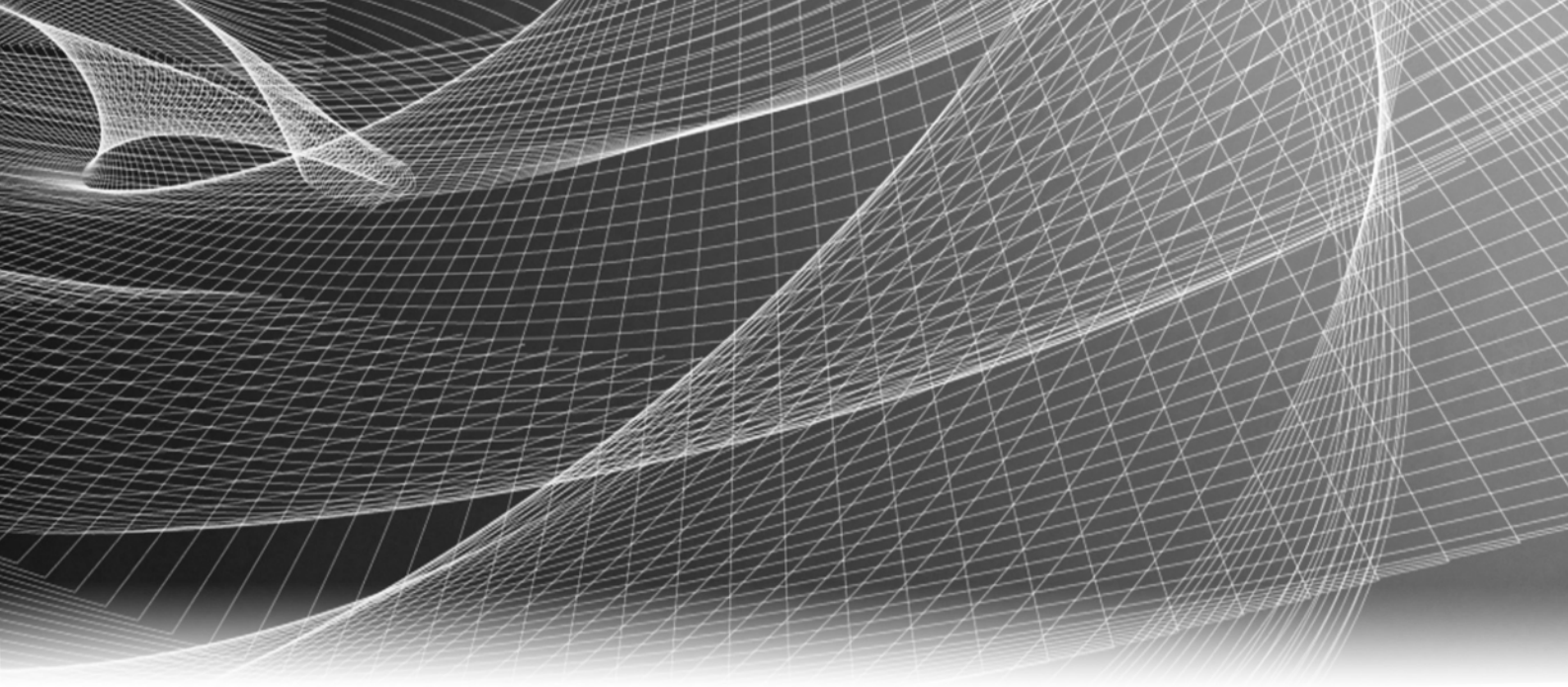


EMC® Solutions Enabler

Version 7.5

Installation Guide

P/N 300-014-868
REV 02

EMC²

Published November, 2012

CONTENTS

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 the EMC Online Support. Check the EMC Online Support 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.

Audience

This guide provides installation procedures for installing the EMC Solutions Enabler software for your specific platform. The EMC Solutions Enabler software provides your host system with an API shared library and a special command set that comprises the Symmetrix Command Line Interface (SYMCLI). (For the z/OS platform, only the SYMAPI server is available.)

Related documentation

The following EMC publications provide additional information:

- *EMC Solutions Enabler Symmetrix CLI Command Reference*
- *EMC Solutions Enabler Security Configuration Guide*
- *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*
- *EMC Solutions Enabler Symmetrix Array Controls CLI Product Guide*
- *EMC Solutions Enabler Symmetrix SRM CLI Product Guide*
- *EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide*
- *EMC Solutions Enabler Symmetrix SRDF/Star CLI Product Guide*
- *EMC Solutions Enabler Symmetrix Migration CLI Product Guide*
- *EMC Solutions Enabler Symmetrix TimeFinder Family CLI Product Guide*
- *EMC Solutions Enabler Symmetrix CLI Quick Reference*
- EMC host connectivity guides for [*your operating system*]

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.

IMPORTANT

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 Symmetrix 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 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 will help us continue to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to:

techpubcomments@emc.com

CHAPTER 1
Pre-install Considerations

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 Symmetrix arrays, but can be CLARiiON® 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:

- ❑ Obtain the software. Solutions Enabler is distributed in the following forms:
  - On the Solutions Enabler installation disc, which includes kits for all supported platforms.
  - As a platform-specific file download from the EMC Online Support at https://support.EMC.com
- ❑ Review the interoperability information in the E-Lab™ Interoperability Navigator which can be reached at http://elabnavigator.EMC.com
- ❑ Review the *EMC Solutions Enabler Release Notes*.
- ❑ If you are upgrading from a previous version on a UNIX, verify that all application processes that use the Solutions Enabler libraries and binaries are stopped. "Stopping the application processes" on page 156 provides instructions.
- ❑ 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 *rollback* to the previous version of Solutions Enabler. The location of these directories vary according to the operating system. Appendix E provides more information.
- ❑ EMC recommend that you read the *Solution Enabler Security Configuration Guide* and apply the settings after installation.

UNIX-specific tasks

The following task is specific to UNIX environments:

- ❑ AIX and OSF1 do 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 87.

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.

Solutions Enabler requires the use of EMC ResourcePak Base version 5.8.0 at a minimum. However, as EMC ResourcePak versions go out of support, you should upgrade to a version that supports your requirements.

If you have already installed ResourcePak Base Version 5.8.0 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.

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.SSEM750 as the prefix, you will allocate EMC.SSEM750.LOADLIB, EMC.SSEM750.PARMLIB, and so on.

  Note: This should generally 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 EMC 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 EMC ResourcePak Base (EMCSCF) version 5.8.0 or higher. The default value is `EMC.SSCF580`, which is the default for the ResourcePak Base product, version 5.8.0. 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 USS 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.

- 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 136 provides more information.

❑ Define the UNIX system services requirements.

The following requirements apply to the userid of the installer, the userid assigned to the started tasks, or batch jobs used to run Solutions Enabler tasks (such as, the SYMAPI server and event daemon). 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: 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 `SEMAGENT`:

  ```
  *ICH408I JOB(semagent) STEP(semagent) 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*.

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

  For example:

```
ICH408I USER(user) Group(group) Name(username) 035
035 /var/symapi CL(DIRACC ) FID(01C8C6E2F0F0F200010D000000000003)
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 Chapter 2, followed by the procedure "Install the Linux I/O module for CKD devices" on page 42.

- ❑ 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 SLES 10 is running as a guest under IBM's z/VM:

  Verify that all Symmetrix 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 Symmetrix device number (for example, on the `symdg` command).

Environment and system requirements

Consider the following when working with Solutions Enabler V7.5.0.

Host systems and Enginuity support

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

`http://elabnavigator.EMC.com`.

Disk space requirements

The disk space requirements are listed in four tables:

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

Table 1 Disk space requirements for AIX, Solaris x86, Solaris Sparc UNIX

Install components (in KBs)	AIX	Solaris x86	Solaris Sparc
Persistent data files	2814	841	841
SSL Certificate component	5	5	5
Thincore components	34353	10640	10315
Base component (base storage, base mapping, and control storage libraries	72544	26674	37779
Command line tools (optional component)	85380	52737	55320
Database mappings - SRM (optional component)	3260	5	606
SMI-S provider (optional component)	N/A	N/A	N/A
Java Native Interface (optional component)	124757	N/A	51937
Symrecover including PERL 5.8 for Star (optional component)	19500	17170	18459
Enable 64-bit component install	119870	N/A	36984

Table 2 Disk space requirements for HP-UX, HP-UX ia64, Linux, and Linux ia64

Install components (in KBs)	HP-UX	HP-UX (ia64)	Linux	Linux (ia64)
Persistent data files	2814	2814	966	970
SSL Certificate component	5	5	5	5
Thincore components	15556	34015	9682	17442
Base Component (Base Storage, Base Mapping, and Control Storage libraries	61938	78492	188772	47191
Command line tools (optional component)	82016	163420	54067	92939
Database mappings - SRM (optional component)	13319	911	733	797
SMI-S Provider (optional component)[1]	N/A	N/A	4604	N/A
Java Native Interface (optional component)	130435	N/A	50651	N/A
Symrecover including PERL 5.8 for Star (optional component)	19932	23972	17313	20294
Enable 64-bit component install	97773	3	110983	N/A

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

Provider kit.

Table 3 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	966	970	966	970
SSL Certificate component	5	5	5	5
Thincore Components	10788	13996	11651	9538
Base component (Base Storage, Base Mapping, and Control Storage Libraries	115800	28704	29107	32498
Command line tools (optional component)	53428	55690	53671	53042
Database mappings - SRM (optional component)	696	75	5	N/A
SMI-S Provider (optional component)	N/A	N/A	N/A	N/A
Java Native Interface (optional component)	51059	N/A	N/A	N/A
Symrecover including PERL 5.8 for Star (optional component)	18068	17772	1551	N/A
Enable 64-bit component install	N/A	N/A	N/A	N/A

Table 4 Disk space requirements for Windows

Install components (in MBs)	Windows (x64)	Windows (ia64)	Windows (x86)
Base component (Base Storage, Base Mapping, and control storage libraries	107	90	80
SSL Certificate component	1	1	1
Command line tools (optional component)	16	175	15
Database Mappings - SRM (optional component)	01	02	01
Java Native Interface (optional component)	38	N/A	36
Symrecover including PERL 5.8 for Star (optional component)	22	25	20

Client/server interoperability

The server component of Solutions Enabler V7.5.0 SYMAPI is compatible with the client component of older SYMAPI versions from V7.1 and up. When planning to upgrade from V7.1 to V7.5.0, it is possible to do so in a staged fashion, upgrading the servers first, and then the clients. If access to V7.5.0 enhanced features is required only from the server systems, then there is no requirement to upgrade client systems. For clients to gain access to V7.5.0 enhanced features, they must be upgraded.

Secured sessions using SSL are only available when both the client and server are running Solutions Enabler V7.1 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 28 and The *Solutions Enabler Security Configuration Guide*.

Security settings

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

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, GNS, SRM, and Star. For more information, refer to Table 17 on page 88.

Platform requirements

EMC Solutions Enabler for z/OS runs on all IBM supported releases of z/OS.

Solutions Enabler 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.

  Note: Solutions Enabler does not support older HPNS or IUCV sockets (non-integrated sockets).

- 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 USS directories were not required to be created.

The SYMAPI directory structure is required on any host running Solutions Enabler V7.1 or higher. Configuration files must reside in the `config` directory under the base directory, and log files will be stored in the `log` directory.

USS file system requirements

The following are z/OS USS file system requirements:

Logging

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

SYMAPI log file

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

USS file system options

The following USS file system options can be configured to meet your environment:

SYMAPI database

Starting with Solutions Enabler V7.1, an MVS dataset (via DD SYM$DB) is not supported. The USS file system will always be used to store the database.

Symmetrix Avoid, Gatekeeper Avoid and Select, and INQ files

If the base daemon is running and any of the select or avoid files are in use, the SYMAPI server should be configured to look for these files in USS. This configuration will eliminate the need to duplicate this information in the PARMLIB dataset, where it was previously stored. By removing the relevant DD statements (`SYM$AVD`, `SYM$GAVD`, `SYM$GSEL`, and/or `SYM$INQ`), SYMAPI will automatically look in USS for these files.

For more information on the avoidance and selection files, refer to "Avoidance and selection files" on page 134.

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

EMC 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 Symmetrix 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:

- #01ECCIN
- #SEMAGNT 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

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 V7.5.0 can only read databases previously written by Solutions Enabler V7.1 or higher. In client/server mode, Solutions Enabler V7.5.0 servers only support clients running Solutions Enabler V7.1 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 V7.1 SYMAPI library call within EMC ControlCenter attempting to access a V7.5.0 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.

VNX or CLARiiON array discovery prerequisites

Solutions Enabler supports CLARiiON FLARE 19, 22, 24, 26, 28, 29, 30 for the CX4 Series, CX3 Series, CX Series, AX Series and AX4 Series of arrays.

Solutions Enabler supports VNX OE 31 and 32 for the VNX Series of arrays.

All hosts except the following require naviseccli to be installed to discover VNX or CLARiiON arrays:

- HPUX RISC32
- Linux x32
- Linux_x64
- SunOS SPARC_32
- WINDOWS_x32
- WINDOWS_x64

Note: You can install naviseccli as part of your VNX or CLARiiON Host Software; it is not part of the Solutions Enabler kit.

Port requirements

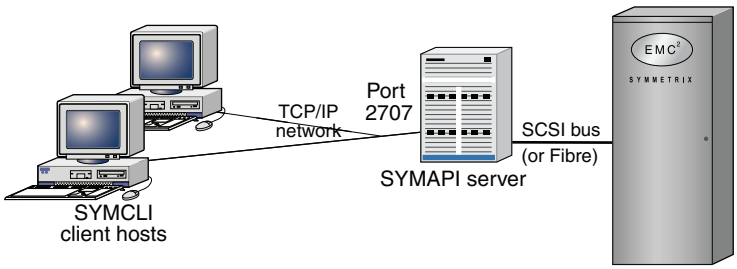
In addition, to allow any host to communicate with and discover a VNX or CLARiiON array, ports 443 or the alternate port 2163 must be opened in the firewall for discovery of CLARiiON and VNX systems.

Client or server installation

If your computer is locally connected to a Symmetrix array, go to Chapter 2. 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 Symmetrix array. The following diagram shows a Symmetrix 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 actually 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 with 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 *Solutions Enabler Security Configuration Guide* provides instructions on modifying this default behavior.

Table 5 lists the host operating systems that support SSL.

Table 5 Host operating system support for SSL

Supported operating system
AIX (32- and 64-bit)
HP-UX (32- and 64-bit) HP-UX Itanium (64-bit)
Linux (32-bit) Linux Itanium (64-bit) Linux AMD (64-bit) Linux/390 (32-bit)
Solaris (32- and 64-bit)
Windows (32-bit) Window Itanium (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:

1. Install Solutions Enabler software in the machine designated as the client, according to the procedures in Chapter 2.
2. Install the same Solutions Enabler software in the machine designated as the server, according to the procedures in Chapter 2.
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 116 provides instructions.
4. Issue a `stordaemon start storsrvd` command on the server machine. "SYMCLI through a remote server" on page 116 provides instructions.
5. Set environment variables `SYMCLI_CONNECT` and `SYMCLI_CONNECT_TYPE` on the client. "SYMCLI through a remote server" on page 116 provides instructions.

Installation cheat sheets

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

Windows installation cheat sheet

Table 6 Windows installation cheat sheet

Task	More Information	Done
Ready the environment for Solutions Enabler.	For instructions and requirements, refer to "Before you begin" on page 18 and "Environment and system requirements" on page 22, respectively.	❑
1. Change directory to the location of the Solutions Enabler kit by entering the following: `cd /Install_disc_mount_point/Windows`	N/A	❑
2. Start the installation wizard by running the following: `se7500-Windows-Processor_type.exe` Where *Processor_type* is either x86, x64, or ia64.	For information on running the installation from the command line, refer to "Using the command line" on page 44. If you select the custom installation option, Table 11 on page 43 describes the available options.	❑
1. Enable the Solutions Enabler features with the following command: `symlmf add`	For more information, refer to "Licensing your software" on page 62.	❑
2. Build the SYMAPI database by entering the following command: `symcfg discover`	For more information, refer to "Building the SYMAPI database" on page 80.	❑
3. Set the environment variables so you can directly 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`	For more information, refer to "Setting environment variables" on page 80.	❑
4. *Optional*: Read the *Solutions Enabler Security Configuration Guide* and apply related security settings.	for more information, refer to *Solutions Enabler 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 84.	❑
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 85.	❑
7. *Optional*: Configure the necessary daemons for the environment.	For instructions, refer to: • "Setting up daemons for distributed application support" on page 87 • "Managing the base daemon" on page 92 • "Setting up the event daemon for monitoring" on page 94	❑

UNIX installation cheat sheet

Table 7 UNIX installation cheat sheet

Task	More Information	Done
Ready the environment for Solutions Enabler.	For instructions and requirements, refer to "Before you begin" on page 18 and "Environment and system requirements" on page 22, respectively.	❑
1. Mount DVD	For operating system-specific commands, refer to "Step 1: Mount the installation DVD" on page 34.	❑
2. Run the installation script. For example, to run the full interactive script, enter the following command: `./se7500_install.sh -install`	For information on running alternative installation methods, such as silent, incremental, or response file, refer to "Step 2: Run the install script" on page 34.	❑
3. Verify the installation by entering the following command: `./se7500_install.sh -check`	For more information, refer to "Verifying your installation" on page 41.	❑
4. *Optional*: Remove the temporary file: `/tmp/emc_app_data_path`	For more information, refer to "Removing temporary file" on page 41.	❑
5. Unmount the installation disc by entering the following command: `umount` *`mount_point`*	N/A	❑

6. In a Linux on System z installation on Novell SLES 10, install the Linux I/O module for CKD devices.
For example, to load the kernel object in a SLES 10 Service Pack 1 environment, enter:
`cd /usr/symapi/ioctl/`
`cd suse10sp1`
`insmod s390ioctl.ko`
	For more information, refer to "Install the Linux I/O module for CKD devices" on page 42.	❑

1. Enable the Solutions Enabler features with the following command: `symlmf add`	For more information, refer to "Licensing your software" on page 62.	❑
2. Build the SYMAPI database by entering the following command: **`symcfg discover`**	For more information, refer to "Building the SYMAPI database" on page 80.	❑
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.	❑

Table 7 UNIX installation cheat sheet

Task	More Information	Done

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 80.	❑

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 80.	❑

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 82.	❑
7. *Optional*: Read the *Solutions Enabler Security Configuration Guide* and apply related security settings.	for more information, refer to *Solutions Enabler Security Configuration Guide*.	❑
8. *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 84.	❑
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 85.	❑
10. *Optional*: Configure the necessary daemons for the environment.	For instructions, refer to: • "Setting up daemons for distributed application support" on page 87 • "Managing the base daemon" on page 92 • "Setting up the event daemon for monitoring" on page 94	❑

CHAPTER 2
Installation

This chapter explains how to install/upgrade Solutions Enabler:

Note: As an alternative to the in-depth UNIX and Windows procedures in this chapter, "Installation cheat sheets" on page 29 provides operating-system-specific cheat sheets 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.

Note: Solutions Enabler V7.5.0 is fully upgradeable. That is, you do not have to remove the previous version before installing V7.5.0.

Note: Before starting this procedure, be sure to review pre-install considerations in Chapter 1 and the *EMC Solutions Enabler Release Notes*.

Note: The default responses to the prompts in this section are in brackets [].

Step 1: Mount the installation DVD

To mount the installation DVD:

1. Log onto the host system as **`root`**.
2. Insert the Solutions Enabler installation DVD into the host's drive.
3. Mount the DVD to a subdirectory (for example, `/dvd`) by entering the appropriate platform-specific `mount` command from Table 8.

Table 8 UNIX mount commands

For	Enter
AIX[a]	`mount -r -v cdrfs /dev/cd0 /dvd`
HP-UX Versions 11.0 and above	`mount -F cdfs /dev/dsk/cxtxdx /dvd`
Linux [b]	**`mount -t iso9660 -o ro /dev/dvd /dvd`**
Solaris	If automounter is running, the disc mounts unattended. To mount the disc manually, enter: `mount -F hsfs -o ro /dev/dsk/cxtxdxs0 /dvd`

a. With AIX, you may get a warning if the device and the directory do not have the same permissions. You can usually ignore these warnings.

b. You should not load Solutions Enabler on the Celerra® File Server Control station, as it is not a Linux client.

Step 2: Run the install script

To run the installation script:

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

   `cd /`*`Install_disc_mount_point`*`/UNIX/`*`operating_system`*

2. Select an installation method from Table 9, and then run the appropriate command. For descriptions of the command options, refer to Table 10 on page 37.

Table 9 Installation method (page 1 of 2)

Method	Command	Comments
Interactive	`./se7500_install.sh -install`	Starts the interactive script documented in the remainder of this chapter. When using this method, continue with "Step 3: Select the installation directories" on page 38.
Silent (all components)	`./se7500_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 41.
	`./se7500_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 41.
	`./se7500_install.sh -install -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 41.
Silent (specific components)	`./se7500_install.sh -install -silent [-nocert] [-all] [-jni] [-srm] [-symrec] [-force] [-64bit] [-daemonuid] [-permission] [-homedir] [-datadir] [-nodeps] [-copy_lic] [-tc]`	Silently installs only the specified components. When using this method, continue with "Step 5: Complete the installation" on page 41.

Table 9 Installation method (page 2 of 2)

Method	Command	Comments
Incremental (specific components)	`./se7500_install.sh -increment [-cert][-jni] [-srm] [-64bit] [-symrec]`	Incrementally adds the specified component to an existing installation. When using this method, continue with "Step 5: Complete the installation" on page 41. 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	`./se7500_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. (Continued on the next page)
Response file (continued from previous page)		Response file entries can be 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 and 64-bit components: 1. Create the following response file: # cat responsefile.txt increment symrec 64bit # 2. Run the command: `./se7500_install.sh -file responsefile.txt` For example, to silently install Solutions Enabler with the Java Interface and SRM components: 1. Create the following response file: # cat responsefile.txt install silent jni srm # 2. Run the command: `./se7500_install.sh -file responsefile.txt` When using this method, continue with "Step 5: Complete the installation" on page 41.

Table 10 defines the various options used when running the installation commands detailed in Table 9 on page 35.

Table 10 UNIX installation options

Option	Description
`-64bit`	Installs the 64 bit libraries. This option is not used in Solutions Enabler V7.2 and higher and is only maintained for backwards compatibility.
`-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.
`-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 command will copy the `symapi_licenses.dat` file from `/tmp` to `/var/symapi/config`: **`bash-3.00# ./se7500_install.sh -install -copy_lic=/tmp -silent`**
`-cert`	Install SSL certificate files.
`-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 *Solutions Enabler Symmetrix CLI Command Reference*.
`-datadir=`*`directory`*	Sets the working root directory [`/usr/emc`]. Used with the `-silent` option.
`-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.
`-jni`	Installs the Solutions Enabler Java Interface component.
`-nocert`	Do not install SSL certificate files.
`-permission=`*`level`*	Sets permission on `/var/symapi` directory. Used with the `-silent` option.
`-silent`	Specifies to perform a silent installation.
`-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).

Note: For help running the installation script, run the following:
`./se7500_install.sh -help`

Note: The installation script creates log files in the directory `/opt/emc/logs`. For more information, refer to Appendix F.

Step 3: Select the installation directories

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."
- If you are upgrading or reinstalling Solutions Enabler, complete "Step 3B: Upgrading /reinstalling" on page 38.

Note: It is recommended that you install EMC Solutions Enabler on your host's internal disks and not on a Symmetrix 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]:
```

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

4. Continue with "Step 4: Select installation options" on page 39.

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]:
```

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:
   storsrvd, storevntd, storgnsd, 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 39.

Step 4: Select installation options

To select your installation options:

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_trustkey.pem in /var/symapi/config/cert. The subject certificate and key files symapisrv_cert.pem and symapisrv_key.pem will also be generated.
   - A [N]o response doesn't install CERT component.

   IMPORTANT

   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 *Solutions Enabler 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 change the default UNIX file permissions. A `[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]:
   ```

8. If you are upgrading, the following prompt displays, asking whether to **move** the previous installation's data files 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]:
   ```

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:

```
./se7500_install.sh -check
```

This command produces the following output in a Linux environment:

```
-bash-2.05b# sh se7500_install.sh -check
#------------------------------------------------------------------------
# EMC Installation Manager
#------------------------------------------------------------------------
Copyright 2012, EMC Corporation
All rights reserved.
The terms of your use of this software are governed by the
applicable contract.
Checking for Solutions Enabler Native Installer kit Installation......
Sl No RPM Version
----- --- --------
1 symcli-base V7.5.0-0
2 symcli-cert V7.5.0-0
3 symcli-data V7.5.0-0
4 symcli-srm V7.5.0-0
5 symcli-symcli V7.5.0-0
6 symcli-symrecover V7.5.0-0
7 symcli-thincore V7.5.0-0
```

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:

```
umount mount_point
```

Install the Linux I/O module for CKD devices

In a Linux on System z installation on Novell SLES 10, you must load a kernel object file in order to issue I/O to Symmetrix arrays by way of CKD devices. In addition, failing to load the object file in an environment where a guest can only see CKD devices will prevent Solutions Enabler from discovering Symmetrix arrays.

To load the kernel object file, locate the operating system-specific object in the directory `/usr/symapi/ioctl/`*`SUSE_Version`*, and then use the `insmode s390ioctl.ko` command to load it.

For example, to load the kernel object in a SLES 10 Service Pack 1 environment, enter:

```
cd /usr/symapi/ioctl/
cd suse10sp1
insmod s390ioctl.ko
```

Enabling the Solutions Enabler components

You must now enable your Solutions Enabler features by entering the appropriate license keys.

Note: For instructions, refer to "Licensing your software" on page 62.

