selected components

- Information on active processes (if any)
- Information on active daemons (if any)
- Information on active components
- Package/fileset/rpm being installed/uninstalled
- List of files installed by package/fileset/rpm only during install
- Successful completion of install /uninstall

Note

In addition to the above information, the log files will also contain operating system-specific information useful in trouble shooting native installations.

INDEX

U

W

Z