Creating certificate files after initial installation

If the Cert 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.sh create
```

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 44), or a response file (refer to "Using a response file" on page 47).

Note: Solutions Enabler V7.5.0 is fully upgradeable. That is, you do not have to remove the previous version before installing V7.5.0.

Note: Before starting this procedure, review the pre-install considerations in Chapter 1 and the *EMC Solutions Enabler Release Notes*.

Using the InstallShield wizard

To install/upgrade Solutions Enabler using the InstallShield wizard:

1. Save all files and exit all Windows applications.
2. Insert the Solutions Enabler installation disc into the host's disk drive.
3. Change directory to the location of the Solutions Enabler kit by entering the following:

```
cd \Install_disc_mount_point\Windows
```

4. Start the installation program by running the following:

   ```
   se7500-Windows-Processor_type.exe
   ```

   Where *Processor_type* can be x86, x64, or ia64.

   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.

5. In the **InstallShield Wizard for Solutions Enabler Welcome** dialog box, click **Next**.
6. In the **Destination Folder** dialog box, select an installation directory and click **Next**.
7. In the **Setup Type** dialog, select **Typical** to install the default components, select **Complete** to install the full Solutions Enabler product set, or select **Custom** to install a subset of the options. Click **Next** when done.
8. If you selected **Custom**, the **Custom Setup** dialog box opens. Select the options (Table 11) to install, where to install them, and then click **Next**.

Table 11 Windows installation options

Option	Description
BASE_COMPONENT	This option is part of the shared library and runtime environment. It is a corequisite 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. • The control storage libraries, which include features like Snap, device masking, and device monitoring. • The Storage Resource Management base mapping library.
CERT_COMPONENT	installs `ssl.rnd, symapisrv_install.cnf, symapisrv_trust.pem symapisrv_trustkey.pem` in `/var/symapi/config/cert.` The subject certificate and key files `symapisrv_cert.pem` and `symapisrv_key.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.

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 *Solutions Enabler Security Configuration Guide*.

9. 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 87 includes descriptions of the Solutions Enabler daemons.
10. In the **Ready to Install the Program** dialog, click **Install**.
11. In the **InstallShield Wizard Completed** dialog box, click **Finish** to complete the setup, and then go to "Licensing your software" on page 62.

Using the command line

The `se7500-Windows-`*`Processor_Type`*`.exe` is a wrapper for MSI installs. The MSI kit is embedded inside the `executable` and provides more flexibility than the InstallShield installation method.

Double-clicking `se7500-Windows-`*`Processor_Type`*`.exe` extracts the MSI kit to the `temp` folder and runs it from there. In general, the `se7500-Windows-`*`Processor_Type`*`.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:

1. Save all files and exit all Windows applications.
2. Insert the installation disc into the host's disk drive.
3. Change directory to the location of the Solutions Enabler kit by entering the following:

   `cd \`*`Install_disc_mount_point`*`\Windows`
4. Select one of the MSI wrapper script installation options, detailed in the remainder of this section.

Note: In the following command examples, *`Processor_type`* can be x86, x64, or ia64.

Note: Starting with Solutions Enabler V7.3, 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 se7500-Windows-Processor_Type.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 gate way 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.

Non default location

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

```
start /wait se7500-Windows-Processor_Type.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 gate way 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 se7500-Windows-Processor_Type.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.

Add non default features

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

```
start /wait se7500-Windows-Processor_Type.exe /S
/V"ADDLOCAL=JNI_COMPONENT,SRM_COMPONENT /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.

Remove non default features

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

```
start /wait se7500-Windows-Processor_Type.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 se7500-Windows-Processor_Type.exe /S
/V"INSTALLDIR=\"C:\Program Files\Some Folder\
" ADDLOCAL=SRM_COMPONENT /qn"
```

Overwrite mode

To run installer in overwrite mode:

```
start /wait se7500-Windows-Processor_Type.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 se7500-Windows-Processor_Type.exe /S
/V"REINSTALLMODE=VOMUS ADDLOCAL=SRM_COMPONENT /qn"
```

Starting services

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

```
start /wait se7500-Windows-Processor_Type.exe /S /V"ADDLOCAL=ALL
STORAPID=1 STOREVNTD=1 STORSRVD=1 /qn"
```

Where:

`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 se7500-Windows-Processor_type /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.

Example 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
STOREMUD=0
STOREVNTD=0
STORGNSD=0
STORORAD=0
STORRDFD=0
STORSQLD=0
STORSRMD=0
STORSRVD=1
```

```
STORSTPD=0
STORUDBD=0
```

Installing Solutions Enabler on z/OS

This section describes how to install Solutions Enabler on a z/OS mainframe 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 Chapter 1 and the *EMC Solutions Enabler Release Notes*.

Step 1: Copy the files from installation disc

To copy files from the installation disc:

1. Insert the Solutions Enabler installation disc into the disk drive of the Windows host from which you will be running the FTP upload.
2. Copy the file `emc.ssem750.zip` from the `zOS` directory of the install disc to a temporary directory on the Windows desktop.
3. In the temporary directory you just created, 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 filesystem. "Installing the SSL certificates" on page 130 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:

1. Do one of the following:
   - From the TSO READY prompt, enter the following command:

```
RECEIVE INDS('high_level_qualifier.EMC.ssem750.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.ssem750.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:

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 51.

To use SEMJCL:

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 7.5.0 Electronic Kit Install JCL ------.
| Command ===> ____________________________________________                    |
| Press PF3 to Cancel or PF1 for Help                                          |
| Press ENTER to run edit macro SEMX750 which                                  |
| will customize the installation JCL                                          |
|                                                                              |
|     Data Set Name Prefix:  EMC.SSEM750                                       |
|    SMP/E Data Set prefix:  EMC.SMPE                                          |
|         SCF Subsystem Id:  EMC                                               |
|       SCF Linklib Prefix:  EMC.SSCF580                                       |
|           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):            |
| //USERID1A JOB ACCT,'EMC SEM 7.5.0',                                         |
| // CLASS=A,            <-- CHANGE IF NEEDED                                  |
| // MSGCLASS=A,         <-- CHANGE IF NEEDED                                  |
| // NOTIFY=&SYSUID      <-- CHANGE IF NEEDED                                  |
|                                                                              |
|                                                                              |
|                                                                              |
'------------------------------------------------------------------------------'
```

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

Tip 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 EMC 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 USS 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

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(SSEM*vrm*), which delivers the Solutions Enabler for z/OS software.

  Note: If job #05RECEV fails with the message:
  GIM23401T the pProgram 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(SSEMvrm)
  ```

  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

  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:

  1. At the following prompt, enter **Y** to begin the registration process:

     ```
     Do you want to enter a registration key? Y
     ```

  2. At the following prompt, enter the 19-byte key value as specified on the key card:

     ```
     Enter the license key:
     ```

  3. 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.SSEM750.LOADLIB,DISP=SHR
000047 //SYM$LIC   DD  DSN=EMC.SSEM750.LICENSE,DISP=OLD
000048 //SYSPRINT  DD  SYSOUT=*
000049 //SYSOUT    DD  SYSOUT=*
000050 //SYSIN     DD  *
000051  Y
000052  0000-1111-2222-3333
000053  Y
000054  3333-2222-1111-0000
000055  N
000056  /*
```

Note: For more on the new licensing mechanism, refer to "Licensing your software" on page 62. For alternative ways of installing licenses in z/OS, refer to "Installing using alternative methods" on page 72.

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 18.

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`

This job creates and loads the database that is supplied to EMC ControlCenter (or SYMCLI) clients. Job `#10ECCIN` attempts to discover every Symmetrix system connected to your mainframe host. If there are many Symmetrix systems connected, this job may run for a considerable period of time. If there are Symmetrix arrays that you do not want remote clients (such as ControlCenter) to view, you may exclude them from the discover process as follows.

Note: If the configuration of any Symmetrix array attached to a host is changed, then you must re-run job `#10ECCIN` to correctly discover the changed Symmetrix array.

In the `#10ECCIN` job, there is a DD statement named `SYM$AVD`. This is the Symmetrix avoid file and it expects a list of Symmetrix serial numbers. If this list is provided, then information about those Symmetrix systems is not stored in the database.

For example, to avoid Symmetrix `000000000001`, the JCL should look like this:

```
//SYM$AVD DD *
000000000001
/*
```

Note: All 12-digits of the serial number are required.

- `#16CFGCP`

  Copies the sample configuration files to the SYMAPI configuration directory.

Step 6: Complete the installation

Do the following to complete the installation:

1. Perform all other customizing and any testing as required. A sample startup job (`#STORSRV`) for the SYMAPI server daemon can be customized and run on a z/OS system. Note that you can either run `STORSRV` as a batch job or convert it to run as a started task.

   When the tests are successfully completed, continue to the next step.

2. Customize and run job `#11ACCPT`. This job accepts the function management ID `SSEM`*`vrm`* 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 137 for further details.

   Note: For Solutions Enabler releases prior to V7.4, control functions had to be specifically enabled. If you are using Solution Enabler release prior to V7.4, refer to appropricate documentation on how to enable these control functions.

Your Solutions Enabler installation is now complete. Next, you need to establish your server environment by performing the configuration and setup procedures explained in Chapter 5.

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 "Installing the SSL certificates" on page 130.

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, be sure to review the pre-install considerations in Chapter 1 and the *EMC Solutions Enabler Release Notes*.

Step 1: Access the software

Solutions Enabler is distributed in the following forms:

- On the Solutions Enabler installation disc, which includes kits for all supported platforms.
- As a platform-specific file download from EMC online help at:

  https://support.EMC.com

  Possible filenames are:

`SE750RT.SAV`	HP Alpha hardware platform.
`SE750RIA.SAV`	HP Integrity hardware platform.

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

From an install disc

To access the software from an installation disc:

1. Insert the Solutions Enabler disc into the host's disk drive.
2. Mount the disc by entering:

```
mount/media=cdrom/undefined=(fixed:cr:32256)/over=id dkd600
```

3. Copy the kit from the Solutions Enabler directory `[other.ovms]` to a temporary directory on your machine by entering:

```
copy /log dkd600:[other.ovms]InstallKit sys$sysdevice:[EMC.KITS]*.*
```

From EMC Online Support

To access the software from EMC online help:

1. On EMC Online Support, select **Support › Software Downloads and Licensing › Downloads S › Solutions Enabler** and click the platform-specific installation kit.
2. 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:

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 5 on page 57 describes the privileges these users require.

   Note: For these users to execute `sym*` commands, the base daemon (**storapid**) must be running. Currently, `sym*` commands executed through **storapid** can run slower than without the **storapid**.

   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_start_storsrvd.com` can be placed in SYSTARTUP_VMS.COM file to start storsrvd on system startup. In a cluster environment, this should be done for only one machine in the cluster since this release only supports one storsrvd process running in the cluster.
   - `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. This allows you to run separate daemons on each of the machines 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 are deleted. Even though they remain intact they are not used by the just installed software.

4. Ensure that each SYMCLI user's login procedure calls the `emc_cli.com` procedure to establish their proper SYMCLI environment.
5. 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.
6. Set the following minimum process quotas for each user account:

   FILLM:1000

   BIOLM:300

   DIOLM:300

   ASTLM:500

ENQLM:4000

BYTLM:500000

WSEXTENT:32768

7. 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 revaluate 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 Symmetrix array.

   SN = Number of locally attached Symmetrix arrays.

   RN = Number of remotely attached Symmetrix 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.

8. Once you have calculated the changes to the quotas for your configuration, edit the file daemon_start_template.com to reflect these new values. This file is located in `emc$root:[emc.symcli.bin]`.

9. The installation is complete. Go to "Licensing your software" on page 62.

Installing Solutions Enabler on Solaris 11 Local Zones

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.

1. Install SUNWpkgcmds using below command:

   ```
   pkg install SUNWpkgcmds
   ```

2. Install Solutions Enabler using install script *se7500_install.sh* or native package installation commands:

   ```
   ./se7500_install.sh -install
   ```

3. The installation is complete. Go to "Licensing your software" on page 62.

CHAPTER 3
Post-Install for UNIX, Windows, and OpenVMS

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, and OpenVMS environments:

Note: As an alternative to the in-depth UNIX and Windows procedures in this chapter, "Installation cheat sheets" on page 29 provides operating-system-specific cheat sheets with high-level installation and configuration steps that advanced users may find useful.

Licensing your software

Starting with the release of Enginuity 5875, Solutions Enabler introduced support for Electronic Licensing (eLicensing). eLicensing is an end-to-end license management solution to help you track and comply with software license entitlement. eLicensing leverages embedded locking functions and back-office IT systems and processes. It provides you with better visibility into software assets, easier upgrade, and capacity planning and reduced risk of non-compliance, while still adhering to a strict "do no harm" policy to your operations. This ensures that when upgrades are performed from a VMAX family array running Enginuity versions lower than 5875 to an array running Enginuity 5875 or higher, the VMAX family array is scanned for Enginuity features currently in use that require eLicenses. If Enginuity features are found in use, and there are no eLicenses registered and applied to support their use, they are internally reported as "IN USE," which allows continued access to the Enginuity 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 array and business. If your eLicensing report does display one or more Enginuity features as "IN USE," it is your responsibility to work with your EMC Sales team to obtain proper eLicensing for those features.

With the introduction of eLicensing, Symmetrix licensing moved from a host-based model to a Symmetrix-based model, with the majority of licenses now being stored internally on the Symmetrix array.

When installing licenses with eLicensing, 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 Symmetrix arrays. Each license file fully defines all of the entitlements for a specific array, including the type of license (Individual or Enterprise), the licensed capacity, and the date the license was created. If you want to add a product title or increase the licensed capacity of an entitlement, you must obtain a new license file from EMC Online Support and push it out to the Symmetrix array.

When managing your licenses, Solutions Enabler, Unisphere for VMAX, EMC z/OS Storage Manager (EzSM), MF SCF native command line, TPF, and IBM i platform console, allow you to view detailed usage reports so that you can better manage your capacity and compliance planning.

Note: For more information on eLicensing, refer to EMC Knowledgebase article EMC251709 on EMC Online Support.

Licenses

Most Symmetrix licenses use the Symmetrix-based model. However, there are still a number of Symmetrix licenses that remain host-based. In addition, there are a number of retired host-based licenses.

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

Symmetrix-based licenses

With the release of Symmetrix 40K, Solutions Enabler is introducing support for Symmetrix-based license bundles. A license bundle is a single license that enables multiple features. For example, the Symmetrix Remote Replication Suite license bundle enables the SRDF, SRDF/A and SRDF/S features. For Symmetrix 10K and 20K arrays, Solutions Enabler continues to support individual Symmetrix-based licenses.

Table 12 on page 63 lists the Symmetrix-based licenses supported with Symmetrix family arrays.

Table 12 Symmetrix-based licenses supported with Symmetrix family arrays (page 1 of 3)

License/Description		Allows you to	With the command[a]
Enginuity 40K	**Enginuity 10K and 20K[b]**		
SYMM_VMAX_Enginuity License for whole array Includes: - Dynamic Cache Partitioning - Symmetrix Priority Controls - Symmetrix Optimizer	SYMM_*Model*_ENGINUITY License for whole array	Virtualize an eDisk for encapsulation	symconfigure
		Use VLUN to migrate from an encapsulated device (use it as a source device)	
		Use an encapsulated device as a clone source	
	SYMM_*Model*_DCP[d] Dynamic Cache Partitioning	Enable cache partitions for a Symmetrix array	symqos -cp
		Create cache partitions	
		Set cache partitions to Analyze mode	
	SYMM_*Model*_SPC[d] Symmetrix Priority Controls	Enable priority of service for a Symmetrix array	symqos -pst
		Set host I/O priority	
		Set copy QoS priority	
	SYMM_*Model*_OPTIMIZER[d] Symmetrix Optimizer	Enable Optimizer functionality, including: • Manual mode • Rollback mode • Manual Migration mode	symoptmz
		Schedule manual swaps	
		Set the following Optimizer-specific parameters: • Device Swap Priority • Any of the Optimizer Advanced parameters	
		Set the following Optimizer/FAST parameters: • User Approval Mode • Maximum Devices to Move • Maximum Simultaneous Devices • Workload Period • Minimum Performance Period	
		Validate or create VLUN migrations	symmigrate
		Create time window	symoptmz symtw

Table 12 Symmetrix-based licenses supported with Symmetrix family arrays (page 2 of 3)

License/Description		Allows you to	With the command[a]
Enginuity 40K	**Enginuity 10K and 20K[b]**		
SYMM_VMAX_SRDF_REPLICATION Symmetrix Remote Replication Suite Includes: - SRDF - SRDF/Asynchronous mode - SRDF/Synchronous mode	SYMM_*Model*_SRDF[c, d] SRDF	Create new RDF groups	symrdf
		Create dynamic RDF pairs in Adaptive Copy mode	
		Create RDF devices	symconfigure
		Convert non-RDF devices to RDF	
		Add RDF mirrors to devices in Adaptive Copy mode	
		Set the dynamic-RDF capable attribute on devices	
		Create SAVE devices	
	SYMM_*Model*_SRDF_A[d] SRDF/Asynchronous mode	Create dynamic RDF pairs in Asynchronous mode	symrdf
		Set RDF pairs into Asynchronous mode	
		Add RDF mirrors to devices in Asynchronous mode	symconfigure
		Create RDFA_DSE pools	
		Set any of the following SRDF/A attributes on an RDF group: • Minimum Cycle Time • Transmit Idle • DSE attributes, including: - Associating an RDFA-DSE pool with an RDF group - DSE Threshold - DSE Autostart • Write Pacing attributes, including: - Write Pacing Threshold - Write Pacing Autostart - Device Write Pacing exemption - TF Write Pacing Autostart	
	SYMM_*Model*_SRDF_S[d] SRDF/Synchronous mode	Create dynamic RDF pairs in Synchronous mode	symrdf
		Set SRDF pairs into Synchronous mode	
		Add an RDF mirror to a device in Synchronous mode	symconfigure
SYMM_VMAX_SRDF_STAR SRDF/Star	SYMM_*Model*_SRDF_STAR[d] SRDF/Star	Perform a setup to initialize the environment	symstar[e]

Table 12 Symmetrix-based licenses supported with Symmetrix family arrays (page 3 of 3)

License/Description		Allows you to	With the command[a]
Enginuity 40K	**Enginuity 10K and 20K[b]**		
SYMMETRIX_VMAX_TIMEFINDER Symmetrix TimeFinder Suite Includes: - TimeFinder/Clone - TimeFinder/Snap	SYMM_*Model*_TF_CLONE[d] TimeFinder/Clone	Create new native clone sessions	symclone
		Create new TimeFinder/Clone emulations	symmir
	SYMM_*Model*_TF_SNAP[d] TimeFinder/Snap	Create new sessions	symsnap
		Duplicate existing sessions	
		Create snap pools	symconfigure
		Create SAVE devices	
SYMM_VMAX_FAST_TIERING Symmetrix Tiering Suite Includes: - FAST for disk groups - FAST for virtual pools	SYMM_*Model*_FAST[d] FAST for disk groups	Create time windows	symoptmz symtw
		Add disk group tiers to FAST policies	symfast
		Enable FAST	
		Set the following Optimizer/FAST parameters: • Swap Non-Visible Devices • Allow Only Swap • User Approval Mode • Maximum Devices to Move • Maximum Simultaneous Devices • Workload Period • Minimum Performance Period	
	SYMM_*Model*_FAST_VP FAST for virtual pools	Create time windows	symoptmz symtw
		Add virtual pool (VP) tiers to FAST policies	symfast
		Enable FAST	
		Set the following FAST VP-specific parameters: • Thin Data Move Mode • Thin Relocation Rate • Pool Reservation Capacity	
		Set the following Optimizer/FAST parameters: • Workload Period • Minimum Performance Period	
SYMM_VMAX_OR_DM RCOPY	SYMM_*Model*_OR_DM[d] RCOPY	Create hot push sessions	symrcopy
		Create cold pull sessions	
		Create cold push sessions	
SYMM_VMAX_SMC - Symmetrix Management Console - Unisphere for VMAX	SYMM_*Model*_SMC - Symmetrix Management Console - Unisphere for VMAX	Manage Symmetrix arrays running Enginuity 5875 Q22011 SR or higher.[f]	N/A

a. For complete command syntax, refer to the *Solutions Enabler Symmetrix CLI Command Reference*.

b. In the license name, *Model* indicates the Symmetrix model on which the license is installed. Possible values are: VMAX or VMAXE.

c. Requires that the Symmetrix array also be licensed for SRDF/A (SYMM_*Model*_SRDF_A) and/or SRDF/S (SYMM_*Model*_SRDF_S).

d. This feature requires a host-based license when the Symmetrix array being managed is running an Enginuity version lower than 5875. See Table 14 on page 66.

e. Requires either SYMM_*Model*_SRDF_A, SYMM_*Model*_SRDF_S or SYMM_VMAX_SRDF_REPLICATION licenses.

f. This license is not required to manage Symmetrix arrays running an Enginuity version lower than 5875.198.148 from a host running SMC V7.3 or higher or Unisphere V1.0 or higher.

Host-based licenses

Table 13 lists the host-based licenses that remain unchanged, regardless of Enginuity level.

Table 13 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 14 lists the host-based licenses required to perform operations on Symmetrix arrays running Enginuity versions lower than 5875 from a Solutions Enabler V7.5 host.

Table 14 Host-based licenses required for Enginuity versions lower than 5875

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

Retired licenses

The following licenses are retired. The features that required these licenses still exist, they just no longer require licenses.

- Base
- Cache
- Configuration Manager
- Config Mgr - Create VDEVs (Snap Configure)
- Delta Mark
- Device Masking
- IPsec
- Open Replicator/LM
- Secure Erase
- SRDF/Automated Replication
- SRDF/Cascading RDF
- SRDF/Consistency Groups
- SRM_BASE
- SRM_Enabler
- SRM_FULL
- SYMAPI Server
- TimeFinder/Consistency Groups
- TimeFinder/Exchange Integration Module
- TimeFinder/SQL Integration Module
- Virtual Provisioning
- Worm

Managing Symmetrix 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 Symmetrix-based licenses in the array's feature registration database (Enginuity 5875 or higher).

Note: The location of this `symapi_licenses.dat` file varies according to the operating system. For more information, refer to Appendix E.

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 V7.2 (or higher) host

When accessing a Symmetrix array running Enginuity 5875 or higher from a host running Solutions Enabler V7.2 or higher, the operations you can perform on the array are based on:

- The licenses in the array's feature registration database (Table 12 on page 63).
- The licenses in the host's `symapi_licenses.dat` file, if using any of the host-based features listed in Table 13 on page 66.

When accessing a Symmetrix array running an Enginuity version lower than 5875 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 13 on page 66 and Table 14 on page 66. If not, you can only perform operations that do not require a license (see "Retired licenses" on page 67).

When accessing a Symmetrix array upgraded from Enginuity 5874 to Enginuity 5875 or higher from a host upgraded to Solutions Enabler V7.2 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 5875 product titles or any of the older product titles you were not using, you must obtain and install a Symmetrix-based license file on the array. "Installing Symmetrix-based licenses" on page 71 describes how to install license files.

Solutions Enabler Pre-7.2 host

When accessing a Symmetrix array running Enginuity 5875 or lower, from a host running Solutions Enabler V7.1 through V7.1.2, the operations you can perform are based on the licenses in the host's `symapi_licenses.dat` file.

Capacity measurements

Symmetrix-based licenses include a *capacity licensed* value that defines the scope of the license. The method for measuring this value depends on the license's *capacity type* (Raw, Registered, or External).

Not all product titles are available in all capacity types, as shown in Table 15.

Table 15 Product title capacity types

Raw only	Raw or Registered	External
Enginuity	SRDF/Asynchronous mode[a]	SRDF/Asynchronous mode[a]
Dynamic Cache Partitioning	SRDF/Synchronous mode[a]	SRDF/Synchronous mode[a]
Symmetrix Optimizer	SRDF/Star	SRDF/Star
Symmetrix Priority Controls	Synthesized SRDF[b]	TimeFinder/Clone[c]
Unisphere for VMAX	TimeFinder/Clone[c]	TimeFinder/Snap[c]
	TimeFinder/Snap[c]	FAST for virtual pools[d]
	FAST for disk groups[d]	
	FAST for virtual pools[d]	
	RCOPY	

a. With Enginuity 5876 or higher, this license is enabled by the Symmetrix Remote Replication Suite.

b. Created from SRDF/A and SRDF/S entitlements in the license file.

c. With Enginuity 5876 or higher, this license is enabled by the Symmetrix TimeFinder Suite.

d. With Enginuity 5876 or higher, this license is enabled by the Symmetrix Tiering Suite.

Raw capacity

Raw capacity is the sum of the rated capacity of all disks of type SATA or Non-SATA in the array (in TB, where 1kB = 1000 bytes), excluding spares for that type.

The type of raw capacity disks can be the SATA disks in the array, or the Non-SATA disks in the array (both types can and do appear in the license file).

Registered capacity

Registered capacity is the amount of user data that will be managed or protected by each particular product title. It is independent of the type or size of the disks in the array.

The methods for measuring registered capacity depends on whether the licenses are part of a bundle or individual.

Registered capacity for license bundles

For license bundles, registered capacity is measured according to the following:

- Symmetrix Tiering Suite:
  - The registered capacity for this bundle is measured as the sum of the registered capacity of all devices associated with all FAST_VP policies and the registered capacity of all devices associated with all FAST policies not associated with a FAST_VP policy.
  - For virtually provisioned devices, the registered capacity is equal to the total space allocated to the thin device. For devices that have compressed allocations, the un-compressed size is used.
  - For disk group provisioned devices, the registered capacity is equal to the total size of the device.
- Symmetrix Remote Replication Suite:
  - The registered capacity for this bundle is measured by the amount of data that can be stored in all forms of RDF devices (R1s, R2s, and R21s) on a Symmetrix array.
    - Concurrent SRDF sources are counted only once when measuring registered capacity usage.
    - In the case of diskless RDF (Extended Data Protection), no registered capacity is reported as used.
  - For virtually provisioned devices, the registered capacity is equal to the total space allocated to the thin device. For devices that have compressed allocations, the un-compressed size is used.
  - For disk group provisioned devices, the registered capacity is equal to the total size of the device.
- Symmetrix TimeFinder Suite:
  - The registered capacity of this bundle is measured as the sum of the capacity of a device if it is a clone source or target, a snap source, or SAVE device in a snap pool. Regardless of whether a device meets two or more of the following criteria, it is only counted once.
  - For virtually provisioned devices, the registered capacity is equal to the total space allocated to the thin device. For devices that have compressed allocations, the un-compressed size is used.
  - For disk group provisioned devices, the registered capacity is equal to the total size of the device.

Registered capacity for individual licenses

For individual licenses, registered capacity is measured according to the following:

- FAST registered capacity is measured as the sum of the registered capacity of all devices that are associated with all FAST policies that contain disk group tiers.
- FAST VP registered capacity is measured as the registered capacity of all devices that are associated with all FAST policies that contain virtual pool tiers.
- SRDF registered capacity is measured as the configured size of all forms of RDF standard devices (R1s, R2s, and R21s). In the case of thin devices, it is the size of allocated tracks associated with all forms of RDF devices (R1s, R2s, and R21s).
  - Concurrent SRDF sources are counted only once when measuring registered capacity usage.
  - In the case of diskless RDF (Extended Data Protection), no registered capacity is reported as used.
- SRDF/Star registered capacity is measured as the configured size of all standard devices participating in Star configurations. In the case of thin devices, it is the size of allocated tracks on devices associated with Star configurations.
- TimeFinder/Clone registered capacity is measured as the configured size of all standard devices that are clone sources or clone targets. In the case of thin devices, it is the size of allocated tracks on devices that are clone sources or clone targets.

  The registered capacity of a device that is both a clone target and a clone source for another is counted once.
- TF/Snap registered capacity is measured as the configured size of all save pools, plus the configured size of all standard devices that are snap source devices. In the case of thin devices, it is the size of allocated tracks on devices that are snap source devices.

The following devices are not counted in registered capacity:

- SAVE devices not in any pool
- DATA devices not in any pool
- Devices not associated with RDF or TimeFinder

Registered capacity is reported in a tenth of a terabyte format (e.g., 42.3 TB) and rounded up or down to the nearest GB. For example, 42.31 TB and 42.25 TB will round to 42.3 TB.

External capacity

External capacity is measured by the sum of the sizes of virtualized LUNs from external storage.

If the entitlement is licensed for registered capacity, any external usage will be added together with the internal usage.

Installing Symmetrix-based licenses

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

Note: Installing licenses requires a Symmetrix authorization role of Storage Admin or higher.

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

- Windows: x86, AMD64, ia64
- Linux: x86, AMD64, ia64
- Solaris: 32 bit (Sparc), 64 bit (Sparc)
- HPUX 11.11: PA-Risc 32, PA-Risc 64
- HP-UX 11.21: ia64
- AIX 5.3 and 6.1: PPC 32, PPC 64

For instructions on installing from a host running a supported operating system, refer to "Installing from a supported host" on page 71. For instructions on installing from a host running a non-supported operating system, refer to "Installing using alternative methods" on page 72.

Note: To obtain Symmetrix-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 a Symmetrix-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 Symmetrix array:

   `symlmf add -type emclm -sid` *`SymmID`* `-file` *`FileName`* `-v`

   Where:

   `SymmID` — Specifies the Symmetrix array on which you are installing the license file.

   `FileName` — Specifies the name of the license file.

   Output similar to the following displays:

```
License SYMM_VMAX_SPC 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_DCP 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_FAST_VP 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_FAST 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_OPTIMIZER 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_TF_SNAP 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_TF_CLONE 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_SRDF_STAR 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_SRDF_S 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_SRDF_A 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_SRDF 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_ENGINUITY 000000001234 15-Jan-2011: Processed successfully
License SYMM_VMAX_OR-DM 000000001234 15-Jan-2011: Processed successfully
```

```
License SYMM_VMAX_SMC 000000001234 15-Jan-2011: 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 a Symmetrix-based license file from a host running a non-supported operating system, use one of the following methods:

- Run `symlmf` directly on the Symmetrix 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 a Symmetrix authorization role of Storage Admin or higher.

To install a host-based license:

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 76.

Displaying Symmetrix-based licenses

To display the current Symmetrix-based licenses activated by a license file, use the following command:

```
symlmf list -type emclm -sid SymmID
```

Output similar to the following displays:

```
Symmetrix ID : 000000001234
Issue Date   : 08/22/2012

                              Activation        Capacity            Install
Feature Name                  Type      ID      Type          Licensed    Date
----------------------------- ----- ---------- ------------- -------- ----------
SYMM_VMAX_ENGINUITY           P-IND 111111111  R-TB-Non-SATA      100 09/13/2012
                                               R-TB-SATA          500
SYMM_VMAX_FAST_TIERING        P-IND 1234567    Reg-TB              60 09/13/2012
SYMM_VMAX_OR_DM               P-IND 1234567    R-TB-Non-SATA      100 09/13/2012
                                               R-TB-SATA          500
                                               R-TB-EXTERNAL      300
SYMM_VMAX_SMC                 P-IND 1234567    R-TB-Non-SATA      100 09/13/2012
                                               R-TB-SATA          500
                                               R-TB-EXTERNAL      300
SYMM_VMAX_SRDF_REPLICATION    P-IND 1234567    Reg-TB              10 09/13/2012
SYMM_VMAX_SRDF_STAR           P-IND 1234567    Reg-TB              20 09/13/2012
SYMM_VMAX_TIMEFINDER          P-IND 1234567    Reg-TB              80 09/13/2012

Legend:
  Activation Type:
    E-IND = Evaluation Individual
    P-IND = Permanent Individual
    P-ENT = Permanent Enterprise Agreement
    P-LTD = Permanent Limited
```

If individual licenses had been purchased, output similar to the following displays:

```
Symmetrix ID : 000194901138
Issue Date   : 10/05/2011

                              Activation        Capacity            Install
Feature Name                  Type      ID      Type          Licensed    Date
--------------------------- ----- ---------- ------------- -------- ----------
SYMM_VMAX_ENGINUITY         P-IND 102938475  R-TB-Non-SATA      100 08/22/2012
                                             R-TB-SATA          500
SYMM_VMAX_FAST              P-IND 1234567    Reg-TB              60 08/22/2012
SYMM_VMAX_OR_DM             P-IND 1234567    Reg-TB              10 08/22/2012
SYMM_VMAX_PROSPHERE         P-IND 1234567    R-TB-Non-SATA      100 08/22/2012
                                             R-TB-SATA          500
SYMM_VMAX_SMC               P-IND 1234567    R-TB-Non-SATA      100 08/22/2012
                                             R-TB-SATA          500
SYMM_VMAX_SRDF              P-IND 1234567    Reg-TB              30 08/22/2012
SYMM_VMAX_SRDF_S            P-IND 1234567    Reg-TB              20 08/22/2012
SYMM_VMAX_SRDF_STAR         P-IND 1234567    Reg-TB              40 08/22/2012
SYMM_VMAX_TF_CLONE          P-IND 1234567    Reg-TB              50 08/22/2012

Legend:
  Activation Type:
    E-IND = Evaluation Individual
    P-IND = Permanent Individual
    P-ENT = Permanent Enterprise Agreement
    P-LTD = Permanent Limited
```

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-based licenses

To display host-based licenses, use the following commmand:

symlmf list -type host

Output similar to the following displays:

```
Host ID:  host1234

                                      Days
                                      Until    Capacity
Feature Name            SymmID        Expr   Type  Units
----------------------  ------------  -----  ----  -----
OraclePak               -                 -     -      -
SPA_BASE                000000001234      -  R-TB  1000


Legend:

  Capacity:
    R-TB    = Raw capacity in TB
    REG-TB  = Configured capacity in TB
    -       = Not applicable
```

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 host -output xml_element

Displaying host and Symmetrix-based licenses

To display the host-based and Symmetrix-based licenses that apply to Symmetrix array, use the following command:

```
symlmf list -type sym -sid 1234
```

Output similar to the following displays:

```
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) Symmetrix arrays
  - individual Symmetrix arrays but the feature is **L**(imi)**T**(e)**D**
  - individual Symmetrix arrays but with a limited **Eval**(uation) time period
  - or to all the Symmetrix arrays in the **Ent**(erprise)

- **Capacity Licensed**: The maximum quantity of data which the functionality of the software is licensed to use, in Terabytes.
- **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 Symmetrix array.
  - **R-TB External**: Indicates that the capacity licensed applies to the raw capacity of the virtualized LUNs in external storage.
- **Capacity Units**: The maximum quantity of data for which the functionality of the software is licensed to use, in Terabytes.
- **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 Symmetrix-based (**EMCLM**).
- **SymmID**: The Symmetrix 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 Symmetrix array.

For example, to display the state and usage number for all activated licenses on Symmetrix 1234, enter the following:

```
symlmf query -type emclm -sid 1234
```

Output similar to the following displays:

```
Symmetrix ID : 000000001234
Issue Date   : 09/10/2012

                                                Capacity
                                    --------------------------------
Feature Name                  Act Type          Licensed Usage
----------------------------- --- ------------- -------- --------
SYMM_VMAX_ENGINUITY           ENT R-TB-Non-SATA      100     40.4
                                  R-TB-SATA          500    260.0
SYMM_VMAX_FAST_TIERING        ENT R-TB-Non-SATA      100     40.4
                                  R-TB-SATA          500    260.0
                                  R-TB-EXTERNAL      300      0.0
SYMM_VMAX_OR_DM               ENT R-TB-Non-SATA      100     40.4
                                  R-TB-SATA          500    260.0
                                  R-TB-EXTERNAL      300      0.0
SYMM_VMAX_PROSPHERE           ENT R-TB-Non-SATA      100     40.4
                                  R-TB-SATA          500    260.0
                                  R-TB-EXTERNAL      300      0.0
SYMM_VMAX_SMC                 ENT R-TB-Non-SATA      100     40.4
                                  R-TB-SATA          500    260.0
                                  R-TB-EXTERNAL      300      0.0
SYMM_VMAX_SRDF_REPLICATION    ENT R-TB-Non-SATA      100     40.4
                                  R-TB-SATA          500    260.0
                                  R-TB-EXTERNAL      600      0.0
SYMM_VMAX_SRDF_STAR           ENT R-TB-Non-SATA      100     40.4
                                  R-TB-SATA          500    260.0
                                  R-TB-EXTERNAL      300      0.0
SYMM_VMAX_TIMEFINDER          ENT R-TB-Non-SATA      100     40.4
                                  R-TB-SATA          500    260.0
                                  R-TB-EXTERNAL      300      0.0

Legend:
  Act(ivation Type):
    ENT = Entitlement
    MAN = Manual Override
    USE = In Use
```

If individual licenses had been purchased, output similar to the following displays:

```
Symmetrix ID : 000000001234
Issue Date   : 08/22/2012

                                          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
    MAN = Manual Override
    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.
  - **MAN**: Indicates that the product title was manually activated by EMC.
  - **USE**: Indicates that the product title is activated because it was in use prior to upgrading from Enginuity 5874 to Enginuity 5875. 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 activated manually (MAN) or because they were 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 Symmetrix array.
  - **R-TB External**: Indicates that the capacity licensed applies to the raw capacity of the virtualized LUNs in external storage.

- **Capacity Licensed**: The maximum quantity of data which the functionality of the software is licensed to use, in Terabytes.
- **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 13 on page 66 and Table 14 on page 66.

Note: You cannot delete Symmetrix-based licenses.

Initial steps for post-install 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 Symmetrix configuration.

Note: To include information on CLARiiON arrays in the SYMAPI database, you must perform an assisted discovery. For more information on building the SYMAPI database, refer to the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*.

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 81 and "Setting the online help path" on page 82.

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 Symmetrix CLI Command Reference*.

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 *Solutions Enabler 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 153.

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, do the following 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.

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 Symmetrix array by way of gatekeepers or the configuration database.

Note: CLARiiON storage systems do not use gatekeepers.

Semaphore requirements on UNIX

Starting with Solutions Enabler V7.3.x, you no longer have to modify semaphore settings on the host when using its default configuration (default options). However, some settings (for example, in the `daemon_options` file. See "Setting the optional base daemon behavior parameters" on page 93 for more information.) 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 Symmetrix 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.

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 Appendix D.

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 and database locking may fail. For more information about adjusting semaphore parameters for your operating system, refer to Appendix D.

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
```

OpenVMS locking

On OpenVMS, SYMCLI uses the Distributed Lock Manager to accomplish locking. These locks are automatically de-allocated from the system when the last process, which has opened the lock, finishes or is terminated. There is no kernel configuration requirement. The lock name is derived from the gatekeeper or database pathname.

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 or database 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 Symmetrix IDs you can use to limit SYMCLI or SYMAPI from seeing certain Symmetrix arrays, devices, or gatekeepers which would otherwise be affected by various commands.

The files hold either physical device names (*PdevNames*) or Symmetrix IDs (*Symmids*) with line entries having only one device name or ID per line. Lines beginning with a "#" (comment) are ignored by SYMCLI.

1. The location of this directory varies according to the operating system. For more information, refer to Appendix E.

gkavoid and gkselect

The `gkavoid` and `gkselect` files affect calls to various online SYMCLI commands that use a gatekeeper to communicate with a Symmetrix array.

Note: For more information on using these files, refer to "Using the gkavoid and gkselect files" on page 151.

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 Symmetrix devices from your host. The inquiry file is formatted with physical (host) device names with one PdevName per line.

Table 16 provides platform specific PdevName examples.

Table 16 PdevName examples

Operating system	Example Pdevname
UNIX	`/dev/rdsk/c2t0d2s2`
Windows	`\\.\PHYSICALDRIVE1`
OpenVMS	`$1$DGA6401:`
z/OS	`VOL001`

Note: For more information on PdevNames, refer to the *EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*.

symavoid

The `symavoid` file affects the operation of `symcfg discover` so that it does not look for devices that belong to the Symmetrix arrays specified in this file. This may be useful if there are multiple Symmetrix arrays connected to the host that you want SYMCLI to avoid. The Symmetrix avoidance file is formatted with 12-character Symmetrix IDs with one ID per line.

To obtain a list of Symmetrix 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 Symmetrix environment inoperative.

The `options` file must be created and placed in the SYMAPI configuration directory.[1]

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 Symmetrix CLI Command Reference.*

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:

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:

1. The location of this directory varies according to the operating system. For more information, refer to Appendix E.

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.

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 Symmetrix storage resource. Applications do not have to run as a privileged user.

The base daemon (storapid) coordinates all Symmetrix locks and parallel application syscalls to your operating system kernel, which optimizes their operations (such as TimeFinder-type actions).

For storage resource management (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 Symmetrix 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 17 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 17 Daemon support matrix

Daemon name	Platforms supported	Description	Daemon-specific parameter documentation
storapid	UNIX[a], Win32, z/OS, AS400, BS2000, Open VMS	Base daemon	Refer to "Managing the base daemon" on page 92 in this guide.
storgnsd	UNIX, Win32	Group Name Services (GNS) daemon	*EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*
storrdfd	UNIX, Win32	RDF daemon	*EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide*
storevntd	UNIX, Win32, z/OS	Event daemon	Refer to "Setting up the event daemon for monitoring" on page 94 in this guide.
storsrvd	UNIX, Win32, z/OS, AS400, BS2000, OpenVMS	SYMAPI Server daemon (executes remote Solutions Enabler API functions)	Refer to Chapter 4 in this guide.
storwatchd	UNIX, BS2000, Open VMS	UNIX only: Watchdog daemon	*EMC Solutions Enabler Symmetrix Array Management CLI Product Guide*
storsrmd storsrmd64	Solaris, AIX, HP-UX, Windows	SRM daemon	*EMC Solutions Enabler Symmetrix Storage Resource Management CLI Product Guide*
storstpd	UNIX, Windows	Statistics (STP) daemon	
stororad		SRM daemon for Oracle DB	
storora64d		SRM daemon for Oracle DB (64-bit)	
storudbd		SRM daemon for UDB DB	
storsqld		SRM daemon for SQL DB	
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)	

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 storoorad -wait 5
```

In an OVMS cluster, a daemon can be started on any member of the cluster as long as the DCL command procedure `emc_install_sys_specific.com` has been executed on the member machine. For example, this allows a base daemon to be started on each member of the cluster.

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
```

To undo this, enter the following:

```
stordaemon uninstall DaemonName
```

Authorizing daemon connections

Typically, daemons run with root/administrator privileges,[1] 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 `adb` and check the per-process file table, the open files 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.

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.[2]

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.

1. Starting with Solutions Enabler V7.1, some daemons can run as non-root on UNIX systems.
2. 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.[1]

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 18). 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`).

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.

1. The location of this directory varies according to the operating system. For more information, refer to Appendix E.

Table 18 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 18 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 94, and for options specific to the SYMAPI server daemon, refer to "Specifying server behavior" on page 121.

Managing the base daemon

The base daemon (`storapid`) provides centralized gatekeeper device management for all Solutions Enabler applications requiring access to Symmetrix 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.

Each host running an instance of the RDF daemon (`storrdfd`) must also run the base daemon, as it requires the use of the gatekeeper management services.

Note: For more on gatekeepers, refer to Chapter 6.

Starting the base daemon

By default, the base daemon will automatically start the first time a Solutions Enabler application attempts to access a Symmetrix 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 89.

- Set the base daemon to automatically start every time the local host is booted using the following command:

  ```
  stordaemon install storapid -autostart
  ```

  Note: Starting with Solutions Enabler V7.1, `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]
```

Applying the -`all` option 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 19 lists some of the possible optional base daemon parameters.

Table 19 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 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.

  The Unisphere for VMAX, SMI Provider and Control Center 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 98) 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, i.e., 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 Symmetrix 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 Symmetrix Audit log into events.

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 a Symmetrix 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 Symmetrix device number as a 4-digit hexadecimal number, for example, 0007 or 0123.
  - A Symmetrix disk ID using the standard Solutions Enabler syntax, for example, 16B:C2.
  - A Symmetrix director ID using the standard Solutions Enabler syntax, for example, FA-3B.
  - A port on a Symmetrix 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 severitywhen 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 89.

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:

1. Run the following command to load the Symmetrix 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 89.

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 97.

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" above:

- If you specified to log events by way of SNMP (`snmp` option), complete "Step 2A: Configure an SNMP event target" on page 99.
- If you specified to log events in a log file (`file` option), continue with "Step 2B: Configure a log file" on page 100.
- 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 101.
- 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 101.

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 Symmetrix 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 20 maps the event daemon severity level to the SNMP severity levels, as specified in the FC-management MIB.

Table 20 Event daemon severity level/SNMP severity level mappings (page 1 of 2)

Event daemon severity	SNMP trap severity
fatal	2 (Emergency)
critical	4 (Critical)
major	5 (Error)
minor	5 (Error)

Table 20 Event daemon severity level/SNMP severity level mappings (page 2 of 2)

Event daemon severity	SNMP trap severity
warning	6 (Warning)
info	8 (Info)
normal	8 (Info)
--	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
```

Step 2B: Configure a log file

The daemon_options file contains parameters (Table 21) 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 21 Event log file configuration options (page 1 of 2)

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.

Table 21 Event log file configuration options (continued) (page 2 of 2)

Parameter	= <OptValue | defaultvalue>	Description
storevntd:log_event_file_size	> 0 - *nn* | 1	When used with the log_event_file_type parameter set to wrap, this parameter specifies the maximum file size (in KB) allowed before wrapping to the alternate file. This value should be a decimal number greater than zero.
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 22) that allow you to configure a Syslog target.

Table 22 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

Many Symmetrix events are organized into categories. These categories are hierarchical in that a category can contain individual events, as well as other categories. An event list is a mechanism for specifying the types of events for which to generate traps.

To build an event list, define the following parameter in the `daemon_options` file:

```
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 Symmetrix 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 Symmetrix 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 - 119 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 shown in Appendix B.

`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 98.

`comp` — Specifies the specific subcomponent 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 96 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.

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 104.

In these examples:

- `Symmetrix: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 Symmetrix 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-ddThh: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 Symmetrix 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 Symmetrix 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]	Symmetrix ID. Optional. Identifies the Symmetrix 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 Symmetrix 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 Symmetrix 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]	Symmetrix 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 Symmetrix 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 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.

- For the time being, the assumption is 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.

  If this becomes a problem, an escape mechanism can be supported in the future -- allowing, for example, user names or other SDEs that contain a ‘]’ 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.

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.
- 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 blue.

```
<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 Symmetrix Audit Log) these will be in UTC time ('Z' suffix). In most (all?) 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 blue (before the fmt SDE) was added by the Solaris sylog logic.

```
Dec 17 10:33:20 182ab139 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 (Symmetrix) 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 Symmetrix 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 Appendix B, "Asynchronous Events." You can return a list of events and descriptionus using the command **`stordaemon action storevntd -cmd list -events`**.

Symmetrix 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 23

Symmetrix 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 23contains the possible values.

Table 23 Solutions Enabler event daemon event UID values (page 1 of 2)

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

Table 23 Solutions Enabler event daemon event UID values (page 2 of 2)

UID (integer value)	Component
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
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 24) that allow you to configure a Syslog target.

Table 24 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.
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 is the recovery table considered valid for the event daemon to load on startup.

CHAPTER 4
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, Windows, and OpenVMS 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

At this point in the install, the `netcnfg` file is a template and an editable file located in the SYMAPI configuration directory.[1]

Using a text editor, a System Administrator must add the network services to the file in the following format:

```
service_name domain_name network_protocol server_node_name server_network_address port_number
security_level
```

where:

service_name is the name of the service.

domain_name should be unspecified and substituted with a hyphen (-).

network_protocol must be TCPIP.

1. The location of this directory varies according to the operating system. For more information, refer to Appendix E.

server_node_name is the name of the server host.

server_network_address is the network address of the server.

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 117.

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 *Solutions Enabler 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.

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 OVMS
set SYMCLI_CONNECT=SYMAPI_SERVER	for Windows

To determine what network services are available, 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 Symmetrix database is strictly read and updated remotely.

`LOCAL`

Defines a local connection to the Symmetrix 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, as well.

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 119).
- **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 *Solutions Enabler Security Configuration Guide* for more information on client/server security and how to configure related settings.

Specifying server behavior

Table 25 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 91.

Table 25 storsrvd options for the daemon_options file (page 1 of 3)

Parameter	Possible values[a]	Reloadable
port Specifies the decimal port number.	= *nnnn* | **2707**	No
security_level Specifies the session security level. For more information, refer to the *Solutions Enabler Security Configuration Guide*.	= NONSECURE | ANY | SECURE | ***based on platform***	Yes
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

Table 25 storsrvd options for the daemon_options file (page 2 of 3)

Parameter	Possible values[a]	Reloadable
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, 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.	Yes

Table 25 storsrvd options for the daemon_options file (page 3 of 3)

Parameter	Possible values[a]	Reloadable
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 server. This parameter is not supported in z/OS. This value is ignored if secure communications are not established.	= NOVERIFY MUSTVERIFY | **VERIFY** 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.	Yes

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
```

Note: For OpenVMS, you can automate this process by placing a call to `emc_start_storsrvd.com` in `SYSSTARTUP_VMS.COM`. For more information, refer to "Installing Solutions Enabler on OpenVMS" on page 55.

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:

- 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 displays the output of a `stordaemon show storsrvd` command:

```
stordaemon show  storsrvd

Daemon State                                : Running
 Daemon Start Time                          : Sun Nov 21 17:38:59 2010
 Version                                    : V7.5.0-1103 (0.0)
 Auto-Restart by Watchdog                   : Disabled

 Total Number of Connections                : 3
 Number of Active Connections               : 1
 Total Number of Requests                   : 5

 ANR0123I Show Server Details               :

 SYMAPI Version                             : V7.5.0.0   (Edit Level: 1103)
 SYMAPI Session Total/Active                : 2/1
 SYMAPI Session Port                        : 2707
 Security Level                             : SECURE
 Show ANR Category                          : Disabled
 Show ANR Message Id                        : Enabled
 Enhanced Authentication                    : Disabled
 Client Verification Level                  : VERIFY
```

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 displays the output of a `stordaemon action storsrvd -cmd show server` command:

```
./stordaemon action storsrvd -cmd show server

ANR0123I Show Server Details:

SYMAPI Version                : V7.5.0.0   (Edit Level: 1103)
SYMAPI Session Total/Active   : 2/1
SYMAPI Session Port           : 2707
Security Level                : NONSECURE
Show ANR Category             : Disabled
Show ANR Message Id           : Enabled
Enhanced Authentication       : Disabled
Client Verification Level     : VERIFY

ANR0123I Show OS Information Details:

Process ID                    : 18090
Host OS Name/Version          : Linux/2.6.18-164.el5
Processor Model/CPUs          : x86_64/3
```

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           : 5000
 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 displays 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:  7.5.0
Session Started: 2008/04/07 17:25:53   Seclevel: NONSECURE
Total Requests:  2
Last Request:    SymUserContextSet  (4190)
     Started:    2010/11/24 13:07:32
     Ended:      2010/11/24 13:07:32   Result:      0 (SYMAPI_C_SUCCESS)
Client host information:
    PID:         11992
    OS:          SunOS
    Addressing:  32-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 (32-bit, 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 121.

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 Appendix A.

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 121.

CHAPTER 5
Post-Install for z/OS

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 `SEMAGENT` to all started tasks whose names start with `SEMAGENT`:

```
RDEFINE STARTED SEMAGENT.* UACC(NONE) STDATA(USER(SEMAGENT))
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.

Installing the SSL certificates

Solutions Enabler optionally allows the use of SSL encrypted communications between the SYMAPI server and the clients connecting to it. You can enable or disable SSL support from either the client or server side.

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.

Note: 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 *Solutions Enabler Security Configuration Guide*.

If SSL is requested on startup (through settings in the `options` file or `daemon_options` file), the SYMAPI server will attempt to read the SSL certificates from the USS file system. Therefore, you must create and install the certificates before starting the server.

To install the certificates into the USS directory:

1. 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 from installation disc" on page 48.

   For example:

   `zoscert.bat create`

Note: When running the `zoscert.bat` batch file on a Windows host, you may receive a Windows error message regarding missing dlls, for example `msvcr80.dll`. This is most likely due to runtime files for Visual Studio 2005 (also known as Visual Studio 8) not being installed. To fix this problem, install the Visual Studio redistribution kit that is provided on the DVD in the Solutions Enabler Version 7.4 kit. The OS-specific files `vcredist_x86.exe` and `vcredist_x64.exe` are located in the folder `Other\zOS\VC8`.

2. When prompted, provide the following information:
   - The fully qualified name of the z/OS host (hostname including the domain name). To get this name, ping the z/OS host from the Windows host.
   - The FTP port number (default 21) of the z/OS host.
   - The z/OS userid used to sign in. The userid must have all the requisite permissions to write to the SYMAPI base directory.
   - The SYMAPI base directory (specified when running the SEMJCL exec on z/OS).
   - The password for the z/OS userid.

Once completed, the certificates will be generated and uploaded to the correct location inside the USS 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 *Solutions Enabler Security Configuration Guide*. The `zoscert.bat` file accepts the same parameters as the batch file `manage_server_cert.bat`, namely `create`, `update`, and `secure`. The file `manage_server_cert.bat` must not be run directly. It will be invoked by `zoscert.bat` with the correct options.

Configuring Solutions Enabler

This section explains how to configure Solutions Enabler in a z/OS environment.

CA TCPAccess support

If you are using the Unicenter:TCPAccess Communications Server stack from Computer Associates, you may want to add a SYSTCPD DD statement to the JCL to identify site-specific configuration information. If the parameters pointed to by this DD are not correct when a client tries to connect, you may receive error messages. The dataset must have the following attributes:

- LRECL=80
- RECFM=FB
- BLKSIZE=Multiple of the LRECL
- DSORG=PS

When configuring TCPAccess to connect to the USS kernel, special system configuration steps are required. For more information on the SYSTCPD statement and USS configuration for Unicenter:TCPAccess, refer to the appropriate Computer Associates documentation.

The following statements are required for the SYMAPI server:

```
TCPIPJOBNAME TCPAccess Jobname
DOMAINORIGIN companyname.COM
NSINTERADDR DNS IP Address
DNRSSID TCPAccess Subsystem ID
```

Note: SYSTCPD statements must not be allocated to JES SYSIN (DD*) files.

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 USS.

Note: Beginning with Solutions Enabler V7.1, the SYMAPI database (symapi_db.bin) uses a new format. Solutions Enabler V7.1 and higher can use a database written by an earlier version of Solutions Enabler as long as it is in the symapi_installation_directory_/db directory in USS. However, versions of Solutions Enabler prior to V7.1 cannot read/use a database written to by V7.1 or higher.

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 USS 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 USS path of the database. When a session requests the default database, SYMAPI attempts to use the fully qualified USS 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 Symmetrix configuration does not apply to the z/OS type platforms. However, z/OS servers do communicate to the Symmetrix system using a UCB on the first device found in the Symmetrix 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 Symmetrix 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 Chapter 6. For more information on specifying devices to use/avoid from using as gatekeepers, refer to "Avoidance and selection files" on page 134.

Solutions Enabler files

Solutions Enabler can reference various files during processing. These files and the following text are for experienced SYMAPI server users and are not a prerequisite for normal use.

For simplicity sake, the examples in the following tables show the use of instream data for input files. Generally, EMC does not recommend instream datasets for SYMOPT, SYMAVD, SYM$GAVD, SYM$GSEL or SYM$INQ. If the SYMAPI server has a high connection arrival rate, use of instream datasets may cause 013-C0 abends. Generally, the startup JCL should reference a PARMLIB member which contains the appropriate statements for the desired file. Prepare a PARMLIB member with the statements for one of the specified files above, and code a JCL statement using DISP=SHR,DSN=*ds-prefix*.PARMLIB(*member*) for each. For an example, see the specification of the SYM$OPT DD statement in the #STORSRV member of the PARMLIB.

SYMAPI files

Table 26 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 USS files.

Note: For USS supported files, SYMAPI will only use a USS location if the corresponding DD name is not specified in the SYMAPI server JCL (comment it out or delete it).

Table 26 SYMAPI files

DD name	File type	Description
SYM$LIC	Dataset, USS	An input file for the Solutions Enabler license information. Dataset: *ds-prefix*.License USS: *symapi_installation_directory*/config/symapi_licenses.dat
SYM$OPT	PARMLIB, USS	The SYMAPI options file. For more information, refer to "Changing the default behavior of SYMCLI" on page 85. PARMLIB: *ds-prefix*.PARMLIB (symopt00) USS: *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	PARMLIB, Dataset, USS	Defines a list of trusted hosts and users who are allowed to connect to the server. For more information, refer to the *Solutions Enabler Security Configuration Guide*. USS: *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 27 lists SYMAPI files or DDs that can exist in the Solutions Enabler startup JCL, which limit the scope or change the performance of Solutions Enabler during the discovery process.

These files can be used to customize and streamline command line coding for your specific environment.

These are editable files with device names or Symmetrix IDs that you use to limit the effect of commands to include or exclude the specified devices, gatekeepers, or Symmetrix arrays. The files hold either volume serial names (*volser*) or Symmetrix IDs (*Symmids*) with line entries having only one device name or ID per line. Lines beginning with a # (comment) are ignored.

Table 27 Solutions Enabler avoidance and selection files (page 1 of 2)

DD name	File type	Description
SYM$AVD[1]	PARMLIB USS	This file affects the operation of the discovery process so that it skips devices that belong to the Symmetrix arrays identified in this file. This may be useful if there are multiple Symmetrix arrays connected to the host that you wish the discovery to avoid. The Symmetrix avoidance file is formatted with 12-character Symmetrix IDs, with one ID per line. For example, to avoid discover of the Symmetrix with a serial number of 0000183600186, code the following statements in your JCL: //SYM$AVD DD * 0000183600186 /* PARMLIB: *ds-prefix*.PARMLIB (*member*) USS: *symapi_installation_directory*/config/symavoid
SYM$INQ[1]	PARMLIB USS	This file affects the inquiry and discovery processes so that they find only the volume serial name (volser) specified in this file. This maybe useful if you want to limit the command(s) to affect only certain Symmetrix devices from your host. The inquiry file is formatted with volume serial names (volser), with one volser per line. For example, to include information on volume ABC123 (only) and the Symmetrix to which it is attached, use a SYM$INQ file that looks like this: //SYM$INQ DD * ABC123 /* PARMLIB: *ds-prefix*.PARMLIB (*member*) USS: *symapi_installation_directory*/config/inqfile
SYM$GAVD[1]	PARMLIB USS	This file affects calls to commands that use a gatekeeper to communicate to a Symmetrix 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 Symmetrix array. This could be useful to designate certain Symmetrix devices that should not be used as gatekeepers. The gatekeeper avoidance file is formatted with volume serial names (volser), with one per line. For example, to instruct Solutions Enabler for z/OS to avoid using volume DEF456 as a gatekeeper device, use the following SYM$GAVD statement: //SYM$GAVD DD * DEF456 /* PARMLIB: *ds-prefix*.PARMLIB (*member*) USS: *symapi_installation_directory*/config/gkavoid

Table 27 Solutions Enabler avoidance and selection files (page 2 of 2)

DD name	File type	Description
SYM$GSEL[1]	PARMLIB USS	In SYM$GSEL, specify serials for the volumes you prefer to be gatekeepers. Specify one volume serial per line, with no other text on the line. **Note:** If a SYM$GSEL list is not defined for a particular Symmetrix array or if the specified volumes to do not exist at the time the file is read (every time a CLI command is run), then normal gatekeeper selection rules will apply for that Symmetrix array. If you specify a volume serial in both the SYM$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$GSEL, be sure to specify at least one volume on each Symmetrix system you want to access through Solutions Enabler. For example, to instruct Solutions Enabler to give preference to volumes GHI123, JKL123 and MNO123, use the following SYM$GSEL DD statement: //SYM$GSEL DD * GHI123 JKL123 MNO123 /* PARMLIB: *ds-prefix*.PARMLIB (*member*) USS: *symapi_installation_directory*/config/gselect **Note:** If you specify a volume in BOTH the SYM$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.

1. This file can also be referred to by way of a PATH statement. However, if you want to store it in USS, it is recommended that you remove or comment out this DD statement and let it default to the correct USS location.

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 50), you can enter the appropriate setting for your time zone. "Installing Solutions Enabler on z/OS" on page 48 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 Symmetrix CLI Command Reference*.

Remote control operations

Symmetrix control operations can be executed by the SYMAPI server on behalf of remote clients such as SYMCLI, Unisphere for VMAX, or EMC ControlCenter. These control operations are enabled by default.

Restricting remote control operations

Starting with Solutions Enabler V7.4.0, 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 28 lists some of the control operations that can be disabled in the z/OS server.

Table 28 Examples of z/OS control operations (page 1 of 2)

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 Symmetrix 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 Symmetrix devices that meet a specified selection criteria.
SymDevListRdfControl	Invokes an RDF control action on a list of devices.
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 Symmetrix 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.

Table 28 Examples of z/OS control operations (page 2 of 2)

Function	Action
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 Symmetrix 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

Hint: Make a copy of member `#12CNTRL` for backup purposes before making any changes.

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.

Modifying the control table

Hint: Use the backup copy of the job as a reference.

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 28 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`.

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 Symmetrix 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.

IMPORTANT

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 123.

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 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 29 compares the syntax of the `stordaemon` commands issued from a USS 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 29 stordaemon command syntax for the z/OS system console (page 1 of 2)

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 stordaemon shutdown storsrvd -immediate	F STORSRVD,SHUTDOWN F STORSRVD,SHUTDOWN [-]IMMediate
Show the current value of an operational variable (port in this example).	stordaemon getvar storsrvd –name port	F STORSRVD,'getvar [-]name port'

Table 29 stordaemon command syntax for the z/OS system console (page 2 of 2)

Command	stordaemon syntax	Console syntax
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, 2012,
IKJ56951I NO BROADCAST MESSAGES,
REXX/SOCKETS z/OS V1R6 January 5, 2007,
Network Info - IP:192.168.1.1,
               Domain:TSO.DOMAIN.COM,
               Hostname:HOST01,
READY

STORDEMN show storsrvd
<output will show here>

CALL 'EMC.SSEM750.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 USS shell

The following example illustrates how you can configure `stordaemon` to run from USS. For the sake of this example, assume that the user has already logged in to the z/OS USS shell either via rlogin or the TSO `OMVS` command:

```
$ cd /var/symapi
$ mkdir bin
$ cd bin
$ ln -e STORDEMN stordaemon
$ export STEPLIB=EMC.SSEM750.LOADLIB
$ stordaemon show storsrvd
$ stordaemon shutdown storsrvd
```

In the example, the user makes an external link from a USS file to the Solutions Enabler load library module. By setting the STEPLIB environment variable, the shell follows the link from the USS 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 optional for the z/OS SYMAPI server. The base daemon provides numerous benefits for the z/OS environment, including improved performance (via caching of syscall results) and enhanced Symmetrix lock management.

Most of the information in this section is similar to the daemon information described in Chapter 3; however, this section describes it from the z/OS point of view.

Installing or uninstalling the base daemon

The base daemon is automatically installed during Solutions Enabler installation, as its load modules reside in the same loadlib.

In z/OS, there is no support for uninstalling the base daemon.

Starting the base daemon

Once the 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 30 lists the commands for stopping the base daemon.

Table 30 Commands for stopping the base daemon

From	Use the command
Console	F STORAPID,SHUTDOWN
TSO	stordemn shutdown storapid
USS shell	stordaemon shutdown storapid

For more information on using these methods, refer to "Controlling the server" on page 141.

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 91.

By default, if the base daemon is running, then the z/OS SYMAPI server will connect to it and use its features. If it is not running, then the server will not attempt to start or use it.

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 91.

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 USS.

You should not use both MVS datasets (for the server) and USS 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 USS 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 USS.

For more information on the avoidance and selection files, refer to "Avoidance and selection files" on page 134.

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 94.

In the z/OS context, the event daemon is primarily used to enable monitoring capabilities on behalf of other clients. For this release, the only client expected to use the event daemon is the EMC Unisphere for VMAX.

Installing or uninstalling the event daemon

The event daemon is automatically installed during Solutions Enabler installation, as its load modules reside in the same loadlib.

In z/OS, there is no support for uninstalling the event daemon.

Starting the event daemon

Once the 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 31 lists the commands for stopping the event daemon.

Table 31 Commands for stopping the event daemon

From	Use the command
Console	F STOREVTD,SHUTDOWN
TSO	stordaemon shutdown storevntd
USS shell	stordaemon shutdown storevntd

For more information on using these methods, refer to "Controlling the server" on page 141.

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 91.

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 91.

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 Damon joblog. 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 104 describes the formats of event messages in detail.

CHAPTER 6
Gatekeeper Devices

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 Symmetrix arrays. It receives user requests via CLI, GUI, or other means, and generates system commands that are transmitted to the Symmetrix 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 and Solutions Enabler V7.4, 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 Symmetrix 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 150). 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
- Virtual devices (VDEVs)

Note: Starting with Enginuity 5876 and Solutions Enabler V7.4, thin devices may be selected as gatekeepers, except on AS400.

Solutions Enabler selects a gatekeeper from the candidate list based on a preestablished 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 Symmetrix 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 Symmetrix 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 Symmetrix 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, Symmetrix 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 Symmetrix 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 150. 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 Symmetrix array is installed, the EMC Customer Engineer selects and configures Symmetrix 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 Symmetrix configurations, refer to EMC Knowledgebase article emc255976 available on EMC Online Support.

You can determine the storage size of a Symmetrix 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 Symmetrix 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 Symmetrix 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 Symmetrix configurations, refer to EMC Knowledgebase article emc255976 available on EMC Online 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=nnn]
[, [mapping to dir DirNum:PortNum
[starting] target = scsi_target,
lun=scsi_lun, vbus=fibre_vbus
[starting] base_address cuu_address]...];
```

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.

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.

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.

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 Symmetrix array for which you want to display information. Issuing this command without the `-sid` option will display information on all Symmetrix 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: 000195700433

                              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                  01/19/2012 10:57:03

    Gatekeeper Utilization             25 %
    Time of Highwater                  01/19/2012 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 Symmetrix array for which you want to display information. Issuing this command without the `-sid` option will display information on all Symmetrix arrays. The `-v` option specifies to display a verbose listing.

For example:

`stordaemon action storapid -cmd show -gk_pdevs -sid 343`

CHAPTER 7
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 159.

Stopping the application processes

To stop the application processes:

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

   ```
   fuser /usr/lib/libsym* /usr/lib/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 89.

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 89.

Uninstalling the software

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

- For UNIX, refer to "Uninstalling Solutions Enabler from UNIX" on page 156.
- For OpenVMS, refer to "Uninstalling Solutions Enabler from OpenVMS" on page 161.

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 (e.g., `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:

```
./se7500_install.sh -uninstall
```

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

```
./se7500_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:

```
./se7500_install.sh -decrement [-cert][-jni] [-srm] [-64bit] [-symrec]
```

Note: This method is not supported on Solaris.

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

```
./se7500_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 32 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 156.

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-V7.5.0-0`

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

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

`rpm -e symcli-thincore-V7.5.0-0`

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:

1. Change 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:

   `se7500-Windows-Processor_type.exe`

   Where *Processor_type* is the operating system. Possible values are x86, x64, and ia64.

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 **InstallShield Wizard Completed** 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\
se7500-Windows-Processor_type.exe /S /X /V/qn
```

Where:

Processor_type can be x86, x64, or ia64.

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.

Removing the msi image

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

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

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

  Possible values for *GUID* are:

  {AE844528-8779-4955-A958-2C3319C3A4A9} Solutions Enabler
  {BA22747A-D76A-4CE3-9D4D-AE3E85D65534} ControlCenter
  {EBB43F68-733E-40B3-A21C-CEF537BEBB81} SP
  {AF2A82C2-C2AF-4AF4-89B2-EAC388937DFD} STORBLK
  {AADF63FA-62CF-4E53-9E12-1EAD1AE97AAD} SMI
  {ECA51E73-0BA1-44D6-8200-E92D3A18159D} SDK
  {BA376762-1C61-4815-B950-3F9F5898067A} TCLIENT

- 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

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

h. 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

i. 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:

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 **Remove**.

Using the Windows Programs and Features dialog

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

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 **Remove**.

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.

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 156.

2. 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.
3. Delete all the files from the installation directory.

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 18:

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 156.

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 Symmetrix Array Management CLI Product 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:
   - If the database can be used, the rollback is done.
   - If the database cannot be used, issue a `symcfg discover` command to create a Symmetrix host database file, `symapi_db.bin`, and import all the exported device groups.

CHAPTER 8
Deploying the Solutions Enabler Virtual Appliance

This chapter explains how to deploy the Solutions Enabler Virtual Appliance in a VMware infrastructure environment:

Introduction

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

- EMC Solutions Enabler V7.5.0 (solely intended as a SYMAPI server for Solutions Enabler client access)
- Linux OS (SUSE 11 SP1 JeOS)
- SMI-S Provider V4.5

In addition, the Solutions Enabler Virtual Appliance includes a browser-based console called EMC vApp Manager for Solutions Enabler to configure your storage environment. vApp Manager enables you to perform the following configuration tasks:

- Monitor the application status
- Start and stop selected daemons
- Import and export persistent data
- Configure the `nethost` file (required for client access)
- Discover storage arrays
- Modify options and daemon options
- Add Symmetrix-based and host-based license keys
- Run a limited set of Solutions Enabler CLI commands
- Configure ESX host and gatekeeper devices
- Launch Unisphere for VMAX (available only in Unisphere versions of the appliance console)
- Configure iSCSI initiator and map iSCSI gatekeeper devices
- Configure additional NIC card (optional)
- Download SYMAPI debug logs
- Import CA signed certificate for web browser
- Import Custom certificate for storsrvd daemon
- Check disk usage
- Restart appliance
- Configure symavoid entries
- Load Symmetrix-based eLicenses
- Enable SSH
- Configure LDAP
- Manager users
- Reset hostname
- Update etc/hosts

Note: For information on using vApp Manager, refer to its online help.

Note: Root login is not supported on the SUSE 11 virtual machine.

Before you begin

Before you begin to deploy the Solutions Enabler Virtual Appliance, be sure to complete the tasks listed in this section:

- ❑ Verify that you are installing the latest version of the appliance by checking EMC Online Support for updates.
- ❑ Verify that the client is running:
  - VMware vSphere Client
  - Either of the following browsers with cookies and javascript enabled:
    - Internet Explorer 6.0 through 8.0
    - Firefox 3.5 and above

    Browsers should have Flash Player 11.2 plug-in installed. If your browser has an outdated version of Flash Player, you will be prompted to download the latest version when you start the web console.
- ❑ Verify that the VMware ESX Server meets the following minimum requirements:
  - Version 4.0 or higher
  - Dual disk. 11 GB of disk space and another 5 GB (expandable) disk space
  - 2GB memory
  - 1 CPU

Deploying the virtual appliance directly to the ESX Server

This section describes how to deploy the Solutions Enabler Virtual Appliance directly to the ESX Server.

Step 1: Import the virtual appliance

To import the virtual appliance:

1. Download the OVF archive file (`*.ova`) containing the installation program from EMC Online Support to a temporary directory.
2. Start the vSphere Client and log in to the ESX Server on which you will be deploying the appliance.
3. Click **Ignore** in the security warning message.
4. From the **File** menu, select **Deploy OVF Template**.
5. Browse to the OVF archive file, located in the temporary directory you created earlier. Select the OVF archive file with the suffix `*vapp_OVF10.ova`.
6. Click **Next**.
7. On the **Details** page, verify the details about the appliance and click **Next**.
8. On the **End User License Agreement** page, select **Accept all license agreements** and click **Next**.
9. On the **Name and Location** page, specify a name for the appliance and click **Next**.
10. If a resource pool is available, the **Resource Pool** page displays. Select the resource pool of your choice and click **Next**. Otherwise, the **Resource Pool** page is skipped.
11. On the **Datastore** page, select the datastore of your choice and click **Next**.

12. On the **Disk Format** page, select the format in which to store the virtual machine's virtual disks and click **Next**.
13. On the **Network Mapping** page, map the source network to the appropriate destination network.
14. On the **Ready to Complete** page, verify the information and click **Finish**.
15. In the Completed Successfully message, click **Close**.
16. Continue with "Step 2: Select gatekeepers" next.

Step 2: Select gatekeepers

Present uniquely defined gatekeeper by way of raw device mappings (RDM). For instructions, refer to the appropriate VMware documentation.

Solutions Enabler manages Symmetrix arrays through gatekeeper devices mapped to the virtual appliance as RDM pass-through devices. The management is done through EMC proprietary commands using SCSI 3B/3C write/read commands. For every call, a WRITE command is issued to send the request, and then a READ command to get the results.

Note: Gatekeepers can be added using vApp Manager. For detailed information, refer to vApp Manager online help.

Continue with "Step 3: Power on and configure the Virtual Appliance" below.

Step 3: Power on and configure the Virtual Appliance

To power on and configure the Virtual Appliance:

1. On the **Summary** page of the Virtual Infrastructure Client, click **Power On**.
2. Click the **Console** tab and watch as the appliance starts up.
3. Read and accept the license by typing **`yes`** at the following prompt and pressing **Enter**:

   ```
   Do you agree with the terms of the end user license agreement? yes/no
   [no]:
   ```

4. At the following prompt, type **`y`** and press **Enter** to configure static IP address:

   ```
   Do you want to configure static IP address? [y]/n:
   ```

   - A [**`y`**]`es` response produces the following series of prompts that will enable you to configure your network:

     – `IP Address [ ]:`

       Type the address assigned to the appliance and press **Enter**.

       Note: The virtual appliance uses this IP address to query the DNS Server and get its hostname. Therefore, you must ensure that the IP address has a hostname mapping in the DNS Server.

     – Netmask [ ]:

Type the mask of the network on which the appliance will be running and press **Enter**.

– Gateway [ ]:

Type the gateway address to the network on which the appliance will be running and press **Enter**.

– DNS1 [ ]:

Type the first DNS server address and press **Enter**.

– DNS2 [ ]:

Type the second DNS server address and press **Enter**.

– Is a proxy server necessary to reach the internet? y/n [n]:

A [**y**]es response enables you to specify the IP address of the proxy server and the port.

- A [**n**]o response continues the configuration.

The network is configured at this point.

5. At the following prompt, specify whether you want to set the time zone:

```
Do you want to set the time zone? y/[n] :
```

- A [**n**]o response continues the configuration. If you select this option, you can use the appliance console to specify the time zone at a later time.
- A [**y**]es response produces the following series of prompts that will enable you to set the time zone:
  – Please select a continent or ocean

    Type the number that corresponds to the time zone location and press **Enter**.

  – Please select a country

    Type the number that corresponds to the country-specific time zone you want to set and press **Enter**.

  – Please select one of the following time zone regions

    Type the number that corresponds to regional time zone you want to set and press **Enter**.

The time zone is now set.

6. At the following prompt, specify whether you want to enter the host ESX Server information:

```
Do you want to set the host ESX Server y/[n]? :
```

- A **n** response continues the configuration. If you select this option, you can use the Configuration Manager to enter the host ESX Server details at a later time. For instructions, refer to the Configuration Manager's online help.
- A **y** response prompts you for the ESX Server hostname. In which case you should type the fully qualified hostname of the ESX Server and press **Enter**.

When prompted to enter the root password, type the root password of the ESX Server and confirm it by typing it again.

A Welcome screen displays. You have now finished installing the Solutions Enabler Virtual Appliance.

7. Continue with "Launching vApp Manager" on page 171.

Deploying the virtual appliance through a vCenter Server

This section describes how to deploy the Solutions Enabler Virtual Appliance through a vCenter Server 4.0 and higher.

Step 1: Import and configure the virtual appliance

To import and configure the virtual appliance:

1. Download the OVF archive file (`*.ova`) containing the installation program from EMC Online Support to a temporary directory.
2. Start the vSphere Client and log in to the vCenter Infrastructure Server through which you will be deploying the virtual appliance.
3. Click **Ignore** in the security warning message.
4. From the navigation tree, select the ESX Server on which you will be deploying the virtual appliance.
5. From the **File** menu, select **Deploy OVF Template.**
6. Browse to the OVF archive file, located in the temporary directory you created earlier. Select the OVF archive file with the suffix `*vapp_OVF10.ova`.
7. Click **Next**.
8. On the **Details** page, verify the details about the appliance and click **Next**.
9. On the **End User License Agreement** page, select **Accept all license agreements** and click **Next**.
10. On the **Name and Location** page, specify a name for the appliance and click **Next**. It is recommended that you name the appliance with the same fully qualified hostname of the virtual appliance.
11. Select the host/cluster to run the virtual appliance.
12. If a resource pool is available, the **Resource Pool** page displays. Select the resource pool of your choice and click **Next**. Otherwise, the **Resource Pool** page is skipped.
13. On the **Datastore** page, select the datastore of your choice and click **Next**.
14. On the **Network Mapping** page, map the source network to the appropriate destination network.
15. Customize the software solution for this deployment by doing the following:
    a. Provide valid values for the following OVF properties:
       – IP Address
       – Netmask

- Gateway
- DNS Server 1
- DNS Server 2

Note: The virtual appliance uses this IP address to query the DNS Server and get its hostname. Therefore, you must ensure that the IP address has a hostname mapping in the DNS Server.

b. Optionally, provide/select valid values for the following OVF properties:

- Proxy Server: Enter the IP address of the proxy server and port. For example: `ProxyServer-IP:Port`
- ESX Server Name: Enter the fully qualified ESX Server hostname.
- ESX Server Password: Enter the ESX Server password in base64 encryption format.

16. On the **Ready to Complete** page, verify the information and click **Finish**.
17. In the Completed Successfully message, click **Close**.
18. Continue with "Step 2: Select gatekeepers" next.

Step 2: Select gatekeepers

1. Select gatekeepers as described in "Step 2: Select gatekeepers" on page 166.

   You can configure the virtual appliance to add two gatekeeper devices per Symmetrix array when it firsts boots up. For instructions, refer to step 10 on page 168.

2. Continue with "Step 3: Power on the virtual appliance" next.

Step 3: Power on the virtual appliance

To power on and configure the Virtual Appliance:

1. On the **Summary** page of the Virtual Infrastructure Client, click **Power On**.
2. Click the **Console** tab and watch as the appliance starts up.

   A Welcome screen appears. You have now finished installing the Solutions Enabler Virtual Appliance.

3. Continue with "Launching vApp Manager" next.

Deloying the virtual appliance using OVFTOOL

Solutions Enabler Virtual Appliance can be deployed through command line from any Linux host. This section how to deloy the virtual appliance using OVFTOOL.

To deploy Solutions Enabler Virtual Appliance using OVFTOOL, the following are required:

- vCenter Server 4.0 and above.
- ESX Server 4.0 and above managed by vCenter Server 4.x.

- ovftool 1.0 and above

Note: Please refer to the appropriate documentation for installing vCenter Server and VMware ovftool.

Here is a brief description of the steps on how to deploy the virtual appliance using OVFTOOL:

1. Install and Setup the vCenter Server.
2. Add the ESX Server to the vCenter Server datacenter.
3. Install VMware OVFTOOL on a Linux host.
4. Move the Solutions Enabler Virtual Appliance kit to the above host.
5. Run the ovftool command with necessary commandline switches. For more information on using the command, refer to "Using OVFTOOL."
6. Solutions Enabler Virtual Appliance is deployed and powered on automatically.
7. Continue with "Launching vApp Manager" next..

Using OVFTOOL

OVFTOOL has the following syntax:

```
/usr/bin/ovftool --acceptAllEulas --overwrite --powerOffTarget
   --powerOn --prop:ipAddress=<IP-ADDRESS> --prop:netmask=<NETMASK>
   --prop:gateway=<GATEWAY> --prop:dns1=<DNS1> --prop:dns2=<DNS2>
   --prop:timezone=<TIMEZONE> --prop:esxServer=<ESX-SERVER>
   --prop:encr yRootPasswd=<ROOT-PASSWORD> --name=<VM-DISPLAYNAME>
   --datastore=<DATASTORE>  --net:Network\ 1=<VM Network Port Group>
   --net:Network\ 2=<VM Network Port Group>   <OVA-FILE>
   vi://Administrator:<vCenter-admin-passwd>@<vCenter-Server>/<DataCen
   ter-Name>/host/<esx-server-name>
```

Where:

‹IP-ADDRESS›	IP Address of the Virtual Appliance.
‹NETMASK›	Netmask of the Virtual Appliance.
‹GATEWAY›	Gateway
‹DNS1›	IP of DNS Server1.
‹DNS2›	IP of DNS Server2.
‹TIMEZONE›	Time Zone setting. (Optional)
‹ESX-SERVER›	Fully qualified hostname of ESX server. (Optional)
‹ROOT-PASSWORD›	Root password of ESX Server in base64 encrypted format. (Optional)
‹VM-DISPLAYNAME›	VM Displayname. To automatically add gatekeeper devices during virtual appliance boot, VM Displayname to be same as fully qualified hostname of Virtual Appliance.
‹DATASTORE›	Name of the datastore attached to ESX Server. Required only if more than one datastore is attached to ESX Server.

‹VM Network Port Group›	VM network port group. If both NIC cards need to be in different network, then the VM Network port group need to be different.
‹OVA-FILE›	Absolute path of ova file.
‹vCenter-Server›	Name of the vCenter.
‹vCenter-admin-passwd›	vCenter Server's Administrator password.
‹esx-server-name›	ESX Server name as displayed in the vCenter Server.

Launching vApp Manager

To launch vApp Manager:

1. Type one of the following URLs in a browser:

   `https://appliance_IP:5480`

   or

   `https://appliance_host_name:5480`

2. On the log in panel, type **`seconfig`** for both the User and Password, and then click **Login**.

   Note: It is recommended that you change your password from vApp Manager on first login. vApp Manager can also be configured to use LDAP for user authentication. For more information on that, refer to vApp Manager online help.

3. vApp Manager displays. For information on using vApp Manager, refer to its online help.

Registering VASA Provider with vSphere

VMware VASA (VMware APIs for Storage Awareness) Provider improves VMware vSphere's ability to monitor and automate storage related operations. VASA Provider reports information about storage topology, capabilities, and status, as well as storage events and alerts to VMware. It is a standard vSphere management plug-in that is deployed on each vCenter server, and it interacts with VMware APIs for Storage Awareness.

To register the VASA Provider with vSphere:

1. Connect to the VMware vCenter Server 5.0 or above using vSphere Client.
2. In the Virtual Data Center, navigate to **Home › Administration › Storage Providers**, and select **Storage Providers** in the navigator bar.
3. In the Vendor Providers pane, select **Add**.
4. Add the vendor provider properties (name, url, and login information).

   For ECOM login credentials refer to SMI-S provider documentation.

   For the url use `https://<vapp-ip>:5989/vasa/services/vasaService`.

   When the VASA Provider is connected, the VI Client displays the SSL certificate.

5. Click **Yes** to complete the registration.
6. Verify registration with vSphere:
    1. Navigate to **Home** › **Administration** › **Storage Providers** › **Vendor Providers**.
    2. Verify that the VASA Provider is listed and displays the list of managed storage arrays.

Updating the Solutions Enabler Virtual Appliance

Periodically, EMC will release security patches and hot-fixes for the Solutions Enabler Virtual Appliance. These patches and fixes are available on EMC Online Support in two forms: OVA files and ISO images.

Updating from an OVA file

To update an existing Virtual Appliance from an OVA file:

1. Login to vApp Manager of the existing appliance.
2. Click **Export Persistent Data** to download an archive file containing Solutions Enabler persistent data to your desktop.
3. Extract the archive file to your machine. Note the location of the file **encrypt_se_export_persistent_data_***time-stamp***.zip.gpg**. You will need this file later to complete this procedure.
4. Power off the old appliance.
5. Import and deploy the new appliance in your ESX server. For instructions, refer to "Deploying the virtual appliance directly to the ESX Server" on page 165 or "Deploying the virtual appliance through a vCenter Server" on page 168, depending on your environment.
6. Login to the new appliance's vApp Manager.
7. Click **Import Persistent Data** and browse to the location of the `gpg` file you extracted earlier in this procedure.
8. Click **Import**.
9. When the message `persistent data stored` appears, close the dialog. The update is complete.
10. Restart the virtual appliance using vApp Manager.

Note: During import and export operations, Solutions Enabler daemons would shutdown in the background. This would take several minutes. Please be patient while Solutions Enabler daemons are being shutdown in the background.

Updating from an ISO image

This procedure explains how to upgrade the virtual appliance from V7.5 to a higher version.

Note: You cannot use this procedure to upgrade from V7.3.x or V7.4 to V7.5 since they are running different versions of the SuSE Linux operating system.

To update an existing Virtual Appliance from an ISO image:

1. Upload the ISO image into the ESX Server using the VI client:
   a. Login to the ESX Server using the VI client.
   b. Select the ESX Server on the left panel.
   c. Select the **Configuration** tab on the right panel.
   d. Select **Hardware**, **Storage** to list the datastores connected to the ESX Server.
   e. Right-click the datastore and select **Browse Datastore**.

      The **Datastore Browser** window displays.
   f. Upload the appliance update ISO file.
   g. Exit the dialog.
2. Mount the ISO image on the virtual appliance CD drive:
   a. Right-click the virtual appliance and select **Edit Settings**.
   b. On the **Hardware** tab, select **CD/DVD Drive 1**.
   c. In the right panel, select **Datastore ISO File**, and click **Browse** to locate the ISO image in the datastore.
   d. Select **Device Status**, **Connected**.
   e. Click **OK** to exit the dialog box.
3. Update the appliance:
   a. On the **Console** tab, go to the virtual appliance console.
   b. Use the Move Up/Down keys and select **Appliance Update**.
   c. Press **Enter** to the update.

The update will take approximately 10 minutes, after which the screen will return to the main console.

Note: Use the welcome screens of the vApp and the vApp Manager to confirm your virtual appliance has been updated correctly.

Reconfigure virtual appliance IP Address

This procedure explains how to re-configure an Virtual Appliance's IP Address.

1. Login to vSphere Client and go to the virtual appliance console.
2. Use the Arrow Keys to select **Configure IP** and press **Enter**. The following prompt displays:

```
Do you want to configure static IP address? [y]/n:
```

3. Type **y** and press **Enter** to start configuring an static IP address for the virtual appliance. The following prompt displays:

   ```
   IP Address [10.0.0.10]:
   ```

4. Type a valid IP address and press **Enter**. Alternatively you can just press **Enter** to accept the current IP address.

   Note: The virtual appliance uses this IP address to query the DNS Server and get its hostname. Therefore, you must ensure that the IP address has a hostname mapping in the DNS Server.

   The following prompt displays:

   ```
   Netmask [255.255.252.0]:
   ```

5. Type a valid netmask and press **Enter**. Alternatively you can just press **Enter** to accept the current netmask. The following prompt displays:

   ```
   Gateway [10.0.0.1]:
   ```

6. Type the gateway address of the network on which the appliance will be running and press **Enter**. Alternatively you can press **Enter** to accept the current gateway. The following prompt displays:

   ```
   DNS1 [10.0.0.2]:
   ```

7. Type the first DNS server address and press **Enter**. Alternatively you can press **Enter** to accept the current DNS server one. The following prompt displays:

   ```
   DNS2 [10.0.0.3]:
   ```

8. Type the second DNS server address and press **Enter**. Alternatively you can press **Enter** to accept the current DNS server two. The following prompt displays:

   ```
   Is a proxy server necessary to reach the internet? y/n [n]:
   ```

9. Type **y** to and press **Enter** configure the proxy server. Otherwise, press **Enter** and skip to Step 12.

10. At the following prompt, enter the IP address of the proxy server and press **Enter**.

    ```
    Proxy Server[]
    ```

11. At the following prompt, enter the port of the proxy server and press **Enter**.

    ```
    Proxy Port[]
    ```

12. The following prompt displays. Type y and press Enter to finish configuring the virtual appliance's IP address. Type n and press Enter to go back and restart the process from Step 3.

    ```
    Are the above mentioned network parameters correct? [y]/n :
    ```

Deleting the Solutions Enabler Virtual Appliance

To delete the Solutions Enabler Virtual Appliance:

1. In the vApp Manager interface, backup the persistent data.

2. In the VMware management interface, power down the appliance.
3. Right-click on the appliance and select **Delete from Disk**.
4. Click **Yes** in the confirmation message.

APPENDIX A
SYMAPI Server Daemon Messages

This appendix describes the log messages issued by the SYMAPI server daemon (`storsrvd`):

Message format

This section describes messages that are written to the SYMAPI server log (see "Controlling and using the storsrvd log files" on page 127) and to the system console in z/OS. All messages begin with a message identifier, followed by message text.

The message is in this format:

```
yyyy/mm/dd hh:mm:ss pid thread_name log_category msgid text
```

where:

`yyyy/mm/dd`	Is the date the message was issued.
hh:mm:ss.xxx	Is the time the message was issued in hours, minutes, seconds, and milliseconds.
pid	Is the process ID of the issuing process.
thread_name	Is the thread name of the issuing thread.
log_category	Is the category specified in the `storsrvd:log_filter` statement in the `daemon_options` file, which caused this message to be generated. The valid categories are: SERVER, SESSION, CONTRO, and APIREQ.
msgid	Is made up of the following: `ANR` — Indicates the server issued the message. *nnnn* — A numeric identifier for the message. *X* — A one byte severity indicator. Valid values are: / indicates an Informational message *W* indicates a Warning message *E* indicates an Error message S indicates a severe condition requiring a message
text	Is the message text.

In this section, each message shows the text of the message with indicators where substitutions are made into the text at runtime. Following the text are four paragraphs giving more information:

- *Set Step Return Code* — In a z/OS environment, some messages will cause the SYMAPI server job step return code to be set to a non-zero value. The following table shows the correlation of message severity to job step return code. Some messages are issued by multiple locations in the code. Not all uses of the message will cause the step return code to be set.

Message identifier	Return codes
I	0
W	4
E	8
S	12

  If multiple messages are issued that cause the step return code to be set, the highest value will be remembered by the server, and returned to the system at job termination.

- The *Destination* of the message — `Log` and/or `Console` is shown. Most messages are written to the server log file. Some messages are written to both the log and console, but not in all cases where the message is generated. Some messages are written to the system console only, particularly those related to operator command processing. The Console destination applies only to z/OS.
- The *Description* paragraph explains the circumstances that cause the message to be issued, and explains each substituted value. This section also describes any action that the Solutions Enabler software will take.
- The *Operator Action* paragraph suggests operator intervention actions where needed.

Messages

ANR0000I

text

Destination: Log and console.

Description: This message is a general purpose message to be used for any arbitrary *text*.

Operator Action: None.

ANR0001I

SYMAPI Server for z/OS ready to accept *security_level* connections

Destination: Log and console.

Description: This message is issued when initialization is complete and the server is prepared to field connection requests from remote clients. *security_level* indicates the types of sessions the server will accept. Possible values are:

- ONLY NONSECURE — Indicates that client must expect to negotiate non-SSL sessions with the server.
- ONLY SECURE — Indicates that the server will require clients to negotiate a secure session.
- Both SECURE and NONSECURE — Indicates that the server will accept sessions from clients that cannot negotiate secure and will negotiate secure sessions with clients who can.

Operator Action: None.

ANR0002I

shutdown_type Shutdown requested

Destination: Log and console.

Description: This message indicates that a shutdown request was made. See message ANR0003I for the description of *shutdown_type.*

Operator Action: None.

ANR0003I

shutdown_type Shutdown *progress*. Number of sessions remaining = *number*

Destination: Log and console.

Description: This message is issued at the start of the shutdown process. *shutdown_type* indicates NORMAL, IMMEDIATE, or STOPPED-NORMAL.

In open systems environments, shutdown is requested by the `stordaemon` command.

In Microsoft Windows, you can use the Service Control Manager; in this case the shutdown process will always be IMMEDIATE.

In a z/OS environment, the system operator will request a NORMAL shutdown using the z/OS `STOP` command or the `SHUTDOWN` command.

The number of currently active sessions is shown in *number.* If this value is not 0, the following rules apply:

- If the *shutdown_type* is NORMAL, the server will wait for the active sessions to end. In this case, *progress* indicates *starting* or *in progress*. Each time a session ends, the *in progress* status will be reported.
- If the *shutdown_type* is IMMEDIATE, the server terminates without waiting for active sessions to end. See the description of the SHUTDOWN command for more details on when to use IMMEDIATE shutdown.

Operator Action: None.

ANR0004I

SYMAPI Server running as a started task

Destination: Log.

Description: In a z/OS environment, the server detects when it is running as a started task (running in *STC mode*). This message serves as a visual confirmation that STC mode is active.

Operator Action: None, unless this is not what is intended.

ANR0005E

Normal shutdown failed, attempting immediate shutdown

Set Step Return Code

Destination: Log and console.

Description: The server attempted to perform a normal shutdown, waiting for active sessions to complete. The normal shutdown process failed, and no recovery was possible. An immediate shutdown was attempted, because there is no other possible recovery action to take.

Operator Action: Be aware that the list of connections noted in message ANR0013I will be terminated before they are able to disconnect.

ANR0006E

Wait returned without connection or console command ready, console ECB contents *value*

Destination: Log.

Description: The server waits for incoming connection requests and instructions from the operator concurrently. If the wait is somehow satisfied but neither of these events occurred, it is considered an error. The server will continue to wait for new events.

Operator Action: This is an abnormal situation and may indicate some error in TCP communications or management of the operator console. If this happens repeatedly, shut the server down and try restarting the server. If the problem persists, examine your system for evidence of other problems in the TCP or console management components of your system.

ANR0008I

Server socket *socket*:_*event* occurred

Destination: Log.

Description: This message is issued to confirm that connection request has arrived, or that some error condition has been reflected to the TCP socket on which the server is listening. The value of *socket_event* will be *connection request* or *exception condition*.

Operator Action: If the *socket_event* is *connection request* no action is necessary since this is a documentation message, and may aid in problem diagnosis. See the description of message ANR0009E, if the *socket_event* is *exception condition*.

ANR0009E

Exceeded maximum exceptions on server socket, indicating PORT_EXCEPTION

Set Step Return Code

Destination: Log and console.

Description: This is issued after an exception condition has been raised (which may cause the issuing of ANR0008I). Currently, the maximum exception count is 1, meaning that there is no retry strategy when an exception occurs on the socket on which the server is listening. The server will stop listening and start a NORMAL shutdown when it notices this condition.

Operator Action: If *exception condition* in message ANR0008I is indicated, there will be other evidence in your system log showing TCP/IP problems. Refer to documentation from your TCP software provider to resolve the problems you find. When the problems are resolved, you can restart the server.

ANR0010I

SYMAPI Server Shutdown complete

Destination: Log and console.

Description: The server has completed its shutdown process and will return to the operating system.

Operator Action: None. This should serve as a visual confirmation that the server is finished.

ANR0011W

SYMAPI Server not executing from an APF-authorized library, cannot continue

Set Step Return Code

Destination: Log and console.

Description: In a z/OS environment, the SYMAPI server program `storsrvd` must execute from a library authorized by the z/OS Authorized Program Facility, if the base daemon is not in use. The server checks to make sure that this condition is met. This message is issued as a warning , but an error condition may not be reflected until a SYMAPI session requests storage discovery services.

Operator Action: The Solutions Enabler load library can be authorized through APF in several ways. You can use the SETPROG APF command to authorize the library temporarily. In order to make the library authorized at subsequent IPLs, you must edit the PROGxx member of SYS1.PARMLIB. Refer to the IBM documentation for your level of z/OS for exact syntax and editing instructions.

ANR0012I

Accepted *seclevel* session *session_number* from *IP_address* on thread *thread_number*

Destination: Log.

Description: The server successfully handled a connection request for a session, and started a thread to process API requests for the session. The session number is shown in *session_number* and it is being processed on a thread with the number *thread_number.* The session is running from a client program executing on the host at address *IP_address*. *seclevel* indicates the negotiated security level of the session. If *seclevel* is SECURE, transmission is protected using SSL; if *seclevel* is NONSECURE, SSL protection is not in use.

Operator Action: None necessary. This message is documenting the start of a session. You should also see ANR0017I at the end of the session.

ANR0013I

Shutdown will wait for client session *session_number* from *IP_address* to terminate itself

Destination: Log and console.

Description: During a normal shutdown, the server will wait for all active sessions to terminate on their own. For each session still active, the server issues this message and will wait for the session(s) to end. The substitution variables are the same as those in message ANR0012I.

Operator Action: None usually. If sessions are taking an excessive amount of time to complete, you can reissue the shutdown command with the IMMEDIATE operand to terminate the session immediately.

ANR0016I

SYMAPI listener thread is running on thread *thread_number*

Destination: Log.

Description: This message is issued during startup simply to report the thread number (*thread_number*) of the SYMAPI listener thread (the server thread which listens for new connection requests).

Operator Action: None.

ANR0017I

Ending session *session_number,* total requests executed *total_requests*

Destination: Log.

Description: See also message ANR0012I. This message documents the end of a session. The total number of API requests executed on the session is shown by *total_requests*.

Operator Action: None.

ANR0018E

Rejecting session *session_number* for *user_name@node: reason*

Destination: Log.

Description: A remote client attempted to connect to the running server, but is refused the session for one of the following reasons:

- **The trusted host file disallowed a client server connection** — the nethost file is allocated to the server, and the combination of the node (either host address or IP address) and the optional user identification (user_name) are not specified in the nethost file. The remote client `SymInit` call returns `SYMAPI_C_HOST_FILE_REJECTION`.
- **The trusted host file could not be read** or **The trusted host file has a syntax error** — the nethost file exists, but could not be read or has a syntax error. The client application will receive either `SYMAPI_C_HOST_FILE_READ_ERROR` or `SYMAPI_C_HOST_FILE_SYNTAX`.
- **The maximum number of network connections has been reached on the server** — the global limit expressed by the `max_sessions` option in the `daemon_options` file is exceeded. The application will receive `SYMAPI_C_MAX_SRVR_CONNECTS_EXCEEDED ()`.

Operator Action: In the case of disallowed connections, the remote client user must ask the server administrator for authorization to use the SYMAPI server. The administrator must add the host (and the optional *user_name*) information to the nethost file to authorize the client application. In the case of host file read or syntax error, make sure tht the trusted host file is readable or correct the syntax error in the file. Refer to the *Solutions Enabler Security Configuration Guide* for the syntax of the nethost file. In the case of max connection error, the server administrator may wish to set `max_sessions` to a higher value, or the client application may have to be scheduled when the server is less busy.

ANR0019E

SYMAPI client directed debugging is disabled

Destination: Log and console.

Description: The SYMAPI server initialization process attempts to prepare for client supplied debugging settings when client sessions specify them. Invocation of an internal service failed which prevents the future use of debugging settings from client applications.

This message is preceded by ANR0200E which documents the reason for the failure to setup for client debugging.

Operation Action: The output of the preceding message ANR0200E gives an indication of the type of failure that is the cause of this situation. Collect and provide documentation as directed by EMC Customer Support.

ANR0020I

SYMAPI server listening on port *port_number* over *protocols*

Destination: Log and console.

Description: This message is issued in conjunction with message ANR0001I to inform the system operator about the port (*port_numbei*) and internet protocols over which the server is communicating. Possible values for *protocols* are:

- IPv4 ONLY — Indicates that the server is listening for connections only using IPv4. Clients that expect an IPv6 connection will fail connecting to the server.
- IPv6 and IPv4 — Indicates that the server is listening explicitly for connections using IPv6 and IPv4.
- IPv6 with IPv4 mapping — Indicates that the IPv6 protocol supports connections from clients who are running either IPv4 or IPv6.

Operator Action: None.

ANR0021I

The current working directory is *directory*

Destination: Log.

Description: This message is issued early in server initialization after the server process attempts to make the SYMAPI database directory the current working directory.

Operator Action: None. This is an informational message.

ANR0022I

SYMAPI server is running on a Symmetrix Service Processor, forcing port *port*

Destination: Log.

Description: This message is written when the server detects it is running on a Symmetrix service processor. In this case, the server forces the use of the default port.

Operator Action: None. This is an informational message.

ANR0023I

SYMAPI server Symmwin Pipe Server is initialized

Destination: Log.

Description: This message is written when the special server thread to field requests from the SymmWin component has been started successfully. This will only happen if the server is running on a Symmetrix service processor.

Operator Action: None. This is an informational message.

ANR0024I

SYMAPI server Enhanced Authentication is ENABLED | DISABLED

Destination: Log and console.

Description: This message is issued during server initialization to indicate Enhanced User Authentication is enabled or disabled.

- ENABLED indicates that if a client sends an authentication message it will be verified.
- DISABLED indicates that if a client sends an authentication message it will not be verified.

Operator Action: On non-Windows hosts, if the authentication mode indicated in the message is not the mode desired, verify that the `/etc/krb5.keytab` file exists, that its permissions indicate that `storsrvd` can access it, and verify that the `klist -k` value in the file shows the correct entry for the host. If the conditions are all correct, turn on high levels of diagnostic logging to look for additional information.

ANR0025E

Rejecting session *session_number* for Host *hostname*: *max_sessions_per_host (limit)* has been reached

Destination: Log.

Description: A remote client attempts to connect to the server, and the server is tracking concurrent sessions per host using the *max_sessions_per_host* configuration option. The current session exceeds the number of concurrent sessions permitted from a specific host. Therefore the session is rejected. *limit* indicates what the current value of *max_sessions_per_host* is and *session_number* is the number of the current session. *hostname* names the host from which the session originates. It may be a simple nodename, a Fully-Qualified Domain Name, or an IP address.

Operator Action: The user of the client application must wait until the number of concurrent sessions from the specific host falls below the limit set by `max_sessions_per_host`, or the server administrator can raise the `max_sessions_per_host` value or disable concurrent user tracking using the stordaemon `setvar storsrvd -name max_sessions_per_host` command. See the *Solutions Enabler Security Configuration Guide* for details on session limits.

ANR0026E

Rejecting session *session_number* for User *user*: *max_sessions_per_user (limit)* has been reached

Destination: Log.

Description: A remote client attempted to connect to the server, and the server is tracking concurrent sessions per user using the *max_sessions_per_user* configuration option. The current session exceeds the number of concurrent sessions permitted from a specific

user. Therefore the session is rejected. *limit* indicates the current value of *max_sessions_per_user* and *session_number* is the number of the current session. *user* is the fully-qualified user name as documented in the *Solutions Enabler Security Configuration Guide*.

Operator Action: The user of the client application must wait until the number of concurrent sessions from the specific user falls below the limit set by `max_sessions_per_user`, or the server administrator can raise the `max_sessions_per_user` value or disable concurrent user tracking using the stordaemon `setvar storsrvd -name max_sessions_per_user` command. See the *Solutions Enabler Security Configuration Guide* for details on session limits.

ANR0027E

Rejecting session *session_number* for Host *hostname*: *max_sessions_per_user* is zero.

Destination: Log.

Description: A remote client attempted to connect to the server, and the server is tracking concurrent sessions. Even though `max_sessions_per_host` may not prevent this session from being initialized, the server detected that `max_sessions_per_user` is set to zero, in which case the session will be refused when the server checks the concurrent sessions allowed per user. Therefore the server rejects the session based on this early detection. *session_number* is the number of the current session, and *hostname* names the host from which the session originates. It may be a simple nodename, a Fully-Qualified Domain Name, or an IP address.

If `max_sessions_per_user` is not set to zero, the concurrent user check is made later in the process, and the session will either be accepted if the session does not exceed the limit set by `max_sessions_per_user`, or refused if it does, in which case the server returns message ANR0026E.

Operator Action: When any of the session limit options `max_sessions`, `max_sessions_per_host`, or `max_sessions_per_user` is set to 0, all new sessions attempting to connect to the host are refused. The server administrator can alter any of the options or disable concurrent host and user session tracking using the stordaemon `setvar storsrvd -name max_sessions_XXXX` command. See the *Solutions Enabler Security Configuration Guide* for details on session limits.

ANR0030E

Failed to load configuration for *name*

Set Step Return Code

Destination: Log and console.

Description: This message is issued when an error is detected in the loading of the configuration settings for the SYMAPI server daemon. The instance name is the name of the daemon for which configuration was attempted.

Operator Action: Examine the messages that precede this message. A syntax error in the configuration file section for the daemon instance *name* is the most likely cause. For example, the port definition may have specified an invalid number for the port, or an invalid security level may have been specified for the security_level option.

ANR0031E

The security_level (or -seclevel) keyword requires a security level to be specified

Set Step Return Code

Destination: Log.

Description: This `–seclevel` operand was specified without a value on the `stordaemon setvar` command line.

Operator Action: If you specify a security level, you must specify a valid value for the security level through the `stordaemon setvar` command. The valid values are NONSECURE, ANY, and SECURE. No abbreviations are accepted.

ANR0032E

The -log_filter keyword requires list of log filter types to be specified

Set Step Return Code

Destination: Log.

Description: This `–log_filter` operand was specified without a value on the `storsrvd` command line.

Operator Action: If you specify `–log_filter`, you must specify the desired list of filter types. Use the `stordaemon getvar storsrvd –name log_categorie`s for the list of appropriate filter types.

ANR0033E

The '-port' or 'storsrvd:port' keyword requires a non-zero decimal number less than 65535

Set Step Return Code

Destination: Log.

Description: An invalid value was specified for the SYMAPI server port. If the `storsrvd` command operand `–port` or the `storsrvd:port` statement is used, the value specified for the port must be a non-zero decimal number less 65535. Many port numbers in the lower ranges must also be avoided since they are used by well known processes (for example, the `inetd` and `ftpd` daemons).

Operator Action: Correct the command line or `daemon_options` file specification, and restart the server.

ANR0034I

The port is not reloaded while the server is running, bypassing any new port definition

Destination: Log.

Description: During execution of the `reload` command, a change to the port specification was detected. This message is issued to alert the administrator to the fact that the port definition cannot be changed during the reload operation.

Operator Action: In order to change the port, you must shut down the storsrvd process, make the port change, and restart `storsrvd`.

ANR0104E

Command syntax error: *explanation*

Destination: Log and console.

Description: The operator entered a command with invalid syntax explained by *explanation.*

Operator Action: Examine the syntax description for the command you want to enter, and re-enter it with the proper operands.

ANR0105E

Ambiguous or invalid command token entered: *token_text*

Destination: Log and console.

Description: The operator entered a command but either the command verb or a keyword name in *token_text* was misspelled or its abbreviation was too short to uniquely identify the intent.

Operator Action: Examine the syntax description for the command you want to enter, and re-enter it with the proper operands.

ANR0106I

Environment variable *name* has been set to *value*

Destination: Console.

Description: The operator entered the `SETENV` command, and the environment variable was successfully set.

Operator Action: None. This message provides confirmation that the variable was set as intended.

ANR0107E

option is not a valid runtime option

Destination: Log and console.

Description: The operator entered the `setvar` command, but the name of the runtime option (*option*) was not recognized as a valid option.

Operator Action: Examine the description of the `setvar` command for the supported options. Re-enter the command with the desired option. `setvar` accepts the runtime option names with or without the dash prefix.

ANR0108E

value is not a valid value for runtime option *option*

Destination: Log and console.

Description: The operator entered the `setvar` command with the name of a valid runtime option (*option*), but the value (*value*) specified for *option* was not valid.

Operator Action: Examine the description of the `setvar` command for the proper values corresponding to each supported option. Re-enter the command with the corrected value for the desired option.

ANR0110E

Invalid *option* command option name found following successful parse: decimal value is *code_value*

Destination: Console.

Description: This message indicates a programming or environmental error in command parsing and execution. The parsing of the command was successful, but the secondary scan performed by the execution phase found an invalid token.

Operator Action: Collect and provide documentation as directed by EMC Customer Support.

ANR0111I

option runtime option has been set to *value*

Destination: Log and console.

Description: The operator entered the `setvar` command to change the value of the runtime option *option.* The command text was successfully parsed, and the command was executed successfully. The new value of the variable is *value.*

Operator Action: None.

ANR0112I

command_name command requires additional operands

Destination: Console.

Description: The operator issued command *command_name* without sufficient operands. Default processing could not be established.

Operator Action: Re-enter the command with desired operands, according to the documentation. You can also use the `help` command to determine the required operands.

ANR0113I

option current value: *value*

Destination: Console.

Description: This message is issued by the `DISPLAY` or `SHOW` command for a runtime option. The *option* is the runtime option specified in the `SHOW` command, and its current setting is *value.*

Operator Action: None. The operator may issue this command before changing the value of a runtime option, or may want to confirm its value after setting it (although message ANR0111I can be used for the latter purpose).

ANR0114I

environment_variable is currently not set

Destination: Console.

Description: The operator entered the `SHOW -ENV` command to display the value of an environment variable. The variable has not been set.

Operator Action: None.

ANR0115I

environment_variable is set to an empty value

Destination: Console.

Description: The operator entered the `SHOW -ENV` command to display the value of an environment variable. The variable is set in the environment of the server, but the value is the empty string.

Operator Action: None.

ANR0116I

The *option* runtime option may not be changed while the server is running

Description: The operator or stordaemon user issued the `stordaemon setvar -name` command to change an option which cannot be changed while the server is running.

Operator Action: To change the desired option on the next run of `storsrvd`, you can use `stordaemon setoption` or edit the `daemon_options` file in the SYMAPI configuration directory. If you use the `setoption` command and then try to use `reload`, additional log messages may be issued indicating that some changed options will not be reloaded.

ANR0120I

SYMAPI Active Session List:

Destination: Console.

Description: The operator issued the `LIST SESSIONS` command and there are active sessions to list. This message is the heading for the list of sessions which follows.

Operator Action: None.

ANR0121I

No active sessions found.

Destination: Console.

Description: The operator issued the `LIST SESSIONS` command or the `SHOW SESSION` command and there are no active sessions to list/show.

Operator Action: None.

ANR0122I

Session *number* is not active

Destination: Console.

Description: The operator issued the `SHOW SESSION` command with the `–NUM` option to display a specific session, and the specified session was not active.

Operator Action: None.

ANR0123I

Show *server* Details:

Destination: Console.

Description: The operator issued the `SHOW –SERVER` command to display the details for the server. This line is written to mark the beginning of the server details output.

Operator Action: None.

ANR0124I

Show Session details for Session *session_number* on Thread *thread_number:*

Destination: Console.

Description: The operator issued the `SHOW SESSION` command to display details of one or more currently active sessions. This line is written at the beginning of the details for each session to be displayed.

Operator Action: None.

ANR0140E

Secure sessions are not supported on this platform. The security level specified is *security_level*

Set Step Return Code

Destination: Log and console.

Description: This message is issued when either the SYMAPI `options` file or `daemon_options` file specified a security level of ANY or SECURE on a platform where secure sessions are not supported. In the case of the `options` file, the SYMAPI_SERVER_SECURITY_LEVEL= statement specified this value. In the case of the `daemon_options` file, the storsrvd:security_level specified ANY or SECURE. The value may have been specified for the `–seclevel` operand of the `storsrvd` command.

Operator Action: If security level is specified through any configuration statement or `storsrvd` command operand, it must specify NONSECURE on platforms where secure sessions are not supported. It is safer to omit the specification altogether, or to specify the dash character '-'. Refer to the *EMC eLab Navigator* for a list of platforms where secure sessions are supported.

ANR0141E

Could not extract server *file* filename, rc=*returncode*

Destination: Log.

Description: During initialization, the SYMAPI server was not able to determine the name of the file to be used in SSL initialization. The string *file* refers to the SSL type file that the server was about to reference. The failing return code is displayed in *returncode*.

Operator Action: In an Open Systems environment, the server certificate and private key files should have been installed by the normal installation procedure. In z/OS and Microsoft Windows, the location of the Solutions Enabler configuration directory can be adjusted to your configuration needs. Follow the platform specific installation instructions to install the default server certificate files.

ANR0142E

function establishment failed with rc= *returncode (error_message)*

Destination: Log.

Description: During SSL initialization, the component referred to by *function* failed to be established. If *function* is CERTIFICATE or PRIVATE KEY, then the `symapisrv_cert.pem` file may be damaged or it may not have been successfully copied to the SYMAPI configuration directory.

Operator Action: If server certificate and key files are not installed by default on the platform where the server is running, additional installation steps are necessary. Refer to the platform specific installation instructions to install the files. You can specify NONSECURE for the security level if desired; in which case, the server will not attempt to load the certificate and key files.

ANR0143E

Rejected session *address*: security level mismatch reason: *error_message*

Destination: Log.

Description: A mismatch of security levels occurred when an initiating client session requested a security mode that the server was not able to honor.

address is the IP address of the client and *error_message* contains the error message indicating the actual problem.

Operator Action: If possible, modify the security level of the client to match the security mode that the server is using. If that is not possible, then (unless other clients will be impacted), modify the security level of the server to match the security level the client is requesting.

ANR0144E

Secure Library Init error: rc=*return_code (error_message)*

Destination: Log.

Description: Some component failed during SSL initialization. The *return_code* value corresponds to the message explained in the string *error_message*.

Operator Action: If you are unable to resolve the problem indicated in string *error_message*, contact EMC technical support for assistance with this error.

ANR0145E

The value *value* specified for security level is invalid

Set Step Return Code

Destination: Log.

Description: This message is issued when an attempt is made to set the security level for the SYMAPI server daemon using one of the supported methods, and the value specified is invalid. The methods to set the security level are: the `storsrvd -seclevel` command line option, the `storsrvd:security_level` statement in the `daemon_options` file, or the `stordaemon setvar` command. The valid values are NONSECURE, ANY, or SECURE. Note that a separate message (ANR0148E) is issued if an invalid value is specified in the SYMAPI options file.

Operator Action: Correct the value specified on the command line or in the `daemon_options` file, and restart the server or re-execute the `stordaemon` command.

ANR0146I

Security level has changed from *old_security_level*. New sessions will use *new_security_level*

Destination: Log.

Description: The security level to be used by the server was changed successfully using the `setvar` or `reload` command through the `stordaemon` CLI on the z/OS console. The level was changed from *old_security_level* to *new_security_level*. New sessions will negotiate based on the new security level set, but existing sessions are unaffected by the new level, and will continue to use the security level negotiated when they started.

Operator Action: Confirm that the *new_security_level* is the intended security level. If so, no further action is required. If not, you may want to refer to the server logs or other logs to determine why the security level was changed.

ANR0147I

The SYMAPI options file specified an empty value for SYMAPI_SERVER_SECURITY_LEVEL, changing to platform internal default *security_level*

Destination: Log.

Description: A configuration file statement specified an empty value for the server security level. Such a specification is an error, but the server will substitute the default security level value for the platform on which the server is running. The default value is ANY for platforms that support secure mode and NONSECURE for those platforms that do not support secure mode.

Operator Action: The omission of the security level on an explicit configuration is most likely a mistake. Refer to the SYMAPI `options` file or the `daemon_options` file to correct the omission, if you want to suppress the appearance of ANR0147I.

ANR0148E

The SYMAPI option SYMAPI_SERVER_SERCURITY_LEVEL specified an invalid value

Set Step Return Code

Destination: Log and console.

Description: This message is issued during server initialization when an invalid value is specified in the SYMAPI `options` file statement SYMAPI_SERVER_SECURITY_LEVEL. The value for security level is extracted from the SYMAPI `options` file only if no other specification was made on the command line or in the `daemon_options` file. Note that a

separate message (ANR0145E) is issued if an invalid value is specified in any of the other methods: `storsrvd` command line, `daemon_options` file, or the `stordaemon setvar` command.

Operator Action: Correct the value specified in the SYMAPI `options` file statement SYMAPI_SERVER_SECURITY_LEVEL . The valid values are NONSECURE, ANY, or SECURE.

ANR0149D

Security level has been taken from the SYMAPI options file

Destination: Log.

Description: This message is issued when the value for the server security level is defined in the SYMAPI `options` file and has not been specified on the `storsrvd` command line, nor in the `daemon_options` file. This message is informational only.

ANR0150E

The value *value* specified for client certificate verification is invalid

Destination: Log.

Description: This message is issued during server initialization when the value for the client certificate verification option, as defined in the `daemon_options` file, is invalid. It can also be issued when attempting to change this option with the `stordaemon` command to an invalid value.

Operator Action: Correct the value specified in the `daemon_options` file statement `security_clt_secure_lvl` or as specified on the command line. The valid values are NOVERIFY, VERIFY or MUSTVERIFY.

ANR0151E

Common Name in client certificate not valid: expected *name*, received *common name*

Destination: Log.

Description: This message is issued during setup of secure mode between client/server. The common name in the client certificate does not match the name the server is expecting.

Operator Action: Check the client certificate to verify that the names contained in the certificate are known hostnames to the server. Either generate a client certificate with the hostname that the server is expecting or add the common name in the client certificate to the applicable `/etc/hosts` file on the server.

ANR0152E

Issue detected with server certificate file *filename*

Destination: Log.

Description: This message is issued during initialization of the secure library. A problem with the certificate file has been detected.

Operator Action: Check for the existence of the certificate file on the server. If you have set the `security_alt_cert_file` parameter in the `daemon_options` file, verify that it points to a valid file.

ANR0153E

Issue detected with server PrivateKey file *filename*

Destination: Log.

Description: This message is issued during initialization of the secure library. A problem with the PrivateKey file has been detected.

Operator Action: Check for the existence of the `Privatekey` file on the server. If you have set the `security_alt_key_file` parameter in the `daemon_options` file, verify that it points to a valid file.

ANR0154E

Host name pattern in certificate is not valid: *pattern* for the client *Host Name*

Destination: Log.

Description: This message is issued during setup of secure mode between client/server. It indicates an illegal pattern has been put into the client certificate. *Pattern* shows the pattern in the client certificate, and *HostName* shows the name of the client host which was attempting to connect to the server.

Operator Action: Generate a new client certificate without the illegal host name pattern. The only characters allowed for a host name pattern are letters, numbers, periods (.), colons (:), and hyphens (-).

ANR0200E

service_name error *return_code: explanation;* from *calling_routine,* line *line_number*

Destination: Log.

Description: Server logic called the routine named by *service_name* and received a failure indicated by *return_code,* where *explanation* is text that corresponds to the *return_code.* The failure was detected at line *line_number* in the routine *calling_routine.* The routine *calling_routine* was not able to continue due to the failure of *service_name.*

Operator Action: None, generally. This message may occur in very rare circumstances during handling of an operator command, and may indicate a syntax error that was not handled properly by parsing logic. Examine the command and reissue it if it was specified incorrectly.

ANR0201E

Unable to allocate *count* bytes for *object_name*

Destination: Log.

Description: The server attempted to allocate *count* number of bytes. *object_name* is a description of what the server was trying to allocate. This message may indicate that the server is over-committed with regard to the number of concurrent sessions, or that there may be a memory leak in the server.

Operator Action: Increase the amount of memory available to the server using the appropriate method for the platform the server is running on. If this does not solve the problem, a memory leak may be indicated by other failure messages. Collect and forward error documentation to EMC Customer Support for analysis.

ANR0202E

Unable to *operation_name* port *port*, error *error_number* indicates *explanation*

Set Step Return Code

Destination: Log and console.

Description: An error occurred operating on the socket on which the server listens for new connections. *operation_name* will indicate an error during bind, listen, initialize, accept, or start new thread. The *error_number* is the decimal value of the system error variable ***errno*** (in Windows, the value returned from the GetLastError() call), and the *explanation* is the text that explains the meaning of *error_number. port* is the TCP/IP port which clients use to connect to the SYMAPI server. The server shuts down after issuing this message.

Operator Action: In most cases, other messages will also be issued giving other details about an error situation. Follow your normal procedures for detecting and correcting problems in your TCP/IP network. Correct the TCP/IP problem and restart the server.

ANR0204E

Unable to decode return value *return_value* from *process*

Set Step Return Code

Destination: Log and console.

Description: The *return_value* from a call to a routine or other logic could not be interpreted. *process* may be the name of the function or may be a general description of processing that resulted in a return value which could not be interpreted.

Operator Action: None. This message will be preceded by other error messages that provide more detail. If your normal processing is unaffected, no action is necessary. Otherwise, you may need to collect and provide documentation as directed by EMC Customer Support.

ANR0205E

action is not currently supported

Destination: Log and console.

Description: An action or feature was requested that is either not yet supported or is no longer supported. The name of the action or feature not supported is *action.*

Operator Action: None. The feature you requested is not available for use in this release. If you receive this message in error, examine the job log for other evidence of a failure which may be related to the action or feature you attempted to use.

ANR0207S

Failed to start *name* thread, error = *code (explanation)*

Set Step Return Code

Destination: Log and console.

Description: This message is issued in two cases:

- During server initialization, the attempt to start the dedicated SYMAPI listener thread failed. In this case, *name* is *SYMAPI Listener*. The server will immediately abort initialization and will stop.
- During handling of the arrival of a SYMAPI session, the attempt to start a dedicated thread for the session failed. In this case, *name* is *SYMAPI session*. The server continues to listen for other sessions, although the ability to start new threads can be limited. Other messages may accompany this one with additional diagnostic detail. The return code and explanation from the thread-start service call are displayed in *code* and *explanation*.

Operator Action: Examine other messages in the log files and other system output. You may be able to determine the cause and corrective action from other messages. In the second case, system resources required to start threads may be exhausted due to the current SYMAPI session count. Your system may be configured to allow a maximum number of threads per process, and this limit may have been exceeded. Complete diagnosis may require assistance of EMC technical support.

ANR0208E

Unable to verify SYMAPI Database directory *db_dir*

Set Step Return Code

Destination: Log and console.

Description: The server attempts to make the SYMAPI database directory the current directory during initialization in order to cause non-default database files to be placed in the database directory if the name is not a fully-qualified pathname. This message is issued during server initialization if the SYMAPI database directory does not exist or is inaccessible. The most common reason is that the database directory does not exist. The name of the directory the server attempted to verify is shown in *db_dir*.

Operator Action: The Solutions Enabler installation process creates the database directory normally. If this operation failed during installation, the installation process would have terminated with an error. You can create the directory using the tool appropriate to your platform. Use the directory name shown in *db_dir* in the message text.

ANR0209I

Authentication service name *service_name* exceeds maximum length

Destination: Log and console.

Description: The `storsrvd` process is attempting to copy *service_name* to an internal structure and is unable to because of its length.

Operator Action: If possible, shorten the name of the host shown in *service_name*. Otherwise, you may need to collect and provide documentation as directed by EMC Customer Support. The server will continue to operate in non-authenticated mode.

ANR0210E

EMCSAI version does not meet minimum version requirement of *nn.nn.nn*

Destination: Log and console.

Description: The version of ResourcePak running on the host does not meet the minimum version required by Solutions Enabler.

Operator Action: Ensure that Solutions Enabler is configured to work with EMC ResourcePak Base at the indicated version or later.

ANR0211E

Unable to obtain EMCSAI version, RC=%a (%b) EMCRC=%c, EMCRS=%d

Destination: Log and, in some cases, the console.

Description: This message is issued as a result of an interface error when Solutions Enabler checks the EMC ResourcePak Base version.

Where:

%a is the return code from the call to the ResourcePak Base EMCSAI interface

%b is a text description of the message.

%c is the EMCSAI Return Code (emcrc)

%d is the EMCSAI Reason Code (emcrs)

The most common cause of error is that Solutions Enabler is configured to work with a version of EMC ResourcePak Base which is not running or which does nor exist. Either one of these conditions will result in the following message being issued:

ANR0211E Unable to obtain EMCSAI version, RC=28 (Symmetrix Control Facility is not available) EMCRC=0, EMCRS=0

Operator Action: In all other cases of the message, contact EMC for support.

ANR0212E

Unable to determine peer *identifier*, System call: *callname*, RC: *return_code*

Set Step Return Code

Destination: Log and console.

Description: During session negotiation, the server attempts to look up the name of the client host which has initiated the session. If the name of the host cannot be determined, the server then attempts to look up the IP address of the client host. The *identifier* will be either "*nodename*" or "*address"* depending on which failure occurs.

Where:

callname is the name of the system function called to execute the lookup.

return_code is the failure return code from that function.

Operator Action: The session continues to be initiated, if possible. If the session is SECURE, then it's very likely that the validation of the hostname in the certificate will fail, since it is compared to the identifier obtained from the system. If the system cannot return a hostname, DNS and local host TCPIP configuration can be changed to configure a hostname properly. In the rare case that the system cannot obtain an IP address, it is an indication of a severe IP configuration problem. Your network system administrator should be consulted to determine the nature of the network configuration problem.

ANR0220I

Thread *thread_number* will execute without condition handling protection

Destination: Log.

Description: In a z/OS environment, the session on thread *thread_number* will be executed without the protection of a condition handler. The `setvar -cond_hdlr OFF` command had been previously issued, causing condition handling suppression. This message is a confirmation that the session will be run without protection. An abend on the thread will cause the operating system to terminate the server address space.

You can associate the thread number with a session number by using the `LIST SESSIONs` command. The second column of the list sessions output is the thread number of the session.

Operator Action: None.

ANR0221E

Unable to set condition handling for thread *thread_number*, msgno=*LE_message_num*, sev=*LE_severity*

Destination: Log and console.

Description: In a z/OS environment, the thread (*thread_number*) handling a session attempted to set condition handling by calling the Language Environment routine CEEHDLR but received a non-zero return value from the call. The Language Environment feedback message number is shown in *LE_message_number* and the severity of the return is shown in *LE_severity.*

Operator Action: None.

ANR0222S

Condition handler involved on thread *thread_number*; writing dump to DD *dump_location*

Destination: Log and console.

Description: In a z/OS environment, an abnormal condition was raised during the session running on *thread_number.* A dump will be written to the DD name *dump_location.* The general format of the DD name is DMP*nnnnn* where *nnnnn* is the *thread_number.*

Operator Action: Consult EMC Customer Support for directions on completing documentation to provide for analysis and correction.

ANR0223I

Dump to *dump_location* is complete; thread *thread_number* will be terminated

Destination: Log and console.

Description: In a z/OS environment, condition handling processing detected a recursive (second) entry into the condition handling routine. This may indicate an abend while attempting to handle an earlier abend.

Operator Action: None.

ANR0224S

Recursive entry to condition handler on thread *thread_number*

Destination: Log and console.

Description: In a z/OS environment, condition handling processing detected a recursive (second) entry into the condition handling routine. This may indicate an abend while attempting to handle an earlier abend.

Operator Action: None.

ANR0225E

Condition handling is not supported on this platform

Destination: Log.

Description: In a z/OS environment, language environment *condition handling* supports capturing abnormal termination of a thread without affecting other threads in the process (job). This message is issued when an attempt is made to set or display the current condition handling setting in a non-z/OS environment, using the `stordaemon getvar` or `setvar` command.

Operator Action: Correct the `setvar` or `getvar` command to specify an option which is supported in the environment where you are using the `stordaemon` command.

ANR0300E

API Request code *SYMAPI_request_code API_name* rejected; it is restricted and disabled

Destination: Log.

Description: A SYMAPI request code that describes a control operation was received. The server checked the *SYMAPI_request_code* (function named in *API_name*) to determine whether execution has been disabled. The API request was found to be disabled. This message will only be issued in the z/OS environment.

Operator Action: None. In a z/OS environment, control operations may have been disabled by using the installation job `#12CNTRL` in the Solutions Enabler RIMLIB dataset.

ANR0301I

API Request code *SYMAPI_request_code API_name* executing

Destination: Log.

Description: The server received a SYMAPI request described by the decimal code *SYMAPI_request_code.* This message is issued when the server begins executing the API request. The name of the SYMAPI function name is *API_name*.

Operator Action: None.

ANR0302I

API Request code *SYMAPI_request_code* complete, processing status *SYMAPI_return_code* (*explanation*)

Destination: Log.

Description: The API request named in message ANR0301I completed executing. The decimal code of *SYMAPI_request_code* corresponds to the API request code. The return value of the API request was *SYMAPI_return_ code,* and the corresponding text is *explanation.*

Operator Action: None.

ANR0303I

Executing SymExit to clean up (client exited without calling SymExit)

Destination: Log.

Description: The client application exited its process before calling SymExit to end the remote session with the SYMAPI server. The server calls SymExit on behalf of the client to free up resources which are still held.

Operator Action: None.

ANR0304I

Cleanup SymExit return: *return_value* (*explanation*)

Destination: Log.

Description: The cleanup call to SymExit completed, and the return value was *return_value.* The *explanation* is the text associated with *return_value.*

Operator Action: None.

ANR0305E

REMOTE_CACHED mode not supported for client node *Host_name* version *client_version_number* - connection rejected

Destination: Log.

Description: A client running a version of Solutions Enabler earlier than V7.1 attempted to connect to a SYMAPI server running V7.5, which is not allowed.

Operator Action: None.

ANR0306E

Connection rejected from client node *HostName* -- its version (*version*) is no longer supported in C/S mode

Destination: Log.

Description: Client connections using SYMAPI versions lower than V7.1 are not supported. *Hostname* is the name of the client host where the connection originated, and *version* is the version of the SYMAPI library with which the client program was built.

Operator Action: The developer of the application must upgrade to a newer version of the SYMAPI library. If the client program is the SymCLI, there may be an incorrect version installed on the client host, or the client may intend to connect to a different server.

ANR0307E

Connection rejected from client node hostname -- its version (*HostName*) is newer than our version

Destination: Log.

Description: The SYMAPI version of the client program is newer than the server's version. Such a connection is not supported. *HostName* is the name of the client host where the connection originated, and *version* is the version of the SYMAPI library with which the client program was built.

Operator Action: Insure the client program is directing its connection request to the correct server.

APPENDIX B
Asynchronous Events

This appendix lists the possible asynchronous error and message events trapped by the event daemon:

Symmetrix event codes

The descriptions in this appendix is focused on running the event daemon in a logging mode - where events are automatically forwarded to a file on disk, syslog, SNMP, or the Windows Event Service.

Events below are described in the following format:

‹Event-ID›	**‹Event-Name›**
Category	‹Event-Category›
Component	‹Event-Component›
Severity	‹Event-Severity›
Message	‹Event-Message›

Where:

‹Event-ID›	The event ID - from the SYMAPI_AEVENT2_UID_T enumeration in symapi.h
‹Event-Name›	The internal name for this event.
‹Event-Category›	The category that this event belongs to, if any. Registering against a category has the effect of registering for all events that belong to that category.
‹Event-Component›	The component, if one is known, that the event is delivered with. For Event Logging (to file, Syslog, SNMP, Windows Events), the component will only be present if a specific component (for example: a specific device, disk, pool, ...) is known.[1]
‹Event-Severity›	The severity that the event is delivered with: Fatal, Critical, Major, Minor, Warning, Info or Normal.
‹Event-Message›	The message that the event is delivered with.

1. Starting with Solution Enabler V7.4, the system ignores leading zero(es) when matching device numbers in event registrations against those in delivered events. That means if you register for events on device 01234 or 001234, events for device 1234 will be received.

Unless all events are delivered with an Entity-Name set to the Symmetrix ID that relates to the event.

Classes of Events

There are 3 general types of events:

- "Event daemon events: Event IDs 0-199" — Events in this range (there are only a handful) are generated by the event daemon itself - and reflect conditions within it.

  These are described below.

- "Symmetrix Events: Event IDs 1050 - 1199" — Events in this range correspond to entries retrieved from the 'Error' log on a Symmetrix array. Some of these are informational in nature; others correspond to actual errors.

- "Symmetrix Events: Event IDs 1200-1999" — Events in this range are manufactured by the event daemon itself based on its regular polling of conditions on a Symmetrix array.

Severity Calculation for status/state events

For a number of the Symmetrix status events, an event severity is calculated dynamically from the status of the component in question (or overall array). In most cases, the mapping to severity is as follows:

Severity	Meaning
Normal	The component is now (back) in a normal state of operation.
Info	The component is no longer present (during certain operations).
Warning	The component is in a degraded sate of operation. The storage array is no longer present (during certain operations). The component is in an unknown state. The component is (where possible) in a write-disabled state.
Major	The component is offline.
Fatal	The component is in a dead or failed state.

Event daemon events: Event IDs 0-199

Events in this range are generated by the event daemon - and reflect its internal state.

They are automatically delivered to any registered applications as needed. There is no need to explicitly register for them.

1

1	SYMAPI_AEVENT2_UID_EVT_RESTARTED
Category	
Component	
Severity	Warning
Message	event daemon restarted; events may have been lost.

Notes

Generated when the event daemon is restarted after a crash.

2

2	SYMAPI_AEVENT2_UID_EVT_EVENTS_LOST
Category	
Component	
Severity	Warning
Message	event daemon communications problem; events may have been lost.

Notes

Generated when the event daemon encounters a communication problem attempting to send events back to a client.

3

3	SYMAPI_AEVENT2_UID_EVT_EVENTS_OVERFLOW
Category	
Component	
Severity	Warning
Message	Event Queue overflow; events may have been lost.

Notes

Generated when one of the internal Event Queues (within a client process or event daemon) overflows and events are discarded.

Symmetrix Events: Event IDs 1050 - 1199

Events in this range correspond to entries retrieved from the Error log on a Symmetrix array. Some of these are informational in nature; others correspond to actual errors.

These correspond to events returned by the `symevent` SYMCLI command.

There are a number of categories that can be used to register for a related subset of these events.

- array subsystem
- db checksum
- diagnostic
- environmental
- device pool
- service processor
- srdf system

srdf link
srdfa session
srdf consistency group
director
device
disk

Symmetrix Events: Event IDs 1200-1999

Events in this range are manufactured by the event daemon itself based on its regular polling of conditions on a Symmetrix array.

There are two categories that can be used to register for a related of these events:

- status
- optimizer

1200

1200	SYMAPI_AEVENT2_UID_ALERT_DEV_STATUS	
Category	status	
Component	Device number Device =1234	
Severity	If Online If Online Degraded If Offline If Not Present	Normal Warning Major Info
Message	Device state has changed to Not Present | Online | Online Degraded | Offline	

Notes

- 'Not Present' means that the device could not be seen by Solutions Enabler.
- 'Online' means that the device service state is normal.
- ‘Online [Degraded]' means one or more of the device's mirrors are in a Not-Ready state.
- 'Offline' means that the device service state is failed.

1201

1201	SYMAPI_AEVENT2_UID_ALERT_ARRAY_STATUS	
Category	status	
Component		
Severity	If Online If Online Degraded If Offline If Not Present If Unknown	Normal Warning Major Warning or Major (depending on situation) Warning or Major (depending on situation)
Message	Array state has changed to Not Present | Unknown | Online | Online Degraded | Offline	

Notes

- This event reflects the overall state of the array - including its Disks, Directors, Ports.
- ‘Not Present' means that the array couldn't be seen by Solutions Enabler.
- ‘Online' means that the array is operational.
- ‘Online [Degraded]' means that:
  - One or more Ports are in an Offline or Write-Disabled state.
  - One or more Directors are in an Offline or Dead state.
  - Device events [1200] events are also enabled and one or more device is in a Not-Ready state.
  - Array sub-component events [1404] are also enabled and one or more are in a failed (Offline) state (Fans, Power Supplies, LCCs, MIBEs, Enclosures, etc.).
- 'Unknown' means that there was a problem communicating with the array.

1202

1202	SYMAPI_AEVENT2_UID_ALERT_DIRECTOR_STATUS	
Category	status	
Component	Director identifier For example: Director=SA-03C	
Severity	If Online If Online Degraded If Offline If Failed If Not Present	Normal Warning Major Fatal Info
Message	Director state has changed to Not Present | Online | Online Degrade] | Offline | Failed	

Notes

- 'Not Present' means the director was not seen by Solutions Enabler.

- 'Online' means that the director status is Online.
- 'Online [Degraded]' means that one or more of the director's ports were in an Offline or Write-Disabled state.
- 'Offline' means that the director status is Offline.
- 'Failed' means that the director status is Dead.

1203

1203	SYMAPI_AEVENT2_UID_ALERT_PORT_STATUS
Category	status
Component	Port identifier For example: Port=SA-03C:2 (for Port 2 on Director SA-03C)
Severity	If Online — Normal If Offline — Major If Write Disabled — Warning If Unknown — Warning If Not Present — Info
Message	Port state has changed to Not Present | Unknown | Online | Write Disabled | Offline

Indicates that the status for some Director Port has changed.

Notes

- 'Not Present' means the port was not seen.
- 'Online' means a port status of On.
- 'Offline' means a port status of Off.
- 'Write Disabled' means a port status of Write-Disabled.

1204

1204	SYMAPI_AEVENT2_UID_ALERT_DISK_STATUS
Category	status
Component	Spindle ID (Disk identifier is supported for internal disks only) For example: Disk=123 or Disk=16B:C2 (for Director 16B, DA Interface C, SCSI ID/Target 2) (internal disks only)
Severity	If Online Normal If Online Spare Normal If Online Degraded Warning If Offline Major If Offline Spare Major If Not Present Info
Message	Disk state is now ‹State› (was: ‹State›). Where State can be: Online Offline Online Spare Offline Spare Online Degraded Not Present

Notes

- 'Not Present' means that the disk could not be seen by Solutions Enabler.
- 'Online' means that one or more of the disk's Hypers are in a Ready state.
- 'Online Spare' means that the disk is a Spare and one or more of the disk’s Hypers are in a Ready state.
- 'Online [Degraded]' means that the disk can only be reached via a single Symmetrix DS controller. This disk state is for external disk only and supported with Enginuity 5876 and later.
- 'Offline' means that all of the disk's Hypers are in a Not-Ready state.
- 'Offline Spare' means that the disk is a Spare and all of the disk's Hypers are in a Not-Ready state

1205

1205	SYMAPI_AEVENT2_UID_ALERT_DEV_CONFIG_CHANGE
Category	status
Component	Device number For example: Device=1234
Severity	Info
Message	Device configuration has changed.

Indicates that the configuration of some device has changed.

Notes

- The following aspects of a device's configuration is considered by this event:
  - The base device configuration.
  - The meta configuration of the device (META_HEAD, META_MEMBER).
  - The bound-vs-bound state of a TDEV (bound vs unbound).
  - Whether a dynamic spare disk is invoked for the device.
  - The RDF mode of the device (of either leg for Concurrent SRDF).
  - The data pool bound to by a TDEV changes. This reflects a device being bound, unbound or re-bound to a different pool, and is also triggered when the name of the pool changes.

1206

1206	SYMAPI_AEVENT2_UID_ALERT_POOL_STATUS	
Category	status	
Component	Pool name For example: SnapPool=Sales, DSEPool=Finance, TPDataPool=Eng	
Severity	If Online If Online Degraded If Offline If Not Present	Normal Warning Major Info
Message	Snap Savedev Pool state has changed to Not Present | Online | Online Degraded | Offline SRDF/A DSE Pool state has changed to Not Present | Online | Online Degraded | Offline Data Pool state has changed to Not Present | Online | Online Degraded | Offline	

Indicates that the status of a Snap, SRDF/A DSE or ThinData Pool has changed.

Notes

- 'Not Present' means that the pool no longer exists.
- 'Online' means that the pool is in an enabled.'
- 'Online [Degraded]' means that the pool is in a mixed state.
- 'Offline' means that the pool is in a disabled state.

1207

1207	SYMAPI_AEVENT2_UID_ALERT_POOL_CONFIG_CHANGE
Category	status
Component	Pool name For example: SnapPool=Sales, DSEPool=Finance, TPDataPool=Eng
Severity	Info
Message	Snap Savedev Pool configuration has changed. SRDF/A DSE Pool configuration has changed. Data Pool configuration has changed.

Indicates that the configuration of a Snap, SRDF/A DSE or ThinData Pool has changed.

Notes

- A pool's configuration changes if:
  - The set of Enabled devices in the pool changes.
  - The total size (free + used) of all the Enabled devices in the pool changes.

1208

1208	SYMAPI_AEVENT2_UID_THRESH_POOL_FREESPACE
Category	status
Component	Pool name For example: SnapPool=Sales, DSEPool=Finance, TPDataPool=Eng
Severity	Determined by Threshold values. See below.
Message	Snap Savedev Pool utilization is now ‹NN› percent. SRDF/A DSE Pool utilization is now ‹NN› percent. Data Pool utilization is now ‹NN› percent.

This is a Threshold event that reflects the amount (as a percentage) of used space within a pool.

Unless threshold values are supplied with the registration, the following defaults are used to derive an event severity:

If value is 100% — Fatal

If value is >= 80% — Critical

If value is >= 70% — Major

If value is >= 65% — Minor

If value is >= 60 — Warning

Otherwise — Nornal

Notes

- Used space size is determined by calls to SymPoolShow().
- Events are only delivered at multiples of 5% ... for ‹NN› equal to 5%, 10%, 15%, ... , 75%, 80%, 85%, 90%, 95% and 100%.
- Threshold events are only delivered when the severity, as determined by threshold values, changes.

1209

1209	SYMAPI_AEVENT2_UID_ALERT_SEL_CHANGE
Category	status
Component	Symmetrix Lock Number For example: SEL=15
Severity	Info
Message	Symmetrix External Lock has been acquired. Symmetrix External Lock has been released.

Indicates that the state (released vs acquired) of one of the monitored Symmetrix External Locks (SELs) has changed.

Note

At this time, only SEL #15 (used by Config Change) is monitored.

1210

1210	SYMAPI_AEVENT2_UID_ALERT_HOTSPARE_CHANGE
Category	status
Component	Disk identifier of the Spare For example: Disk=16B:C2 (for Director 16B, DA InterfaceC, SCSI ID/Target 2)
Severity	For 5x74 and newer arrays: Normal For older arrays: If invoked Warning If no longer invoked Normal
Message	For 5x74 and newer arrays: Disk is lo longer a Spare. Disk is now a Spare. Disk is now an invoked Spare. For older arrays: Spare has been invoked against a failed disk. Spare is no longer invoked against a failed disk.

Indicates that a disk has started or stopped acting as a spare.

Note

With Permanent Sparing on newer arrays, a failing disk and a spare will exchange roles. The failed disk will end up as a failed spare, and the spare will end up as a normal disk. The “Disk is now an invoked Spare” event will rarely if ever be delivered.

1211

1211	SYMAPI_AEVENT2_UID_ALERT_NUM_HOTSPARES_T
Category	
Component	
Severity	Determined by Threshold values. See below.
Message	Number of available disk spares is ‹NN›.

This is a Threshold event that reflects the number of available Spare Disks on the Symmetrix array.

Unless threshold values are supplied with the registration, the following defaults are used to derive an event severity:

If value is 0 — Critical

If value is 1 — Major

If value is 2 — Warning

Otherwise — Info

Note

Threshold events are only delivered when the severity, as determined by threshold values, changes.

1212

1212	SYMAPI_AEVENT2_UID_THRESH_TDEV_ALLOCATED
Category	
Component	Device number For example: Device=1234
Severity	Determined by Threshold values. See below.
Message	Thin Device is now ‹NN› percent allocated.

This is a Threshold event that reflects the amount (as a percentage) of a Thin Device that is backed by space in a Data Pool.

Unless threshold values are supplied with the registration, the following defaults are used to derive an event severity:

If value is 100% — Fatal

If value is ›= 80% — Critical

If value is ›= 70% — Major

If value is ›= 65% — Minor

If value is ›= 60 — Warning

Otherwise — Nornal

Notes

- Events are only delivered at multiples of 5% ... for ‹NN› equal to 5%, 10%, 15%, ... , 75%, 80%, 85%, 90%, 95% and 100%.
- Threshold events are only delivered when the severity, as determined by threshold values, changes.

1213

1213	SYMAPI_AEVENT2_UID_THRESH_TDEV_USED
Category	
Component	Device number For example: Device=1234
Severity	Determined by Threshold values. See below.
Message	Thin Device is now ‹NN› percent allocated.

This is a Threshold event that reflects the amount (as a percentage) of a Thin Device that has been written to

Unless threshold values are supplied with the registration, the following defaults are used to derive an event severity:

If value is 100% — Fatal

If value is ›= 80% — Critical

If value is ›= 70% — Major

If value is ›= 65% — Minor

If value is ›= 60 — Warning

Otherwise — Nornal

Notes

- Events are only delivered at multiples of 5% ... for ‹NN› equal to 5%, 10%, 15%, ... , 75%, 80%, 85%, 90%, 95% and 100%.
- Threshold events are only delivered when the severity, as determined by threshold values, changes.

1215

1215	SYMAPI_AEVENT2_UID_ALERT_PORT_CONFIG_CHANGE
Category	status
Component	Port For example: Port=SA-03C:2(for Port 2 on Director SA-03C)
Severity	Info
Message	Port configuration has changed.

Indicates that the configuration changes for a Port on a Front End (FE) Director.

Notes

- At this time, the only aspects of a port's configuration that are considered the following flags from the FA port flags:
- The _VCM_ENABLED flag.
- The _VOL_SET_ADDR (VSA) flag.

1216

1216	SYMAPI_AEVENT2_UID_ALERT_POOL_DEV_STATE_CHANGE
Category	status
Component	Pool name For example: SnapPool=Sales , DSEPool=Finance , TPDataPool=Eng
Severity	Info
Message	Snap Savedev Pool device state has changed. SRDF/A DSE Pool device state has changed. Data Pool device state has changed.

Indicates that the state of a device in a Snap, SRDF/A DSE or ThinData Pool has changed.

1217

1217	SYMAPI_AEVENT2_UID_DFR_SVC_STATE	
Category	status	
Component		
Severity	If the replacement threshold has been exceeded Otherwise	Warning Info
Message	The deferred services replacement threshold has been exceeded - service is required. The deferred services replacement threshold is no longer exceeded.	

Indicates that the Deferred Service replacement threshold indicator for a Symmetrix has changed. This change can be in either direction – from not-exceeded to exceeded … or from exceeded to not-exceeded.

Note

- This event will only be generated if Deferred Service is enabled for the Symmetrix array.

1218

1218	SYMAPI_AEVENT2_UID_DEV_CFG_CHKSUM
Category	status
Component	Device number For example: Device=1234
Severity	Info
Message	The device configuration checksum has changed.

Indicates that the configuration of a device has changed. The implementation makes use of a checksum maintained by the event daemonover a device's core configuration data. An event is generated when this checksum changes.

1219

1219	SYMAPI_AEVENT2_UID_MIGRATE_COMPLETE
Category	status
Component	Migrate Session name For example: MigrSess=jones17
Severity	If success Info If terminated Info If timed out Warning If failed Major
Message	The migrate operation is complete: success. The migrate operation is complete: timed out. The migrate operation is complete: terminated. The migrate operation is complete: failed.

Indicates that a VLUN migration has completed or failed.

Note

- This is only generated for explicitly initiated VLUN migrations – not movements being performed by FAST.

1220

1220	SYMAPI_AEVENT2_UID_POOL_REBAL_COMPLETE
Category	status
Component	The Data Pool name. For example: TPDataPool=Eng
Severity	If success Info If terminated Info If timed out Warning If failed Major
Message	Thin Pool rebalancing operation is complete: success. Thin Pool rebalancing operation is complete: timed out. Thin Pool rebalancing operation is complete: terminated. Thin Pool rebalancing operation is complete: failed.

Indicates that a Thin Pool rebalancing activity has completed.

Note

This event is only supported for Symmetrix arrays running Enginuity 5875.

1230

1230	SYMAPI_AEVENT2_UID_ALERT_ARRAY_CONFIG_CHANGE
Category	status
Component	
Severity	Info
Message	Array configuration has changed.

Indicates that some change has been made to the configuration of the Symmetrix array.

Note

This event is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1231

1231	SYMAPI_AEVENT2_UID_ALERT_MASKING_CHANGE
Category	status
Component	
Severity	Info
Message	Device Masking database has changed.

Indicates that some change have been made to the device masking database on the Symmetrix array.

Note

This event is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1232

1232	SYMAPI_AEVENT2_UID_ALERT_ACCESS_CONTROL_CHANGE
Category	status
Component	
Severity	Info
Message	Access Control definitions have changed.

Indicates that some change has been made to the Access Control [symacl] database on the Symmetrix array.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1233

1233	SYMAPI_AEVENT2_UID_ALERT_DYNAMIC_RDF_CONFIG
Category	status
Component	
Severity	Info
Message	Dynamic RDF operation performed on device.

Indicates that a dynamic RDF operation has been performed on some device.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1234

1234	SYMAPI_AEVENT2_UID_ALERT_SNAP_CLONE_CONFIG
Category	status
Component	
Severity	Info
Message	Snap session created, activated or deleted.

Indicates that a snap / clone session has been created, activated or deleted.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1235

1235	SYMAPI_AEVENT2_UID_ALERT_BCV_CONTROL_CONFIG
Category	status
Component	
Severity	Info
Message	BCV device pairing has changed.

Indicates that the BCV pairing for some device has changed.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1236

1236	SYMAPI_AEVENT2_UID_ALERT_DEV_NAME_HP_ID_CONFIG
Category	status
Component	
Severity	Info
Message	HPUX device identifier has changed.

Indicates that the HPUX device identifier for some device has been changed.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1237

1237	SYMAPI_AEVENT2_UID_ALERT_DEV_NAME_CONFIG
Category	status
Component	
Severity	Info
Message	Device Name has changed.

Indicates that the device name for some device has been changed.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1238

1238	SYMAPI_AEVENT2_UID_ALERT_DEV_NICE_NAME_CONFIG
Category	status
Component	
Severity	Info
Message	Device Nice Name has changed.

Indicates that the device nice name for some device has been changed.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1239

1239	SYMAPI_AEVENT2_UID_ALERT_DEV_NAME_VMS_ID_CONFIG
Category	status
Component	
Severity	Info
Message	OpenVMS device identifier has changed.

Indicates that the OpenVMS device identifier for some device has been changed.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1240

1240	SYMAPI_AEVENT2_UID_DEVICE_RESV_CHANGE
Category	
Component	
Severity	Info
Message	Device Reservations data has changed.

Indicates that the Device Reservation state for some device on the Symmetrix has changed.

Note

This is a fairly expensive event to implement - since it requires checking for modifications to file(s) within SFS.

1241

1241	SYMAPI_AEVENT2_UID_SRDFA_CYCLE_TIME_T
Category	
Component	The SRDF Group. For example: SRDF-grp=13
Severity	Determined by Threshold values. See below.
Message	Time since last SRDFA cycle switch exceeds minimum cycle time by ‹NN› seconds.

This ia a Threshold event that indicates the amount (in seconds) by which an SRDFA Group's Cycle Time exceeds the minimum that is configured.

Unless threshold values are supplied with the registration, the following defaults are used to derive an event severity:

If value is ›= 5 — Warning

Otherwise — Info

Notes

This is determined by calling SymReplicationGet() and examining the time_since_last_switch and duration_of_last_cycle quantities for Active, R1, non-MSC sessions.

The event value corresponds to the number of seconds that the larger of these two is beyond the configured min_cycle_time. If the time(s) are less than min_cycle_time (everything normal), the event value is 0. To protect against rounding problems, the test is actually against min_cycle_time+1. If the times are less than min_cycle_time+1, the event value will be 0. Therefore, possible event values are: 0, 2, 3, 4, 5, etc.

For example, assuming a min_cycle_time of 10:

time_since_last_switch	event value
9	0
10	0
11	0
13	3

1242

1242	SYMAPI_AEVENT2_UID_SRDFA_WP_CACHEUSE_T
Category	
Component	
Severity	Determined by Threshold values. See below.
Message	SRDFA cycles now using ‹NN› percent of the cache available for it.

This is a Threshold event that indicates the percentage of cache that is available for SRDFA use that is actually holding SRDFA Write Pending data.

Unless threshold values are supplied with the registration, the following defaults are used to derive an event severity:

If value is ›= 90%— Warning

Otherwise — Info

Notes

- This is determined by calling SymReplicationGet() and summing the active_cycle_size and inactive_cycle_size values for all active R1 or R2 sessions. The maximum available cache is computed in the usual manner:

```
if ((max_host_throttle == 0) and
    (rdfa_max_cache_usage > 0) and
    (rdfa_max_cache_usage < 100))
   max_avail = (max_wr_pend_slots * rdfa_max_cache_usage) / 100
else
   max_avail = max_wr_pend_slots
```

The event value is the sun of the active and inactive cycle sizes expressed as a percentage of this max avail cache size.

- **warning**: Exercise caution when assigning significance to this event. The fact that an amount of cache is available for SRDFA to use (max_avail above) doesn't mean that it is guaranteed to be available for its use. There are other sources of Write Pending data that can use up this space as well - leaving it unavailable for SRDFA's use.

1243

1243	SYMAPI_AEVENT2_UID_WP_CACHEUSE_T
Category	
Component	
Severity	Determined by Threshold values. See below.
Message	Write Pending data is now using ‹NN› percent of the cache.

Notes

This is a Threshold event that indicates the percentage of Symmetrix Cache that is holding Write Pending data.

Unless threshold values are supplied with the registration, the following defaults are used to derive an event severity:

If value is ›= 90% — Warning

Otherwise — Info

1244

1244	SYMAPI_AEVENT2_UID_ALERT_ARR_COMP_STATUS
Category	
Component	Power=xxx Fan=xxx LCC=xxx Enclosure=xxx MM=xxx IOMC=xxx Dir=xxx
Severity	If Online Normal If Online Degraded Warning If Offline Major If Unknown Warning
Message	Component state has changed to Online | Online Degraded | Offline | Unknown

Notes

Indicates a change in environmental status for one of the following types of sub-components within the Symmetrix:

Fans	[Fan]
Power Supplies	[Power]
Link Control Cards	[LCC]
Management Modules	[MM]
IO Module Carriers	[IOMC]
Directors (for environmental alerts)	[Dir]
Enclosures or Matrix Interface Board Enclosures	[Enclosure]

- 'Online' means that the component is a Normal or Degraded state.
- 'Online [Degraded]' means that the component is in a degraded state.
- 'Offline' means that the component is in a Failed state.

The format of the specific component name ('xxx' above) may differ depending on the Symmetrix model. Some examples you might encounter are:

SB-1/Fan-A	Fan in System Bay
SB-1/ENC-1	Enclosure within System Bay
SB-1/ENC-1/Fan-A	Fan in Enclosure-Slot within System Bay
SB-1/MIBE-L-2A	MIBE within System Bay
SB-1/MIBE-L-2A/PS-A	Power Supply in MIBE within System Bay
DB-1/PS-A	Component in Drive Bay
DB-1/ENC-2	Enclosure within Drive Bay
DB-1/ENC-2/LCC-B	Component in Enclosure within Drive Bay

1245

1245	SYMAPI_AEVENT2_UID_DSE_SPILL_TIME_T
Category	
Component	The SRDF Group. For example: SRDF-grp=13
Severity	Determined by Threshold values. See below.
Message	DSE Spillover has been occurring on the RDF group for ‹N› minutes. or DSE Spillover is no longer occurring on the RDF group.

This is a Threshold event that indicates the amount of time (in minutes) that SRDF DSE Spillover has been occurring for.

Unless threshold values are supplied with the registration, the following defaults are used to derive an event severity:

If value is ›= 30 (minutes) — Warning

Otherwise — Nornal

Note

Threshold events are only delivered when the severity, as determined by threshold values, changes.

1246

1246	SYMAPI_AEVENT2_UID_ALERT_DISK_GRP_CHG
Category	
Component	The Symmetrix Disk group number (decimal). For example: DiskGrp=2
Severity	INFO
Message	Disk Group has changed. or Disk Group has been deleted or Disk Group has been created

Note

This event is only supported on Symmetrix arrays running Enginuity 5876 and later.

1247

1247	SYMAPI_AEVENT2_UID_ALERT_DISK_SPARE_CVG
Category	
Component	Disk identifier For example: Disk=16B:C2 (for Director 16B, DA InterfaceC, SCSI ID/Target 2)
Severity	INFO
Message	Disk has spare coverage. or Disk no longer has spare coverage.

Note

This event is only supported on Symmetrix arrays running Enginuity 5876 and later.

1280

1280	SYMAPI_AEVENT2_UID_ALERT_CACHE_PART_CHANGE
Category	
Component	
Severity	Info
Message	Cache Partitioning configuration has changed.

Indicates that the Cache Partitioning data on the Symmetrix has been changed.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1281

1281	SYMAPI_AEVENT2_UID_ALERT_DYNAMIC_MAPPING_CHANGE
Category	
Component	
Severity	Info
Message	Dynamic Mapping configuration for a device has changed.

Indicates that the Dynamic Mapping info for some device has been changed on the Symmetrix array.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1282

1282	SYMAPI_AEVENT2_UID_ALERT_META_CONFIG_CHANGE
Category	
Component	
Severity	Info
Message	Meta configuration for a device has changed.

Indicates that the Meta configuration for some device has been changed on the Symmetrix array.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1283

1283	SYMAPI_AEVENT2_UID_ALERT_INITIATOR_GRP_CHANGE
Category	
Component	
Severity	Info
Message	Initiator Group has changed.

Indicates that some Initiator Group on the Symmetrix has been changed.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1284

1284	SYMAPI_AEVENT2_UID_ALERT_STORAGE_GRP_CHANGE
Category	
Component	
Severity	Info
Message	Storage Group has changed.

Indicates that some Storage Group on the Symmetrix has been changed.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1285

1285	SYMAPI_AEVENT2_UID_ALERT_DIR_PORT_GRP_CHANGE
Category	
Component	
Severity	Info
Message	Director Port Group has changed.

Indicates that some Director Port Group on the Symmetrix has been changed.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1286

1286	SYMAPI_AEVENT2_UID_ALERT_MASKING_VIEW_CHANGE
Category	
Component	
Severity	Info
Message	Masking View has changed.

Indicates that some Masking View on the Symmetrix has been changed.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1287

1287	SYMAPI_AEVENT2_UID_ALERT_FEAT_REG_CHANGE
Category	
Component	
Severity	Info
Message	Feature Registration DB has changed.

Indicates that a change has been made to the Feature Registration DataBase on the Symmetrix array.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1288

1288	SYMAPI_AEVENT2_UID_ALERT_APP_REG_CHANGE
Category	
Component	
Severity	Info
Message	Application Registration DB has changed.

Indicates that a change has been made to the Application Registration DataBase on the Symmetrix array.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1289

1289	SYMAPI_AEVENT2_UID_ALERT_TIERS_CHANGE
Category	
Component	
Severity	Info
Message	FAST tiers have changed.

Indicates that a change has been made to the FAST (Fully Automated Storage Tiering) Tiers on the Symmetrix array.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1290

1290	SYMAPI_AEVENT2_UID_ALERT_FAST_POLICY_CHANGE
Category	
Component	
Severity	Info
Message	FAST policies have changed.

Indicates that a change has been made to the FAST (Fully Automated Storage Tiering) Policies on the Symmetrix array.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1291

1291	SYMAPI_AEVENT2_UID_ALERT_FAST_ASSOCS_CHANGE
Category	
Component	
Severity	Info
Message	FAST associations have changed.

Indicates that a change has been made to the FAST (Fully Automated Storage Tiering) Associations on the Symmetrix array.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1292

1292	SYMAPI_AEVENT2_UID_ALERT_FAST_TIME_WDS_CHANGE
Category	
Component	
Severity	Info
Message	Optimizer/FAST time windows have changed.

Indicates that a change has been made to the FAST (Fully Automated Storage Tiering) time windows on the Symmetrix array.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1293

1293	SYMAPI_AEVENT2_UID_ALERT_FAST_CTL_PARMS_CHANGE
Category	
Component	
Severity	Info
Message	Optimizer/FAST control parameters have changed.

Indicates that a change has been made to the FAST (Fully Automated Storage Tiering) control parameters on the Symmetrix array.

Note

This is derived from one of the QuickConfig indication maintained on the Symmetrix array.

1400

1400	SYMAPI_AEVENT2_UID_AUTHZ_RULES_CHANGED
Category	
Component	
Severity	Info
Message	User Authorization rules have changed.

Indicates that a change has been made to the User Authorization [symauth] database on the Symmetrix array.

Note

This is determined by checking for modifications to the User Authorization file stored in SFS.

1401

1401	SYMAPI_AEVENT2_UID_AUDIT_LOG_SIZE_T
Category	
Component	
Severity	Determined by Threshold values. See below.
Message	Audit log is at ‹NN› percent of capacity (before wrapping).

This is a threshold event that tracks as a percentage the amount of data in a Symmetrix Audit Log - how close the log is to its *wrapping* point where existing entries begin to be over-written.

Unless threshold values are supplied with the registration, the following defaults are used to derive an event severity:

If value is ›= 80% — Warning

Otherwise — Nornal

Notes

- What is actually reported is the position of the write pointer within the Audit Log as a percentage: 0% for the beginning, 100% for the end.
- This event is intended to be used as an indication that a backup of the Audit Log is needed - if appropriate.

1402

1402	SYMAPI_AEVENT2_UID_ALERT_SEC_AUDIT
Category	
Component	
Severity	Info
Message	<< The actual message from the Audit Record >>

Indicates that a Security related record was written to the Symmetrix Audit Log.

Notes

- This event is delivered when audit records with an Audit Class of SECURITY are detected in the Audit Log.
- The audit message is a free-form string that may span multiple lines (containing multiple new line characters).

1403

1403	SYMAPI_AEVENT2_UID_ALERT_SEC_FAIL_AUDIT
Category	
Component	
Severity	Info
Message	<< The actual message from the Audit Record >>

Indicates that a Security alert was written to the Symmetrix Audit Log.

Notes

- This event is delivered when audit records corresponding to one of the following are detected in the Audit Log:
- Access Control failures (host based access control, symacl).
- User Authorization failures (user based access control, symauth).
- SymmWin / SSC Logon failures.
- SymmWin Logins
- iSCSI authorization failures
- The audit message is a free-form string that may span multiple lines (containing multiple new line characters).

1404

1404	SYMAPI_AEVENT2_UID_ALERT_ALL_AUDIT
Category	
Component	
Severity	Info
Message	<< The actual message from the Audit Record >>.

Indicates some (any) record written to the Symmetrix Audit Log.

Note

The audit message is a free-form string that may span multiple lines (containing multiple new line characters).

1500

1500	SYMAPI_AEVENT2_UID_ALERT_OPTMZ_SWAP_ACT
Category	Optimizer
Component	
Severity	Info
Message	Optimizer Swap activity (from Audit Log).

Indicates some Optimizer Swap activity.

Note

This is derived by detecting a record written by the Optimizer to the Symmetrix Audit Log:

1501

1501	SYMAPI_AEVENT2_UID_ALERT_OPTMZ_MOVE_ACT
Category	Optimizer
Component	
Severity	Info
Message	Optimizer Move activity (from Audit Log).

Indicates some Optimizer Move activity.

Note

This is derived by detecting a record written by the Optimizer to the Symmetrix Audit Log:

1502

1502	SYMAPI_AEVENT2_UID_ALERT_OPTMZ_SCHEDULE
Category	Optimizer
Component	
Severity	Info
Message	Optimizer configuration change (from Audit Log).

Indicates some Optimizer configuration change.

Note

This is derived by detecting a record written by the Optimizer to the Symmetrix Audit Log:

1503

1503	SYMAPI_AEVENT2_UID_ALERT_FAST_SWAP_ACT
Category	Optimizer
Component	
Severity	Info
Message	FAST Controller Swap activity (from Audit Log).

Indicates some FAST Controller activity.

Note

This is derived by detecting a record written by the Optimizer to the Symmetrix Audit Log:

1504

1504	SYMAPI_AEVENT2_UID_ALERT_FAST_MOVE_ACT
Category	Optimizer
Component	
Severity	Info
Message	FAST Controller Move activity (from Audit Log).

Indicates some FAST Controller Move activity.

Note

This is derived by detecting a record written by the Optimizer to the Symmetrix Audit Log:

1505

1505	SYMAPI_AEVENT2_UID_ALERT_FAST_SCHEDULE
Category	Optimizer
Component	
Severity	Info
Message	FAST Controller configuration change (from Audit Log).

Indicates some FAST Controller configuration change.

Note

This is derived by detecting a record written by the Optimizer to the Symmetrix Audit Log:

1506

1506	SYMAPI_AEVENT2_UID_ALERT_OPTMZ_RB_ACT
Category	Optimizer
Component	
Severity	Info
Message	Optimizer Rollback activity (from Audit Log).

Note

This is derived by detecting a record written by the Optimizer to the Symmetrix Audit Log:

1507

1507	SYMAPI_AEVENT2_UID_ALERT_OPTMZ_APPRVL_NEEDED
Category	
Component	
Severity	Info
Message	User approval is required for a Config Change plan generated by the Optimizer/FAST Controller.

Note

Indicates that user approval of the a swap state is required and user approval is required.

1508

1508	SYMAPI_AEVENT2_UID_ALERT_FAST_STATE_SWITCH
Category	Optimizer
Component	
Severity	Info
Message	The FAST (DP or VP) controller has switched to state: *‹current_state›* (was: *‹previous_state›*). Where ‹current_state› and ‹previous_state› can be one of the following possible values: • Disabled • Enabled • Disable in progress • Enable in progress • Disable with error • Degraded

Note

Indicates that the FAST state has changed from a previous state to the current state.

1509

1509	SYMAPI_AEVENT2_UID_ALERT_OPTMZ_MODE_SWITCH
Category	Optimizer
Component	
Severity	Info
Message	The Optimizer has switched to a different mode.

Note

Indicates that the Optimizer status state has changed.

1510

1510	SYMAPI_AEVENT2_UID_ALERT_FAST_ALLOC_CHANGE
Category	Optimizer
Component	‹policy_name›
Severity	Info
Message	The combined allocation in pools has changed.

Note

This event checks for allocated capacity change of all associated pools under the same FAST VP policy. And as such, if FAST DP policy is accidently used, this event will never be generated.

1511

1511	SYMAPI_AEVENT2_UID_ALERT_FAST_TIER_PERF_CHANGE
Category	Optimizer
Component	‹tier_name›
Severity	Info
Message	FAST Tier (‹tier_name›) is performing as expected (NORMAL). or FAST Tier (‹tier_name›) is performing worse than expected (LOW).

Note

This event is only supported with Enginuity 5876 Q42012 SR and above.

1600

1600	SYMAPI_AEVENT2_UID_GROUP_CONFIG
Category	
Entity	Not set -- set to NULL.
Component	DG or CG group. For example: DG=prod17 or CG=prod18
Severity	Info
Message	Group has changed.

Indicates that the composition of a DG or CG has changed.

Notes

- The Entity name and type (normally a Symmetrix ID) are not provided for this event. When registering to receive the event, there is no need to supply an ID (symid=000194900123) - if one is supplied, it will be ignored.
- If GNS is not enabled, this event indicates that a group definition in the Solutions Enabler DB file on this host has changed.
- If GNS is enabled, this event indicates that a global group definition stored within GNS (on Symmetrix arrays) has changed.

APPENDIX C
UNIX Native Installation Support

This appendix 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.

❑ 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.

❑ 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.7.5.0.0.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.7.5.0.0.bff.sig

❑ 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 (x86, ia64, PPC64, and 390)
- Solaris (SunOS Sparc and SunOS x86)

The kits use the following naming convention:

se*MmPp-OS-ARCH*-ni.tar.gz

Where:

M = Major version

m = Minor version

P = Point

p = Patch

OS = Operating System

ARCH = Processor architecture

For example:

```
se7500-SunOS-sparc-ni.tar.gz
```

Table 33 on page 243 lists the kit components by operating system.

Note: Table 33 on page 243:
N/A indicates that the component is not supported in the corresponding operating system.

Components within shaded rows are required.

Table 33 Solutions Enabler PureNative kit contents (page 1 of 2)

OS-specific component names				
AIX	**HP-UX**	**Linux**	**SunOS**	**Description**
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.

Table 33 Solutions Enabler PureNative kit contents (continued) (page 2 of 2)

OS-specific component names				
AIX	**HP-UX**	**Linux**	**SunOS**	**Description**
SYMCLI.BASE.rte	SYMCLI.BASE	symcli-base	N/A	Installs: • Solutions Enabler core functionality, 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 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.
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:

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 33 on page 243.
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:

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 33 on page 243.
3. Repeat step 2 for each component to install.

Using a response file

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

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, a response file, or Yum.

Using RPM

To install on a Linux host using the command line:

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

```
rpm -i se750*-Linux-*.rpm
```

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

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

Using a response file

To install on a Linux host using a response file:

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 se750*-Linux-*.rpm
```

3. Run the following command to verify the installation:

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

Using Yum

To install on a Linux host using Yum:

1. Run the following command to create a directory for the Solutions Enabler repository:

```
mkdir /symapi.repo
```

2. Change directory to the Solutions Enabler repository:

   ```
   cd /symapi.repo
   ```

3. Depending on whether the kit is in the form of a tar ball or an RPM, run the following command to extract all files into the Solutions Enabler repository:

   - If in a tar ball, run:

     ```
     tar -xvf se750*-Linux-*.tar
     ```

   - If in an RPM, run:

     ```
     rpm2cpio se750*-Linux-*.rpm | cpio -id
     mv kit_arch_dir/*.rpm current_working_dir
     rm -rf kit_arch_dir
     ```

4. Verify that the rpm files (components) and an XML file are extracted into the `/symapi.repo` directory. For file names and descriptions, refer to Table 33 on page 243.

5. Run the following command to create Yum Solutions Enabler repository:

   ```
   createrepo -g symapi.xml /symapi.repo
   ```

6. Run the following command to add the Solutions Enabler repository into the Yum repositories:

   ```
   cat > /etc/yum.repos.d/symapi.repo << EOF
   [symapi]
   baseurl=file:///symapi.repo
   enabled=1
   gpgcheck=0
   EOF
   ```

7. Run the following command to start the installation:

   ```
   yum groupinstall SYMAPI -y
   ```

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:

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 33 on page 243.

   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:

   SYMdse
   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:

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 33 on page 243.

5. Install the components in the following order:

   data
   cert
   thincore
   base
   symcli
   symrecover
   srm
   64bit
   jni

   Note: For component descriptions, refer to Table 33 on page 243.

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 33 on page 243.

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 33 on page 243.

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 33 on page 243.

APPENDIX D
Host Issues

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 chanaged 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, a rule of thumb is to allow 16 threads per locally attached Symmetrix 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 V11.0 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 Symmetrix Array Management CLI Product Guide*.

To create pseudo-devices and specify devices as gatekeepers and BCV devices:

1. Execute the `ioscan` command and find the full pathnames of the gatekeeper and BCV devices.

   For example, the full pathname of the Symmetrix 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
       75          -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 75 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.

   `75` 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 Symmetrix 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 in this release of Solutions Enabler and will fail with the following error:

```
# swverify SYMCLI:/opt/emc
=======  09/27/12 03:38:44 EDT  BEGIN verify AGENT SESSION (pid=29292)
         (jobid=hostname-4939)

       * Agent session started for user "root@hostname.company.com".
         (pid=29292)

       * Beginning Analysis Phase.
       * Target:           hostname:/
       * Target logfile:   hostname:/var/adm/sw/swagent.log
       * Reading source for file information.
       *     Configured    SYMCLI.64BIT,l=/opt/emc,r=V7.5.0.0
       *     Configured    SYMCLI.BASE,l=/opt/emc,r=V7.5.0.0
       *     Configured    SYMCLI.SYMCLI,l=/opt/emc,r=V7.5.0.0
       *     Configured    SYMCLI.SYMRECOVER,l=/opt/emc,r=V7.5.0.0
       *     Configured    SYMCLI.THINCORE,l=/opt/emc,r=V7.5.0.0
ERROR:   File "/opt/emc/SYMCLI/PERL/unzip" should have mode "555" but
         the actual mode is "755".
ERROR:   File "/opt/emc/usr/lib/libemc_crypto64.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libemc_crypto64.sl.0.9.8" missing.
ERROR:   File "/opt/emc/usr/lib/libemc_ssl64.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libemc_ssl64.sl.0.9.8" missing.
ERROR:   File "/opt/emc/usr/lib/libemcmcl.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libemcslc.sl" missing.
ERROR:   File "/opt/emc/usr/lib/liboslevtd64mt.sl" missing.
```

```
ERROR:   File "/opt/emc/usr/lib/libsapacosprep_emc.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libsnmpevtd64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorapi64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorbase64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorcore64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorctrl64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstormap64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorpds64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorsil64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorssl64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libsymapi64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libsymevtd64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libsymlvm64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libemc_crypto64.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libemc_crypto64.sl.0.9.8" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libemc_ssl64.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libemc_ssl64.sl.0.9.8" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libemcmcl.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libemcslc.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/liboslevtd64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libsnmpevtd64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libstorapi64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libstorbase64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libstorcore64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libstorctrl64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libstormap64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libstorpds64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libstorsil64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libstorssl64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libsymapi64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libsymevtd64mt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/pa20_64/libsymlvm64mt.sl" missing.
ERROR:   Fileset "SYMCLI.64BIT,l=/opt/emc,r=V7.5.0.0" had file errors.
ERROR:   File "/opt/emc/usr/lib/libEmcpegclient.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libEmcpegcommon.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libEmcpegexportclient.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libEmcpegexportserver.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libEmcpeggeneral.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libEmcpeglistener.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libEmcpegslp_client.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libclarevtdmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/liboslevtdmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libsnmpevtdmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorapimt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorbasemt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorctrlmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorfilcimmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstormapmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorsilcimmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorsilmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libsymevtdmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libsymlvmmt.sl" missing.
ERROR:   Fileset "SYMCLI.BASE,l=/opt/emc,r=V7.5.0.0" had file errors.
ERROR:   File "/opt/emc/SYMCLI/PERL/unzip" should have mode "555" but
         the actual mode is "755".
ERROR:   Fileset "SYMCLI.SYMRECOVER,l=/opt/emc,r=V7.5.0.0" had file errors.
ERROR:   File "/opt/emc/usr/lib/libemc_crypto.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libemc_crypto.sl.0.9.8" missing.
ERROR:   File "/opt/emc/usr/lib/libemc_ssl.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libemc_ssl.sl.0.9.8" missing.
ERROR:   File "/opt/emc/usr/lib/libstorcoremt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorpdsmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libstorsslmt.sl" missing.
ERROR:   File "/opt/emc/usr/lib/libsymapimt.sl" missing.
ERROR:   Fileset "SYMCLI.THINCORE,l=/opt/emc,r=V7.5.0.0" had file errors.
```

```
       * Summary of Analysis Phase:
ERROR:       Verify failed SYMCLI.64BIT,l=/opt/emc,r=V7.5.0.0
ERROR:       Verify failed SYMCLI.BASE,l=/opt/emc,r=V7.5.0.0
             Verified      SYMCLI.SYMCLI,l=/opt/emc,r=V7.5.0.0
ERROR:       Verify failed SYMCLI.SYMRECOVER,l=/opt/emc,r=V7.5.0.0
ERROR:       Verify failed SYMCLI.THINCORE,l=/opt/emc,r=V7.5.0.0
ERROR:   4 of 5 filesets had Errors.
       * 1 of 5 filesets had no Errors or Warnings.
ERROR:   The Analysis Phase had errors.  See the above output for
         details.


=======  09/27/12 03:38:48 EDT  END verify AGENT SESSION (pid=29292)
         (jobid=hostname-4939)
```

HP OpenVMS-specific issues

For Solutions Enabler V7.1 and higher, 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 non OpenVMS host to ANY. For instructions, refer to the *Solutions Enabler 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 E
Solutions Enabler Directories

This appendix contains the directory list for UNIX and Windows installations:

UNIX directories

Table 34 lists the directories for UNIX platforms. Your directories may differ from this list since the location of these directories is configurable at installation.

Table 34 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.
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. 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 Symmetrix arrays.
Log files	/var/symapi/log	Contains SYMAPI logs and daemon logs.

Windows directories

Table 35 lists the default directories for Windows. Your directories may differ from this list since the location of these directories is configurable at installation.

Table 35 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\IBMUDB C:\Program Files\EMC\SYMCLI\shlib\sql\Oracle C:\Program Files\EMC\SYMCLI\shlib\sql\SQLSERVER C:\Program Files\EMC\SYMCLI\shlib\sql\ASM	IBM database engine. Oracle database engine. SQL server database engine. ASM database engine.
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.
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.
Security data	C:\Program Files\EMC\SYMAPI\authz_cache	Acts as a cache of authorization data from attached Symmetrix arrays.
SYMAPI log files	C:\Program Files\EMC\SYMAPI\log	Contains SYMAPI logs and daemon logs.
Providers	C:\Program Files\EMC\SYMCLI\shlib	VSS and VDS providers.
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 36 lists the default directories for OpenVMS. Your directories may differ from this list since the location of these directories is configurable at installation.

Table 36 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$HELP	STORCLI man pages. STORAPI man pages. SYMCLI man pages. SYMAPI and CLARAPI man pages.
Daemons	SYMCLI$DAEMONS	Location of the daemon executables.
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 USS directories

Table 37 lists the USS directories for z/OS. Your directories may differ from this list since the location of these directories is configurable at installation.

Table 37 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 creation. Used for client/server security.
Security data	/var/symapi/authz_cache	Acts as a cache of authorization data from attached Symmetrix arrays.
Log files	/var/symapi/log	Contains SYMAPI logs and daemon logs.

APPENDIX F
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 `se7500_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_V7.5.0.110525_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.

APPENDIX G
Legal Notices

This appendix contains legal attribution for acknowledging open-source and third-party software copyright, and licensing requirements for the EMC Solutions Enabler V7.5.

OpenSSL copyright information

Redistribution and use in source and binary forms, with or withoutmodification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment:

   "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed.

If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used.

This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

   "This product includes cryptographic software written by
   Eric Young (eay@cryptsoft.com)"
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement:

   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

Perl licensing information

Solutions Enabler uses Perl and Perl extensions software.

The standard version of code is located at:

```
http://www.perl.com/pub/a/language/info/software.html
```

For license information, refer to:

```
http://dev.perl.org/licenses/artistic.html
```

XML:: Parser licensing information

Solutions Enabler uses software from the XML Parser and an extension to Perl.

For further information, refer to:

```
http://search.cpan.org/src/MSERGEANT/XML-Parser-2.34/README
```


This program is free software; you can redistribute it and/or modify it under the same terms as Perl itself.

Expat Parser licensing information

Solutions Enabler uses software from the Expat XML Parser as part of the XML::Parser.

For further information, refer to

```
http://www.libexpat.org/
```


Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OFMERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANYCLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THESOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Info-ZIP licensing information

Solutions Enabler uses Info-ZIP.

For further information, refer to:

```
ftp://ftp.info-zip.org/pub/infozip/license.html
```


For the purposes of this copyright and license, "Info-ZIP" is defined as the following set of individuals:

Mark Adler, John Bush, Karl Davis, Harald Denker, Jean-Michel Dubois, Jean-loup Gailly, Hunter Goatley, Ian Gorman, Chris Herborth, Dirk Haase, Greg Hartwig, Robert Heath, Jonathan Hudson, Paul Kienitz, David Kirschbaum, Johnny Lee, Onno van der Linden, Igor Mandrichenko, Steve P. Miller, Sergio Monesi, Keith Owens, George Petrov, Greg Roelofs, Kai Uwe Rommel, Steve Salisbury, Dave Smith, Christian Spieler, Antoine Verheijen, Paul von Behren, Rich Wales, Mike White

This software is provided "as is," without warranty of any kind, express or implied. In no event shall Info-ZIP or its contributors be held liable for any direct, indirect, incidental, special or consequential damages arising out of the use of or inability to use this software.

ncFTP licensing information

Solutions Enabler uses software from ncFTP Software, Inc.

For further information about the product and specific instructions on downloading ncFTP for your own purposes, refer to:

```
http://www.nfctp.com
```


The Clarified Artistic License

Preamble:

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:

"Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder as specified below.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Distribution fee" is a fee you charge for providing a copy of this Package to another party.

"Freely Available" means that no fee is charged for the right to use the item, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.
2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain, or those made Freely Available, or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.
3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:
   a. place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major network archive site allowing unrestricted access to them, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.
   b. use the modified Package only within your corporation or organization.
   c. rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.
   d. make other distribution arrangements with the Copyright Holder.
   e. permit and encourage anyone who receives a copy of the modified Package permission to make your modifications Freely Available in some specific way.
4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:
   a. distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.
   b. accompany the distribution with the machine-readable source of the Package with your modifications.
   c. give non-standard executables non-standard names, and clearly document the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.
   d. make other distribution arrangements with the Copyright Holder.
   e. offer the machine-readable source of the Package, with your modifications, by mail order.
5. You may charge a distribution fee for any distribution of this Package. If you offer support for this Package, you may charge any fee you choose for that support. You may not charge a license fee for the right to use this Package itself. You may distribute

this Package in aggregate with other (possibly commercial and possibly nonfree) programs as part of a larger (possibly commercial and possibly nonfree) software distribution, and charge license fees for other parts of that software distribution, provided that you do not advertise this Package as a product of your own. If the Package includes an interpreter, You may embed this Package's interpreter within an executable of yours (by linking); this shall be construed as a mere form of aggregation, provided that the complete Standard Version of the interpreter is so embedded.

6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whoever generated them, and may be sold commercially, and may be aggregated with this Package. If such scripts or library files are aggregated with this Package via the so-called "undump" or "unexec" methods of producing a binary executable image, then distribution of such an image shall neither be construed as a distribution of this Package nor shall it fall under the restrictions of Paragraphs 3 and 4, provided that you do not represent such an executable image as a Standard Version of this Package.
7. C subroutines (or comparably compiled subroutines in other languages) supplied by you and linked into this Package in order to emulate subroutines and variables of the language defined by this Package shall not be considered part of this Package, but are the equivalent of input as in Paragraph 6, provided these subroutines do not change the language in any way that would cause it to fail the regression tests for the language.
8. Aggregation of the Standard Version of the Package with a commercial distribution is always permitted provided that the use of this Package is embedded; that is, when no overt attempt is made to make this Package's interfaces visible to the end user of the commercial distribution. Such use shall not be construed as a distribution of this Package.
9. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.
10. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE