



CHAPTER 4

Installation and Upgrading

This module describes how to install the Cisco Service Control Management Suite Subscriber Manager (SCMS SM); additionally, it describes how to upgrade and uninstall. This module also discusses topics related to installation, upgrading, and uninstalling.

How to Install, Upgrade, or Uninstall the Subscriber Manager

This module describes the procedures to install, upgrade, or uninstall the Subscriber Manager.

- [Information About Installing the SM, page 4-2](#)
- [Installing the Subscriber Manager, page 4-13](#)
- [Verifying the Installation, page 4-21](#)
- [Configuring the Subscriber Manager, page 4-21](#)
- [How to Perform Additional Installation Procedures, page 4-22](#)
- [How to Upgrade the Subscriber Manager, page 4-23](#)
- [How to Downgrade the Subscriber Manager, page 4-41](#)
- [How to Uninstall the Subscriber Manager, page 4-41](#)

Information About Installing the SM

Installing the SM is an automated process. It consists of executing an installation script residing on the root of the SM distribution files supplied by Cisco.


Note

For Solaris: The procedure also requires modifying the `/etc/system` file. Do this manually or use some other automated utility.


Note

For Linux: The procedure also requires modifying the `/etc/sysctl.conf` file. Do this manually or use some other automated utility.

- [Installation Overview, page 4-2](#)
- [Contents of the Distribution Files, page 4-3](#)
- [Documentation, page 4-6](#)
- [System Requirements, page 4-7](#)
- [Installation Procedures, page 4-9](#)
- [Information About System Changes Made by Installation Scripts, page 4-9](#)
- [Information About Advanced System Memory Configuration, page 4-11](#)

Installation Overview

The installation procedure installs the following components:

- SM and Command-Line Utilities (CLU)
- TimesTen database and DSN
- Java Runtime Environment (JRE)
- SM Veritas Cluster Agents

The installation procedure also includes:

- Setting up a `pcube` user and group
- Adding startup and shutdown scripts
- System configuration for TimesTen (performed manually or using a script)
- Replication scheme setting (performed by running a CLU). (Relevant only for cluster setups).

After completing installation and configuration, you can use the SM to introduce subscribers to the system.


Note

Login Event Generators and the SCA BB application are now installed by default on the SM.

Contents of the Distribution Files

The SCMS SM components are supplied in three distribution files:

- SM for Solaris
- SM for Linux
- Login Event Generators (LEGs)

Each distribution file is supplied as a tar file, which is compressed by gzip and has an extension of **.tar.gz**.

Table 4-1 lists the contents of the SM installation distribution files for Solaris and Linux.

Table 4-1 Contents of SM Distribution Files

Path	Name	Description
DIST_ROOT	—	Cross-platform files
	dhcp_forwarder.tar.gz	DHCP Forwarder distribution
	hooks.sh	User-defined function for upgrade
	install	Typical installation procedure description
	install-forwarder.sh	Installation script for the DHCP Forwarder used in conjunction with DHCP Lease Query LEG
	install-sm.sh	SM installation script
	install-vcs-agents.sh	VCS agents installation script
	linux-def.sh	Linux-specific definitions (only in the Linux distribution file)
	solaris-def.sh	Solaris-specific definitions (only in the Solaris distribution file)
	MANIFEST	CD information
	p3sm.sh	Startup and shutdown script
	Prerequisites	System minimal requirements list
	sm-common.sh	General installation script
	sm-dist.tar.gz	SM distribution
	tt-sysconf.sh	TimesTen system configuration script
	uninstall-sm.sh	SM uninstall script
	upgrade-sm.sh	SM upgrade script
	vcs-agents-dist.tar.gz	VCS agents distribution
	install-def.cfg	Contains definitions for several SM parameters that can be configured before SM installation or upgrade.
	verify-subscriber.sh	Script to verify whether a subscriber exists without running the SM. To be used when performing a cluster upgrade.
cluster-upgrade.sh	Upgrade script for cluster installations.	

Table 4-1 Contents of SM Distribution Files (continued)

Path	Name	Description
DIST_ROOT/Java/	—	Java Runtime Environment files
	jre1.5.0_14-linux.tar.gz	JRE for Linux (only in the Linux distribution file)
	jre1.5.0_14-solaris.tar.gz	JRE for Solaris (only in the Solaris distribution file)
	LICENSE	JRE license
DIST_ROOT/TimesTen/	—	TimesTen files
	pqb_resp_uninst.txt	Response file for TimesTen uninstall
	pqb-odbc.ini.txt	Open Database Connectivity (ODBC) definitions
	pqb-response70.txt	Response file for TimesTen installation
	pqb-sys-odbc.ini.txt	Open Database Connectivity (ODBC) definitions
	timesten70520.linux86.tar.Z	TimesTen for Linux (only in the Linux distribution file)
	timesten70500.sparc64.tar.Z	TimesTen for Solaris 64-bit (only in the Solaris distribution file)
DIST_ROOT/scripts	.bash_profile	User profile for bash
	.cshrc	User profile for C-Shell
	scmssm.sh	User profile utility file for bash
	scmssm.csh	User profile utility file for C-Shell
	dropRep.sql	SQL file used during the upgrade procedure
	install-dsn.sh	TimesTen DSN configuration script
	installjava.sh	JRE installation script
	install-tt.sh	TimesTen installation script
	get_name.sql	SQL file used during the upgrade procedure
	p3db	Utility file used by the installation/upgrade procedures
	deleteLastPqiInfo.sql	SQL file used during the upgrade procedure
	clu-common	Utility file used by the installation and upgrade procedures

Table 4-2 lists the contents of the LEG distribution file:

Table 4-2 Contents of the LEG Distribution File

Path	Name	Description
DIST_ROOT	—	Cross-platform files
	MANIFEST	Distribution information
DIST_ROOT/bgp_leg	—	Border Gateway Protocol (BGP) LEG files
	bgp_leg.tar.gz	BGP LEG distribution
	Install	LEG installation procedure description
	install-bgp-leg.sh	BGP LEG installation script
	linux-def.sh	Linux-specific definitions
	sm-common.sh	General installation script
	solaris-def.sh	Solaris-specific definitions
DIST_ROOT/cnr_leg	—	Cisco Network Register (CNR) LEG files
	cnr-leg-dist.tar.gz	CNR LEG distribution
	Install	LEG installation procedure definitions
DIST_ROOT/sce_api	—	SCE Subscriber API files
	Readme	API setup procedure description
	sce-java-api-dist.tar.gz	API distribution
DIST_ROOT/sm_api	—	SM API files
	Readme	API setup procedure description
	sm-c-api-dist.tar.gz	C API distribution
	sm-java-api-dist.tar.gz	Java API distribution

Documentation

The SM installation distribution file contains the following documents:

- Manifest—Contains the version and build numbers for all components from which the distribution files were built
- Install—The SCMS SM typical installation procedures
- Prerequisites—Minimal system requirements for installation of the SM

System Requirements

You can install the SM on the following platforms:

- Solaris—Sun SPARC machine (64-bit) running 64-bit versions of Solaris 9 or Solaris 10 with a 64-bit version of the Java Virtual Machine. See [Table 4-3](#), and [Table 4-4](#).
- Linux—Intel based machine with a 32-bit or 64-bit CPU running a 32-bit version of Linux with a 32-bit version of the Java Virtual Machine. See [Table 4-3](#), and [Table 4-5](#).



Caution

The SM must run on its own machine. You cannot run it on the same machine as the Collection Manager.

The machine should conform to the system requirements listed in [Table 4-3](#), [Table 4-4](#), and [Table 4-5](#).



Note

The specifications listed in [Table 4-3](#) are minimal. They should be verified in order to guarantee specific performance and capacity requirements.

Table 4-3 Minimal System Hardware Requirements

Item	Requirement
CPU	<ul style="list-style-type: none"> • SUN SPARC, 64-bit, minimum 500 MHz (for Solaris) • Intel processor, 32- or 64-bit, minimum 1 GHz (for Linux Red Hat)
RAM	Minimum 1 GB; see Table 4-6 or Table 4-8 .
Free Disk Space	Minimum 3 GB total, of which: <ul style="list-style-type: none"> • Minimum 1 GB free on partition where VARDIR (SM database repository) is installed • Minimum 0.5 GB free on partition where PCUBEDIR (SM files) is installed • Minimum 200 MB free on partition where <code>/tmp</code> is mounted
Network Interface	Depends on whether or not the configuration includes a cluster: <ul style="list-style-type: none"> • Without cluster—One (1) 100BASE-T Ethernet • With cluster—Six (6) 100BASE-T Ethernet
CD-ROM drive	Recommended



Note

For the hardware and software system requirements for the Veritas Cluster Server, see [Veritas Cluster Server, page E-1](#).

Table 4-4 Solaris System Software Requirements

Item	Requirement
OS	Solaris 5.9 64-bit or later; currently, only 64-bit versions of Solaris 5.9 and 5.10 are supported. Solaris Core Installation
System Packages	Mandatory: <ul style="list-style-type: none"> • SUNWbash—GNU Bourne-Again shell (bash) • SUNWgzip—GNU Zip (gzip) compression utility • SUNWzip—Info-Zip (zip) compression utility • SUNWlibC—Sun Workshop Compilers Bundled libC • SUNWlibCx—Sun WorkShop Bundled 64-bit libC • sudo (superuser do) package Optional: <ul style="list-style-type: none"> • SUNWadmap—system administration applications • SUNWadmc—system administration core libraries

**Note**

It is strongly recommended to apply the latest patches from SUN. You can download the latest patches from the SUN patches website.

Table 4-5 Red Hat System Software Requirements

Item	Requirement
OS	Red Hat Enterprise Linux AS/ES 4.0/5.0; currently, only 32-bit versions are supported. Red Hat Core Installation
System Packages	Mandatory: <ul style="list-style-type: none"> • GNU Bourne-Again shell (bash-2.05b-29.i386.rpm) • GNU Data Compression Program (gzip-1.3.3-9.i386.rpm) • File compression and packaging utility (zip-2.3-16.i386.rpm) • Standard C++ libraries for Red Hat Linux 6.2 backward compatibility (compat-gcc-7.3-2.96.122.i386.rpm) • sudo (superuser do) package For integrating with the C API: <ul style="list-style-type: none"> • GNU cc and gcc C compilers (gcc-3.2.3-20.i386.rpm) • C++ support for the GNU gcc compiler (gcc-3.2.3-20.i386.rpm)

**Note**

It is strongly recommended to apply the latest patches from Red Hat.

**Note**

Only 32-bit versions of Linux are supported, but it is possible to install 32-bit Linux on a 64-bit CPU.

Installation Procedures

All installations can be performed by executing an installation script located on the root of the SM distribution file.

In most cases, the SM installation script is the only script needed for completing the installation.

The installation script displays messages describing the significant steps that are being performed. These messages are also sent to the system log for future reference. See [Logging Script Messages](#) for more information about the system log messages.

If you try to install the SM on a machine on which the SM is currently running, or to a directory in which the SM is already installed (even if not running), the operation will fail and you will be requested to upgrade the SM. See [How to Upgrade the Subscriber Manager, page 4-23](#).

The specific installation procedure to be applied depends on the required SM topology.

For the installation procedure for the standalone topology, see [Installing the Subscriber Manager, page 4-13](#).

For the installation procedure for the cluster topology, see [Installing an SM Cluster, page 4-22](#).

Information About System Changes Made by Installation Scripts

This section describes the system changes applied automatically by the SM installation. The SM installation adds a dedicated user and group, and startup and shutdown scripts.

- [Logging Script Messages, page 4-9](#)
- [pcube User and Group, page 4-9](#)
- [Startup and Shutdown Scripts, page 4-10](#)
- [Bash and C-Shell Profiles for the User pcube, page 4-10](#)

Logging Script Messages

Script messages are logged into the system log in the following manner:

- For Solaris—The installation scripts log all their messages into the system log, which is usually the file located at `/var/adm/messages`. The messages are logged to the `user.info` syslog category.
- For Linux—The installation scripts log all their messages into the system log, which is usually the file located at `/var/log/messages`. The messages are logged to the `user.info` syslog category.

pcube User and Group

During installation, a user named `pcube` is created (unless it already exists) with its own group. This user owns all installed SM and CLU files. The user home directory is the installation directory selected during installation. For security purposes, the user is initially created with a locked password. You must assign a new password.

Startup and Shutdown Scripts

The SM is started on boot to run level 2, and is stopped when leaving this run level (for example, when the machine is shut down).

The installer installs the following files for startup and shutdown:

- For Solaris:

```
-rwxr--r-- 1 root other /etc/init.d/p3sm
lrwxrwxrwx 1 root other /etc/rc0.d/K44p3sm ->/etc/init.d/p3sm
lrwxrwxrwx 1 root other /etc/rc1.d/K44p3sm ->/etc/init.d/p3sm
lrwxrwxrwx 1 root other /etc/rc2.d/S92p3sm ->/etc/init.d/p3sm
lrwxrwxrwx 1 root other /etc/rcS.d/K44p3sm ->/etc/init.d/p3sm
```

- For Linux:

```
-rwxr--r-- 1 root other /etc/init.d/p3sm
lrwxrwxrwx 1 root other /etc/rc0.d/K44p3sm ->/etc/init.d/p3sm
lrwxrwxrwx 1 root other /etc/rc1.d/K44p3sm ->/etc/init.d/p3sm
lrwxrwxrwx 1 root other /etc/rc2.d/S92p3sm ->/etc/init.d/p3sm
lrwxrwxrwx 1 root other /etc/rc3.d/S92p3sm ->/etc/init.d/p3sm
lrwxrwxrwx 1 root other /etc/rc5.d/S92p3sm ->/etc/init.d/p3sm
lrwxrwxrwx 1 root other /etc/rc6.d/K44p3sm ->/etc/init.d/p3sm
```

The TimesTen installer creates similar startup and shutdown scripts.

Bash and C-Shell Profiles for the User pcube

The SM is controlled using the CLUs that are located in `~pcube/sm/server/bin`. If such shell-profiles do not exist, the installation and upgrade scripts create profiles setting the CLU directory in the user `pcube` path environment variable.

This operation can be performed manually as well by copying the content of these profile scripts (`scmssm *` files) from the SM distribution under `DIST_ROOT/scripts/`.

Information About Advanced System Memory Configuration

- [Configuring /var/TimesTen/sys.odbc.ini, page 4-11](#)
- [Configuring the SM Process Memory Settings, page 4-11](#)

Configuring /var/TimesTen/sys.odbc.ini

Some installations might require changing TimesTen parameters so that the database will run as desired. However, do not make any changes if the default values suit your requirements.

Setting the multi-processor optimization

If your system is a multi-processor machine, the value of the **SMPOptLevel** parameter in the **[Pcube_SM_Repository]** section in the **sys.odbc.ini** file should be set to 1. Otherwise, it should be set to 0 or not set at all. The installation script automatically sets this parameter according to the number of available processors.

Setting the database size

If your system needs to support more than 100,000 subscribers, set the values of the **PermSize** and **TempSize** parameters in the **[Pcube_SM_Repository]** section in the **sys.odbc.ini** file.

See [Step 2 Determine the system memory settings, page 4-14](#).

For example:

```
PermSize=500
TempSize=150
```



Note

If you change the database size, you must also make the following changes:

- Solaris—Set the value of parameter **shmsys:shminfo_shmmax** in the **/etc/system** file to be larger than the sum of **PermSize** and **TempSize**.
- Red Hat—Set the value of parameter **kernel.shmmax** in the **/etc/sysctl.conf** file to be larger than the sum of **PermSize** and **TempSize**.

Configuring the SM Process Memory Settings

By default, the SM process uses 256 MB of RAM memory. However, in certain application component configurations, the SM process needs to allocate additional memory to work correctly. Setting an environment variable called **PCUBE_SM_MEM_SIZE** with the desired memory size (in megabytes) instructs the SM start-up scripts to allocate the defined memory size for the SM process.

You can set the memory size value for this environment variable for the user **pcube**, or you can configure the desired process memory size in the **sm.sh** file located in the root directory of the user **pcube** (**~pcube/sm.sh**).

The following example, which shows a line in the **sm.sh** file, defines a memory size of 512 MB for the SM process:

```
PCUBE_SM_MEM_SIZE=512
```



Note

If the **PCUBE_SM_MEM_SIZE** environment variable does not exist, you must add this line to the **sm.sh** file.

**Note**

You can configure **PCUBE_SM_MEM_SIZE** in the **install-def.sh** script file before running the installation script. This ensures that the SM will be installed with the correct value configured.

**Note**

To prevent performance degradation because of memory swapping, make sure that the machine has enough RAM for the SM process, the SM database, and all of the other applications running on this machine.

**Note**

To determine the correct memory values for your installation, see [Step 2 Determine the system memory settings, page 4-14](#).

Installing the Subscriber Manager

This section describes how to install the Subscriber Manager.

**Note**

In a high availability setup (see [SM Cluster, page 2-13](#)), you must install the SM Cluster VCS agents. See [Installing SM Cluster Agents, page 4-23](#).

Prerequisites

Make sure that the disk space requirements listed in [System Requirements](#) are satisfied.

Installation Procedure

Step 1 Extract the distribution files.

Before you can install the SM, you must first load and extract the distribution files on the installed machine or in a directory that is mounted to the installed machine.

- a. Download the distribution files from the Cisco web site.
- b. Use FTP to load the distribution files to the SM.
- c. Unzip the files using the **gunzip** command.

```
gunzip SM_dist_<version>_B<build number>.tar.gz
```

- d. Extract the tar file using the **tar** command:

```
tar -xvf SM_dist_<version>_B<build number>.tar
```

Step 2 Determine the system memory settings

Set the system memory configuration requirements according to the maximum number of subscribers. There are two methods to determine the system memory settings: without the Quota Manager or with the Quota Manager.

Table 4-6 and Table 4-7 list the recommended memory configuration values based on the number of supported subscribers. The settings apply when the Quota Manager is disabled.

Table 4-6 *Memory Configuration Parameters Versus Number of Subscribers: without Quota Manager*

Maximum Number of Subscribers	(TimesTen Memory Settings) Shared Memory	(TimesTen Memory Settings) PermSize	(TimesTen Memory Settings) TempSize
100,000	512 MB	200 MB	100 MB
500,000	1024 MB	512 MB	256 MB
1,000,000	1280 MB	768 MB	256 MB
2,000,000	2048 MB	1536 MB	256 MB
3,000,000	2560 MB	2048 MB	256 MB
4,000,000	3328 MB	2816 MB	256 MB
5,000,000	3840 MB	3328 MB	256 MB
10,000,000	6912 MB	6400 MB	256 MB
15,000,000	9984 MB	9472 MB	256 MB
20,000,000	13312 MB	12800 MB	256 MB

Table 4-7 SM Process Memory Setting Versus Cache Size: without Quota Manager

Cache Size	SM Process Memory Setting
100,000	256 MB
500,000	512 MB
1,000,000	768 MB
2,000,000	1280 MB
3,000,000	1792 MB
4,000,000	2048 MB
5,000,000	2560 MB
10,000,000	5120 MB
15,000,000	7680 MB
20,000,000	10240 MB

**Caution**

The SM process RAM in the table is calculated for:

- 40 SCE connections per SM—For each additional SCE you should add an additional 25 MB for the SM process memory setting.
- 20 PRPC (SM API/CNR LEG) connections to the SM—For each additional connection you should add an additional 25 MB for the SM process memory setting.
- If you use the virtual-links ability of the service control solution you should add an additional 60 MB to the Perm Size setting for each additional 100,000 subscribers.

**Note**

For Linux installations, there are further limitations:

- The maximum number of subscribers is two million.
- The maximum number of SCEs is 200.
- The maximum value for the SM process memory settings is 1.8 GB.
- The combined value of the TimesTen shared memory setting and the SM process memory setting should not exceed 3 GB.

Description of the table columns:

- Maximum Number of Subscribers—The maximum number of subscribers that the SM has to support.
- Cache Size—The number of subscriber record references the SM process maintains. The default value is 100,000 records.



Note The default cache size can be enlarged to increase performance; however, this does require additional memory. The default settings are suitable for most installations.

- SM Process Memory Setting—The required memory configuration for the SM process itself. For additional information about the SM process memory configuration, see [Table 4-7](#).

- The configuration required for TimesTen to run correctly. For additional information, see [Step 3 Configure the shared memory settings, page 4-17](#).

If the previous tables do not list the maximum number of subscribers that you require, use the settings specified for the next higher value of Maximum Number of Subscribers. For example, for 1,200,000 subscribers, use the values specified for 2,000,000 subscribers.

[Table 4-8](#) and [Table 4-9](#) list the recommended memory configuration values based on the number of supported subscribers. The settings apply when the Quota Manager is enabled.

Table 4-8 *Memory Configuration Parameters Versus Number of Subscribers: with Quota Manager*

Maximum Number of Subscribers	(TimesTen Memory Settings) Shared Memory	(TimesTen Memory Settings) PermSize	(TimesTen Memory Settings) TempSize
500,000	1280 MB	768 MB	256 MB
1,000,000	1792 MB	1280 MB	256 MB
2,000,000	3072 MB	2560 MB	256 MB
3,000,000	4096 MB	3584 MB	256 MB
4,000,000	5376 MB	4864 MB	256 MB
5,000,000	6400 MB	5888 MB	256 MB
10,000,000	12032 MB	11520 MB	256 MB
15,000,000	17664 MB	17152 MB	256 MB
20,000,000	23552 MB	23040 MB	256 MB

Table 4-9 *SM Process Memory Setting Versus Cache Size: with Quota Manager*

Cache Size	SM Process Memory Setting
100,000	512 MB
500,000	512 MB
1,000,000	768 MB
2,000,000	1280 MB
3,000,000	1792 MB
4,000,000	2048 MB
5,000,000	2560 MB
10,000,000	5120 MB
15,000,000	7680 MB
20,000,000	10240 MB



Caution

The SM process RAM in the table is calculated for:

- 20 SCE connections per SM—For each additional SCE you should add an additional 50 MB for the SM process memory setting.

- 20 PRPC (SM API/CNR LEG) connections to the SM—For each additional connection you should add an additional 25 MB for the SM process memory setting.
- If you use the virtual-links ability of the service control solution you should add an additional 60 MB to the Perm Size setting for each additional 100,000 subscribers.

**Note**

For Linux installations, there are further limitations:

- The maximum number of subscribers is two million.
- The maximum number of SCEs is 200.
- The maximum value for the SM process memory settings is 1.8 GB.
- The combined value of the TimesTen shared memory setting and the SM process memory setting should not exceed 3 GB.

Step 3 Configure the shared memory settings

TimesTen requires that certain changes be made in the operating system kernel configuration file:

- For Solaris, modify file **/etc/system**.
- For Linux, modify file **/etc/sysctl.conf**.

These changes increase the shared memory and semaphore resources on the SM machine from their defaults. For additional information regarding these changes, refer to the TimesTen documentation.

**Note**

It is recommended that you review the **/etc/system** or the **/etc/sysctl.conf** file before running the **tt-sysconf.sh** script, because the script overwrites the current file settings with the values listed in the “Making the changes manually” procedure. If you want to keep some or all of the current file settings, edit the configuration file by performing the changes manually.

- a. Make the changes automatically or manually.
 - To make the required changes automatically, run the **tt-sysconf.sh** script. The root user must invoke this script file, without arguments, as follows:


```
# tt-sysconf.sh
```
 - To make the required changes manually:

**Note**

Editing the configuration file manually is required when you require support for more than 100,000 subscribers in the SM. Your system's sizing requirements only affect the shared memory size. To determine the correct configuration values for your system, see the tables in [Step 2 Determine the system memory settings, page 4-14](#).

- For Solaris, make the required changes manually by adding the following lines to the **/etc/system** file and configuring the shared memory size:

```
*---- Begin settings for TimesTen
set semsys:seminfo_semmni = 20
set semsys:seminfo_semmsl = 100
set semsys:seminfo_semmns = 2000
set semsys:seminfo_semmnu = 2000
set shmsys:shminfo_shmmax = 0x20000000
*---- End of settings for TimesTen
```

- For Linux, make the required changes manually by adding the following lines to the `/etc/sysctl.conf` file and configuring the shared memory size:

```
*---- Begin settings for TimesTen
kernel.shmmax = 536870912
kernel.sem = 250 32000 100 100
*---- End of settings for TimesTen
```

Step 4 Edit the `install-def.cfg` file.



Note

This step is optional when performing the SM installation. However, it is recommended to edit the file if one of the parameter values should not be set to the default value.

The `install-def.cfg` file contains several parameters that can be preconfigured before installation/upgrade of the SM. These parameters are copied by the install/upgrade routine to the relevant SM configuration files. By default, all of the parameters are commented out and the default values are used.

The file contains the following parameters:

- `max_subscribers_num`

Resides in the **[SM Definitions]** section. Defines the maximum number of subscribers the SM supports. You can set the maximum number of subscribers using this parameter or by setting the **max_number_of_subscribers** parameter in `p3sm.cfg` configuration file. See [Data Repository Section, page A-21](#).

There is a limit to the maximum number of subscribers that can be stored in the SM database. The limit is 20 million subscribers for Solaris and two million subscribers for Linux.

The SM default configuration supports a maximum of 200,000 subscribers.

- `sm_memory_size`

Resides in the **[SM Definitions]** section. Defines the amount of memory allocated for the SM process in MB. You can set the parameter here or edit `PCUBE_SM_MEM_SIZE` in the `sm.sh` file that resides under the `~pcube` folder.

- `database_perm_size`

Resides in the **[Database Definitions]** section. Defines the `PermSize` allocated for the database in MB. You can set the parameter here or edit the `PermSize` parameter in the `/var/TimesTen/sys.odbc.ini` file.

- `database_temp_size`

Resides in the **[Database Definitions]** section. Defines the `TempSize` allocated for the database in MB. You can set the parameter here or edit the `TempSize` parameter in the `/var/TimesTen/sys.odbc.ini` file.

Step 5 Execute the `install-sm.sh` script



Note

The `install-sm.sh` script is customizable.



Note

It is not possible to run the script if the `/etc/motd` file exists. The file should be moved *or* removed prior to running the `install-sm.sh` script.

- a. From your machine shell prompt, move to the directory to where the distribution file was extracted and run the **install-sm.sh** script:

```
# install-sm.sh [ command options ]
```

Table 4-10 lists the command options:

Table 4-10 Options for *install-sm.sh*

Options	Description
-d	Specifies the install directory for ~pcube . This directory must <i>not</i> be an existing directory. This directory must be specified as a full pathname beginning with "/". The default is /opt/pcube .
-o	Specifies the existing home directory of user pcube as the install directory. Note The options -d and -o are mutually exclusive.
-v	Specifies the directory for data storage. This directory must <i>not</i> be an existing directory. This directory must be on a partition with at least 1 GB of free space. This directory must be specified as a full pathname beginning with "/". The default is InstallDirectory/var .
-h	Shows this message.

The script performs the following steps:

- Checks for validity of arguments and sufficient disk space.
- Adds (or verifies the existence of) a user **pcube** and a group **pcube**.
- Populates the **pcube** home directory with the SM and CLU directory structure.
- Invokes the JRE installation script with **pcube** home as the target directory. The JRE installation does not affect any existing Java installations.
- Invokes the TimesTen installation script with **pcube** home as the target directory.
- Creates the SM DSN for TimesTen with **pcube** home as the target directory. It is possible to install the SM DSN for TimesTen in a specified directory by using the -v option.
- Creates startup and shutdown scripts in **/etc**.
- Creates the shell preamble **~pcube/sm.sh**, which contains environment variables that depend on the actual folder in which the SM was installed.

Examples for the **install-sm.sh** Script

These examples demonstrate how to use the **install-sm.sh** script to install the SM.

- [Installing the SM and CLU: Example, page 4-20](#)
- [Installing the SM and CLU to a Default Directory: Example, page 4-20](#)

Installing the SM and CLU: Example

This example installs the SM and CLU to a directory named `/usr/local/pcube` using the default data storage directory.

```
# install-sm.sh -d /usr/local/pcube
```

Installing the SM and CLU to a Default Directory: Example

This example installs the SM and CLU to the default directory of the user `pcube`.

```
# install-sm.sh -o
```

Step 6 Set the password for the `pcube` user

After the installation script has completed successfully, set the password for the `pcube` user by running the `# passwd pcube` command.

**Note**

It is important to remember the password you have selected.

Step 7 Reboot the computer.

It is necessary to reboot the computer to complete the installation.

Step 8 (Optional) Install the MPLS/VPN BGP LEG.

For further information, see the [Cisco SCMS SM LEGs User Guide](#), the “[Installing the MPLS/VPN BGP LEG](#)” chapter.

Step 9 Add a user for PRPC authentication.

It is necessary to add a user for PRPC authentication because SCA BB requires a username and password when connecting to the SM.

To add a user for PRPC authentication, use the `p3rpc` command-line utility. For example:

```
>p3rpc --set-user --username=username --password=password
```

For cluster installations, use the `--remote` option, after both devices are installed, as shown in the following example:

```
>p3rpc --set-user --username=username --password=password --remote=OTHER_SM_IP[:port]
```

For troubleshooting the installation, see [Troubleshooting](#), page D-1.

Verifying the Installation

To verify that the installation was successful, run a CLU utility, such as the **p3sm** command, to display general information about the SM.

Step 1 From your machine shell prompt, change to the **~pcube/sm/server/bin** directory.

Step 2 Run the **p3sm** command.

The following **p3sm** command displays the current status of the SM.

```
>p3sm --sm-status
```



Note

Wait a few minutes after the installation before running this command to allow the SM to become operational.

The output of this command should indicate that the SM is running.

In case of errors during installation, the command will output a description of these errors.

Configuring the Subscriber Manager

After installing the SM, you can configure the SM to your specific needs. In particular, you should address the following parameters at this point:

- **topology**—Cluster or standalone
- **introduction_mode**—Pull or push
- **support_ip_ranges**—Whether IP-ranges should be used in the installed setup

To configure the SM, edit the **p3sm.cfg** configuration file using any standard text editor. The configuration file is described in detail in the Configuration and Management module and in the Configuration File Options module. After you finish editing the **p3sm.cfg** configuration file, use the **p3sm** utility to update the SM with the new settings:

Step 1 From your machine shell prompt, run the **p3sm** command.

The following **p3sm** command loads the configuration file and updates the SM configuration accordingly.

```
>p3sm --load-config
```

How to Perform Additional Installation Procedures

The following procedures complement the ones described in [Installing the Subscriber Manager, page 4-13](#):

- [Installing an SM Cluster, page 4-22](#)—Should be used if installing two SM nodes for the first time.
- [Installing SM Cluster Agents, page 4-23](#)—Must be installed when in a High Availability setup where Veritas Cluster Server (VCS) is used.

Installing an SM Cluster

The installation of an SM cluster is very similar to installing the SM on two machines.

-
- Step 1** Install the Veritas Cluster Server software on both machines.
- Step 2** Install the SM on both machines.
For further information, see [Installing the Subscriber Manager, page 4-13](#).
- Step 3** Configure the SM topology parameter to cluster.
For further information, see [Configuring the Subscriber Manager, page 4-21](#).
- Step 4** Configure the replication scheme.
It is necessary to configure the replication scheme for the data-store replication to the redundant machine by running the following CLU:
- ```
p3db --set-rep-scheme
```
- Step 5** Install the SM VCS agents.  
For further information, see [Installing the Subscriber Manager, page 4-13](#).
- Step 6** Configure the VCS.  
For further information, see the [Veritas Cluster Server](#) module.
-

## Installing SM Cluster Agents

The installation distribution file contains a set of customized Veritas Cluster Agents for supporting monitoring and controlling of SM-related resources in cluster topology. You must install the cluster agents under the VCS **bin** directory.



### Note

It is not possible to run the script if the **/etc/motd** file exists. The file should be moved *or* removed prior to running the **install-vcs-agents.sh** script.

**Step 1** From your machine shell prompt, run the **install-vcs-agents.sh** script.

```
install-vcs-agents.sh [command-options]
```

Table 4-11 lists the command options.

**Table 4-11** Options for *install-vcs-agents.sh*

| Options | Description                                                                                                                                                                                                                                                                    |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -d      | Specifies the installation directory for the agents, which must be the bin directory of the VCS.<br><br>This directory must be an existing directory.<br><br>This directory must be specified as a full pathname that begins with '/'.<br><br>Default: <b>/opt/VRTSvcs/bin</b> |
| -h      | Prints a help message and exits.                                                                                                                                                                                                                                               |

The script performs the following steps:

- Checks that the installation directory exists.
- Extracts the agent distribution file to the specified directory.
- Copies the VCS default-script-agent-executable from the installation directory to all agent directories.

## How to Upgrade the Subscriber Manager

The Subscriber Manager supports several types of upgrade procedures, according to the SM version that was previously installed and the requirement (or lack of requirement) for fail-over in the new installation.

There are three types of upgrade procedure:

- [Upgrading from a Standalone Setup, page 4-25](#)
- [Upgrading from a Standalone Setup to a Cluster Setup, page 4-29](#)
- [Upgrading Cluster Setups, page 4-31](#)

## Data Duplication Procedure

The data duplication procedure enables the user to duplicate or copy the entire database from one machine to the other, and then keep the databases synchronized by running the replication agent at the end. Some of the upgrade procedures described in the previous sections use this procedure.

For details of the procedure, see [Database Duplication Recovery, page 3-7](#).

## Automatic Upgrade of Subscribers with VLAN Mappings

The 3.1.5 release of the SM introduces a new managed entity—VPN. This new entity means that VLAN mappings are no longer related to a subscriber but to a VPN. During the upgrade procedure, the SM automatically creates a VPN with the VLAN-ID of the subscriber and associates a subscriber with the full range IP mapping in the new VPN.

For example, subscriber 'sub1' with VLAN-ID=15 will result in the creation of VPN 15 with VLAN-ID=15 and subscriber 'sub1' with the mapping 0.0.0.0/0@VLAN-ID.

## Automatic Upgrade of RADIUS Listener

The 3.1.5 release of the SM introduces new manipulation abilities to the RADIUS Listener.

During the upgrade procedure, the SM modifies the RADIUS sections in the configuration file according to the following rules:

- The **radius\_attribute** and **radius\_attribute\_type** properties are moved to a new section.
- A new field property is added to replace the **radius\_attribute** and **radius\_attribute\_type** properties.
- The **strip\_type=remove\_suffix** property is replaced with **field\_manipulation.<field name>=(.\*)<strip\_character >.\***.
- The **strip\_type=remove\_prefix** property is replaced with **field\_manipulation.<field name>=.\*<strip\_character >(.\*)**.
- The **use\_default** property and default value are replaced with **mapping\_table.^\$=<default>**.
- The **radius\_attribute\_vendor\_id** and **radius\_sub\_attribute** properties are replaced with the format **radius\_attribute(radius\_attribute\_vendor\_id; radius\_sub\_attribute)**.



## Upgrading from a Standalone Setup



**Note** To upgrade from Version 2.x to Version 3.5.0, you must first upgrade to Version 3.0 or 3.1.

The following upgrade procedure is supported only from Release 3.x software versions, to upgrade the SM from a Release 2.x software version, the user must first upgrade their SM environment to a Release 3.1.x software version and then perform an additional upgrade to Release 3.5.0.

For more information about how to upgrade from a Release 2.x software version to a Release 3.1.x software version, please refer to the Release 3.1.x manuals.

This procedure applies to the SM 3.0.x and up. This upgrade procedure requires service down-time.



**Note** For the upgrade procedure from a standalone setup to a cluster setup, see [Upgrading from a Standalone Setup to a Cluster Setup, page 4-29](#).

### Configuring the Required Memory Settings

To prepare the SM for the upgrade, configure the system kernel configuration file on the SM according to the procedure described in [Step 3Configure the shared memory settings, page 4-17](#).

**Step 1** Extract the distribution files.

Before you can upgrade the SM, you must first load and extract the distribution files on the installed machine or in a directory that is mounted to the installed machine.

- a. Download the distribution files from the Cisco web site.
- b. Use FTP to load the distribution files to the SM.
- c. Unzip the files using the **gunzip** command.

```
gunzip SM_dist_<version>_B<build number>.tar.gz
```

- d. Extract the tar the file using the **tar** command.

```
tar -xvf SM_dist_<version>_B<build number>.tar
```

**Step 2** Disable state exchange.

If upgrading from version 2.x, disable the state exchange between the SM and the SCE platform by editing the SM configuration file (**p3sm.cfg**) and set **save\_subscriber\_state=false**, then load the configuration file using the following command:

```
>p3sm --load-config
```



**Note** You must use this CLU as user **pcube**.

**Step 3** Edit the **install-def.cfg** file.

Edit the **install-def.cfg** configuration file and set the **PermSize** and **TempSize** parameters according to the recommendations described in [Step 3Configure the shared memory settings, page 4-17](#). For further information, see [Step 4Edit the install-def.cfg file., page 4-18](#).

**Step 4** Run the **upgrade-sm.sh** script.

In order to upgrade from non-cluster setups, the Subscriber Manager distribution provides an upgrade script that implements an upgrade from previous versions. The upgrade procedure script preserves the subscriber database and the entire SM configuration, including network elements, domains, and application-specific components.

**Note**

For Solaris: Previous versions of the SM on Solaris used a 32-bit or 64-bit Java Virtual Machine (JVM) and database. The SM is currently installed with a 64-bit JVM and database. There is no choice as to whether to upgrade to 64-bit.

**Note**

For Linux: Upgrades on Linux systems are only from SM 2.5.x and 3.x releases. The Linux platform is used only with a 32-bit JVM and database.

**Note**

It is not possible to run the script if the `/etc/motd` file exists. The file should be moved *or* removed prior to running the `upgrade-sm.sh` script.

- a. From your machine shell prompt, run the `upgrade-sm.sh` script.

```
upgrade-sm.sh [command-options]
```

**Table 4-12** Options for `upgrade-sm.sh`

| Options | Description         |
|---------|---------------------|
| -h      | Shows this message. |

The script performs the following steps:

- Detects existing SM version.
- Detects new SM version.
- Verifies that Java is installed on the machine.
- Verifies that the user **pcube** exists.
- Verifies that an SM of version 3.0.x or later is present on the system.
- Stops the current SM (if running).
- Backs up existing contents of the subscriber database to an external file.
- Removes the TimesTen database.
- Backs up SM configuration files.
- Installs the updated versions of SM and TimesTen.
- Invokes a separate program for upgrading the SM and database configuration files.
- Restores the backed up contents of the subscriber database.
- Starts the upgraded SM.

**Upgrading the SM: Example**

This example upgrades the SM and keeps the current database.

```
upgrade-sm.sh
```




---

**Note** An SM reboot is not required after the upgrade procedure.

---

**Step 5** Add a user for PRPC authentication.

If upgrading from a version of the SM prior to 3.0.5, it is necessary to add a user for PRPC authentication because SCA BB requires a username and password when connecting to the SM.

To add a user for PRPC authentication, use the **p3rpc** CLU. For example:

```
>p3rpc --set-user --username=username --password=password
```

**Step 6** Remove obsolete state information.

If upgrading from version 2.x, remove any obsolete subscriber state information, by running the SM CLU as **pcube** user:

```
>p3subsdb --clear-all-states
```

**Step 7** Remove obsolete subscriber properties (Method A)

If upgrading from version 2.x, remove any obsolete subscriber properties.




---

**Note** All CLU commands must be run as user **pcube**.

---

a. Export any existing subscribers to a csv file.

```
>p3subsdb --export -o csv-file
```

b. Clear the subscriber database.

```
>p3subsdb --clear-all
```

c. Remove any obsolete properties from the csv-file.

See [Table 4-13](#) for a list of properties to be removed.

d. Import the subscribers from the revised file.

```
>p3subsdb --import -f csv-file
```

**Step 8** Remove obsolete subscriber properties (Method B)

If upgrading from version 2.x, remove any obsolete subscriber properties.




---

**Note** All CLU commands must be run as user **pcube**.

---

a. Remove the obsolete properties from the SM database by running the **p3subsdb** command.

```
>p3subsdb --remove-property --property=prop
```

The obsolete properties to be removed are listed in [Table 4-13](#).

b. Resynchronize all SCEs.

```
>p3sm --resync-all
```

**Table 4-13** *Obsolete Properties to be Removed*

| <b>QP Tunable</b>      | <b>Applicable to Version</b>                                                     |
|------------------------|----------------------------------------------------------------------------------|
| QpLimit[1]–QpLimit[16] | Version 2.5.5 or higher                                                          |
| QpSet[1]–QpSet[16]     | Version 2.5.5 or higher                                                          |
| QpAdd                  | Versions of 2.5 prior to version 2.5.5<br>Versions of 2.1 prior to version 2.1.7 |
| QpDelta[1]–QpDelta[16] | Versions of 2.5 prior to version 2.5.5<br>Versions of 2.1 prior to version 2.1.7 |

**Step 9** Configure the SCE platforms.

If using a cascade SCE setup, configure the cascade SCE pair in the **p3sm.cfg** file as described in [SCE.XXX Section, page A-18](#).

## Upgrading from a Standalone Setup to a Cluster Setup


**Note**

To upgrade from Version 2.x to Version 3.5.0, you must first upgrade to Version 3.0 or 3.1.

The following upgrade procedure is supported only from Release 3.x software versions, to upgrade SM from a Release 2.x software version, the user must first upgrade their SM environment to a Release 3.1.x software version and then perform an additional upgrade to Release 3.5.0.

For more information about how to upgrade from a Release 2.x software version to a Release 3.1.x software version, please refer to the Release 3.1.x manuals.

This section describes the basic procedure for upgrading from a standalone setup to a cluster setup. This procedure applies for the SM from version 3.0.x and up. This upgrade procedure requires service down-time.


**Note**

This procedure attempts to minimize the SM downtime as much as possible. Therefore, if subscriber service is not an issue, use instead the procedure for installing a new machine and upgrading a new machine.

In the following procedure, SM-A is the original SM machine running SM version 3.0.x and later, and SM-B is the new SM machine being added for redundancy.

**Step 1** Install the VCS on both machines.

**Step 2** Install SM-B.

To install SM-B, follow the procedure described in [Installing the Subscriber Manager, page 4-13](#).

**Step 3** Upgrade SM-A.

To upgrade SM-A, follow the procedure described in [Upgrading from a Standalone Setup, page 4-25](#).


**Note**

From this step until the upgrade procedure is completed, there is no SM to handle subscribers.

**Step 4** Replicate the SM configuration from SM-A to SM-B (copy all the configuration files from ~pcube/sm/server/root/config folder).

Copy the **p3sm.cfg** configuration file manually from SM-A to SM-B. To load the configuration file, see [Reloading the SM Configuration \(p3sm\), page 5-18](#).

**Step 5** Duplicate the subscriber database.

The data duplication procedure is described in [Data Duplication Procedure, page 4-24](#).

Configure the replication scheme for the data store replication to the redundant machine.


**Note**

This CLU must run on both machines, and as user **pcube**.

```
>p3db --set-rep-scheme
```

**Step 6** Create a cluster.

- a. Configure SM-A and SM-B to support a cluster.
- b. Make SM-B standby.  
Use the CLU command **p3cluster --standby**.
- c. Ensure that SM-A is active.  
Use the CLU command **p3cluster --active**.
- d. Configure the VCS.
- e. Run the VCS on the setup.

**Step 7** Configure the LEG applications to send logins to the cluster virtual IP.

---

## Upgrading Cluster Setups

- [Upgrading from a Cluster Setup Version 3.x, page 4-31](#)
- [Upgrading from a Cluster Setup Version 2.x to Version 3.0 or 3.1, page 4-36](#)



**Note**

To upgrade from Version 2.x to Version 3.5.0, you must first upgrade to Version 3.0 or 3.1.

### Upgrading from a Cluster Setup Version 3.x

This section describes the basic procedure for upgrading from a cluster setup to a cluster setup, from SM version 3.0 and up.



**Note**

This procedure does not have a service down time.

The upgrade procedure when upgrading from a cluster setup involves three high level steps:

1. Perform the upgrade procedure on the standby machine.
2. Perform a manual failover on the SM that was upgraded.
3. Perform the upgrade procedure on the SM that became standby after performing the failover.

**Step 1** Configure the system kernel configuration file on both machines.

Before starting the upgrade procedure, it is necessary to configure the system kernel configuration file on both machines.

- a. Configure the system kernel configuration file on the *standby* SM.  
The configuration procedure is described in [Step 3Configure the shared memory settings, page 4-17](#).
- b. Reboot the *standby* SM.
- c. Manually trigger a failover using the Veritas cluster manager and wait until the standby SM becomes active and the active SM becomes standby.

Run the following VCS CLU command from **/opt/VRTSvcs/bin**:

```
hagrp -switch service group name to System
```

- d. Repeat steps a and b on the new *standby* SM.

**Step 2** Extract the distribution files.

Before you can upgrade the SM, you must first load and extract the distribution files on the installed machine or in a directory that is mounted to the installed machine.

- a. Download the distribution files from the Cisco web site.
- b. Use FTP to load the distribution files to the SM.
- c. Unzip the files using the **gunzip** command.

```
gunzip SM_dist_<version>_B<build number>.tar.gz
```

- d. Extract the tar the file using the **tar** command.

```
tar -xvf SM_dist_<version>_B<build number>.tar
```

**Step 3** Stop VCS monitoring.

- a. Log in as the *root* user.
- b. Stop the VCS monitoring of the SM.

Use the following VCS CLU command from `/opt/VRTSvcs/bin` to stop VCS monitoring:

```
#./hastop -local
```

**Step 4** Edit the `install-def.cfg` file.

Edit the `install-def-cfg` configuration file and set the `PermSize` and `TempSize` parameters according to the recommendations described in [Step 3 Configure the shared memory settings, page 4-17](#). For further information, see [Step 4 Edit the install-def.cfg file., page 4-18](#).

**Step 5** Pause database replication




---

**Note** Perform the following commands only when upgrading the first SM machine.

---

- a. On the *Active* machine, change directory to the location where you extracted the distribution files.
- b. Run the `p3db --rep-pause` CLU from the scripts directory.
- c. Run the `p3db --rep-status` CLU from the scripts directory and verify that replication is in 'pause' state
- d. Return to the Standby workstation.

**Step 6** Run the `cluster-upgrade.sh` script.

In order to upgrade from cluster setup to cluster setup, Subscriber Manager version 3.5.0 provides an upgrade script to perform an upgrade from previous versions. The upgrade procedure script preserves the subscriber database and the entire SM configuration, including network elements, domains, and application-specific components.




---

**Note** For Solaris: Previous versions of the SM on Solaris used a 32-bit or 64-bit Java Virtual Machine (JVM) and database. From SM version 3.0.3, the SM is installed with a 64-bit JVM and database. There is no choice as to whether to upgrade to 64-bit.

---




---

**Note** For Linux: Upgrade on Linux systems is supported only from SM 3.x releases. The Linux platform is used only with a 32-bit JVM and database.

---

- a. From the standby machine shell prompt, run the `cluster-upgrade.sh` script.

```
cluster-upgrade.sh [command-options]
```

The script performs the following steps:

- Detects existing SM version.
- Detects new SM version.
- Verifies that Java is installed on the machine.
- Verifies that the user `pcube` exists.
- Verifies that an SM of version 3.x or later is present on the system.
- Verifies the values configured in the `install-def.cfg` (if any exist).



- Stops the current SM (if running).
- Backs up existing contents of the subscriber database to an external file.
- Removes the TimesTen database.
- Backs up SM configuration files.
- Installs the updated versions of SM and TimesTen.
- Invokes a separate program for upgrading the SM and database configuration files.
- Restores the backed up contents of the subscriber database. When activated on the second machine, the script copies the contents of the database from the currently active SM since the currently active SM contains the most up-to-date data.

Table 4-14 lists the cluster-upgrade script command options.

**Table 4-14 Options for cluster-upgrade.sh**

| Options | Description                                                       |
|---------|-------------------------------------------------------------------|
| -h      | Shows this message.                                               |
| -1      | Use this option when activating the script on the first machine.  |
| -2      | Use this option when activating the script on the second machine. |

#### Examples for Running the cluster-upgrade.sh Script

- [Upgrading the First SM: Example, page 4-33](#)
- [Upgrading the Second SM: Example, page 4-33](#)

#### Upgrading the First SM: Example

This example upgrades the first SM and keeps the current database.

```
cluster-upgrade.sh -1
```



#### Note

An SM reboot is not required after the upgrade procedure.

#### Upgrading the Second SM: Example

This example upgrades the second SM and keeps the current database.

```
cluster-upgrade.sh -2
```

Do not start the SM after running **cluster-upgrade.sh**.

**Step 7** Wait until the cluster-upgrade.sh script finishes.

**Step 8** Continue database replication

- On the *Active* machine, change directory to the location where you extracted the distribution files.
- Run the **p3db --rep-continue** CLU from the scripts directory.
- Run the **p3db --rep-status** CLU from the scripts directory and verify that replication is in the 'start' state
- Return to the Standby workstation.



#### Note

Run this command only when upgrading the first machine and only if [Step 5](#) was performed.

**Step 9** Verify that changed data has been replicated.

Wait until all the data that was changed while the upgrade script was running has been replicated:

- On the active SM add a dummy subscriber using the **p3subs** CLU:

```
>p3subs --add -s dummySub
```



**Note**

When upgrading the second SM add a subscriber with a name other than **dummySub** since it was added during the upgrade of the first SM due to the replication.

- On the standby SM run the **verify-subscriber.sh** script to verify the subscriber was replicated:

```
#!/verify-subscriber.sh dummySub
```



**Note**

The **verify-subscriber.sh** script should be run as the *root* user.

**Step 10** From your machine shell prompt, run the **install-vcs-agents.sh** script. For additional information on running the `install-vcs-agents.sh` script refer to the [“Installing SM Cluster Agents”](#) section on page 4-23.

```
install-vcs-agents.sh [command-options]
```

**Step 11** Restart VCS monitoring.

Run the following VCS CLU command from `/opt/VRTSvcs/bin`:

```
#!/hastart
```

VCS monitoring starts the SM process automatically in the Initialization state.

Use the **p3sm** CLU to check when the SM is up.

```
>p3sm --sm-status
```

Use the **p3cluster** CLU to set the SM to standby state:

```
>p3cluster --standby
```



**Note**

The SM boot time after the upgrade will be longer than usual due to the extra time taken to initialize the database indexes.

**Step 12** (Optional) Install the MPLS/VPN BGP LEG.

For further information, see the [Cisco SCMS SM LEGs User Guide](#), the [“Installing the MPLS/VPN BGP LEG”](#) chapter.

**Step 13** Manually trigger a failover using the Veritas cluster manager and wait until the standby SM becomes active and the active SM becomes standby.

Run the following VCS CLU command from `/opt/VRTSvcs/bin`:

```
hagr -switch service group name -to System
```

For further information about the **hagr** CLU refer to your Veritas Cluster Server documentation.

After performing the manual failover, the standby SM on which you perform the upgrade procedure becomes the active SM. The previous active SM becomes the new standby SM.

**Step 14** Repeat the upgrade procedure on the standby SM.

To upgrade the second SM, repeat the procedure from [Step 2](#) “Extract the distribution files”.



**Note** When upgrading the second SM, do not perform [Step 5](#) “Pause database replication” or [Step 8](#) “Continue database replication”.

**Step 15** Upgrade the database replication protocol version.



**Note** You must perform the following commands as the admin user and you must perform them on both machines to upgrade the database replication protocol version. You must perform this operation after both SMs are upgraded.

a. Stop the VCS monitoring of the standby SM.

Use the following VCS CLU command from /opt/VRTSvcs/bin:

```
#./hastop -local
```

b. Upgrade replication protocol.

On the standby SM run the following CLU:

```
p3db --upgrade-rep-protocol
```

This command performs the following actions:

- Removes the DB security flag
- Stops the SM
- Restarts the DB daemon
- Starts the SM
- Starts SM replication

c. Restart VCS monitoring.

Run the following VCS CLU command from /opt/VRTSvcs/bin:

```
#./hastart
```

VCS monitoring starts the SM process automatically in the Initialization state.

d. Use the **p3cluster** CLU to set the SM to standby state:

```
>p3cluster --standby
```

e. Manually trigger a failover using the Veritas cluster manager and wait until the standby SM becomes active and the active SM becomes standby.

Run the following VCS CLU command from /opt/VRTSvcs/bin:

```
hagrps -switch service group name -to System
```

For further information about the **hagrps** CLU refer to your Veritas Cluster Server documentation.

f. Repeat the upgrade procedure on the standby SM.

**Step 16** Add a user for PRPC authentication.

If upgrading from a version of the SM prior to 3.0.5, it is necessary to add a user for PRPC authentication because SCA BB requires a username and password when connecting to the SM.

To add a user for PRPC authentication, use the **p3rpc** CLU. For example:

```
>p3rpc --set-user --username=username --password=password --remote=OTHER_SM_IP[:port]
```

**Step 17** Configure the SCE platforms

If using a cascade SCE setup, configure the cascade SCE pair in the **p3sm.cfg** file as described in the [SCE.XXX Section, page A-18](#).

**Step 18** Remove the dummy subscribers

After successfully upgrading both SMs it is recommended to remove the dummy subscribers that were added in order to verify the replication during the upgrade.

On the new *active* SM run the following CLU:

```
>p3subs --remove -subscriber=first dummy subscriber name
```

```
>p3subs --remove -subscriber=second dummy subscriber name
```

**Upgrading from a Cluster Setup Version 2.x to Version 3.0 or 3.1**

This section describes the basic procedure for upgrading from a cluster setup to a cluster setup, from SM versions 2.x.

**Note**

To upgrade to Version 3.5, you must first upgrade to Version 3.0 or 3.1.

**Note**

This procedure has a service down time.

The upgrade procedure when upgrading from a cluster setup involves three high level steps:

1. Perform the upgrade procedure on the standby machine.
2. Perform a manual failover on the SM that was upgraded.
3. Perform the upgrade procedure on the SM that became standby after performing the failover.

**Step 1** Configure the system kernel configuration file on both machines.

Before starting the upgrade procedure, it is necessary to configure the system kernel configuration file on both machines.

- a. Configure the system kernel configuration file on the *standby* SM.

The configuration procedure is described in [Step 3 Configure the shared memory settings, page 4-17](#).

- b. Reboot the *standby* SM.
- c. Manually trigger a failover using the Veritas cluster manager and wait until the standby SM becomes active and the active SM becomes standby.

Run the following VCS CLU command from **/opt/VRTSvcs/bin**:

```
hagr -switch service group name to System
```

- d. Repeat steps a and b on the new *standby* SM.

**Step 2** Extract the distribution files.

Before you can upgrade the SM, you must first load and extract the distribution files on the installed machine or in a directory that is mounted to the installed machine.

- a. Download the distribution files from the Cisco web site.
- b. Use FTP to load the distribution files to the SM.
- c. Unzip the files using the **gunzip** command.

```
gunzip SM_dist_<version>_B<build number>.tar.gz
```

- d. Extract the tar the file using the **tar** command.

```
tar -xvf SM_dist_<version>_B<build number>.tar
```

**Step 3** Uninstall the VCS agents and stop VCS monitoring.

- a. Log in as the *root* user.
- b. Uninstall the VCS agents.

Uninstalling the VCS agents is described in [Uninstalling VCS Agents, page 4-43](#). The resource names to use are *PcubeSm*, *OnOnlyProcess*, and *TimesTenRep*.

- c. Stop the VCS monitoring of the SM.

Use the following VCS CLU command from **/opt/VRTSvcs/bin** to stop VCS monitoring:

```
./hastop -local
```

**Step 4** Disable state exchange.

Disable the state exchange between the SM and the SCE platform by editing the SM configuration file (**p3sm.cfg**) and set **save\_subscriber\_state=false**, then load the configuration file using the following command:



**Note**

You must use this CLU as user **pcube**.

```
>p3sm --load-config
```

**Step 5** Drop the old replication scheme.

Use the following CLU:



**Note** You must use this CLU as user **pcube**.

```
>p3sm --drop-rep-scheme
```

**Step 6** Edit the **install-def.cfg** file.

Edit the install-def-cfg configuration file and set the PermSize and TempSize parameters according to the recommendations described in [Step 3 Configure the shared memory settings, page 4-17](#). For further information, see [Step 4 Edit the install-def.cfg file., page 4-18](#).

**Step 7** Run the upgrade-sm.sh script.

For further information, see [Step 4 Run the upgrade-sm.sh script., page 4-25](#).

**Step 8** Upgrade the application and LEGs.

Perform the specific upgrade instructions of your application or LEGs.

**Step 9** Configure the replication scheme.

Configure the replication scheme for the datastore replication to the redundant machine using the following CLU:



**Note** You must use this CLU as user **pcube**.

```
>p3sm --set-rep-scheme
```

**Step 10** Install the VCS agents and configure and restart VCS monitoring

a. Install the SM VCS agents.

Installing the SM VCS agents is described in [Installing SM Cluster Agents, page 4-23](#).

b. Configure the VCS.

Configuration of the VCS is described in [Veritas Cluster Server, page E-1](#).

c. Restart VCS monitoring.

Run the following VCS CLU command from **/opt/VRTSvcs/bin**:

```
#!/hastart
```

**Step 11** Remove obsolete state information.

Remove any obsolete subscriber state information, by running the SM CLU as **pcube** user:

```
>p3subsdm --clear-all-states
```

- Step 12** Remove obsolete subscriber properties (Method A)  
Remove any obsolete subscriber properties.



**Note** All CLU commands must be run as user **pcube**.

- a. Export any existing subscribers to a csv file.  
`>p3subsdb --export -o csv-file`
- b. Clear the subscriber database.  
`>p3subsdb --clear-all`
- c. Remove any obsolete properties from the csv-file.  
See [Table 4-13](#) for a list of properties to be removed.
- d. Import the subscribers from the revised file.  
`>p3subsdb --import -f csv-file`

- Step 13** Remove obsolete subscriber properties (Method B)  
If upgrading from version 2.x, remove any obsolete subscriber properties.



**Note** All CLU commands must be run as user **pcube**.

- a. Remove the obsolete properties from the SM database by running the **p3subsdb** command.  
`>p3subsdb --remove-property --property=prop`  
The obsolete properties to be removed are listed in [Table 4-13](#).
- b. Resynchronize all SCEs.  
`>p3sm --resync-all`

- Step 14** Manually trigger a failover

Manually trigger a failover using the Veritas cluster manager and wait until the standby SM becomes active and the active SM becomes standby.

Run the following VCS CLU command from **/opt/VRTSvcs/bin**:

```
hagrps -switch service group name to System
```

- Step 15** Repeat the upgrade procedure on the standby SM.

After performing the manual failover (see [Step 14 Manually trigger a failover, page 4-39](#)), the standby SM on which you perform the upgrade procedure becomes the active SM. The previous active SM becomes the new standby SM.

To upgrade the second SM, repeat the procedure from [Step 2 Extract the distribution files.](#) to [Step 12 Remove obsolete subscriber properties \(Method A\).](#)

**Step 16** Add a user for PRPC authentication.

It is necessary to add a user for PRPC authentication because SCA BB requires a username and password when connecting to the SM.

To add a user for PRPC authentication, use the **p3rpc** CLU. For example:

```
>p3rpc --set-user --username=username --password=password --remote=OTHER_SM_IP[:port]
```

**Step 17** Configure the SCE platforms

If using a cascade SCE setup, configure the cascade SCE pair in the **p3sm.cfg** file as described in the [SCE.XXX Section, page A-18](#).

---



## Additional Upgrade Procedures

### Upgrading SubscriberID Maximum Length to 64 Characters

In version 3.0.5, the length of the SubscriberID was increased to 64 characters. For new installations the maximum length of the SubscriberID is 64 characters. However, when upgrading from earlier versions, the length is not increased automatically.

- 
- Step 1** Export the subscriber database using the **p3subsdb** CLU.
- ```
>p3subsdb --export --output=output file name
```
- Step 2** Destroy the database using the **p3db** CLU.
- ```
>p3db --destroy-rep-db
```
- Step 3** Restart the SM.
- ```
>p3sm --restart
```
- Step 4** Import the subscribers back into the database using the **p3subsdb** CLU.
- ```
>p3subsdb --import --file=file name from Step 1
```

**Note**

This procedure requires system downtime because the SM database is destroyed. Moreover, after the restart, all the SCEs will automatically lose all the subscriber information and it will be restored only after the subscribers are imported back into the SM database.

---

## How to Downgrade the Subscriber Manager

This section describes the procedure to downgrade the SM to a previous version.

- 
- Step 1** Perform the uninstall procedure described in [How to Uninstall the Subscriber Manager, page 4-41](#).
- Step 2** Perform the installation procedure described in [Installing the Subscriber Manager, page 4-13](#).

**Note**

The **upgrade-sm.sh** and **cluster-upgrade.sh** upgrade scripts do not support SM downgrade.

---

## How to Uninstall the Subscriber Manager

- [Uninstalling the Subscriber Manager, page 4-42](#)
- [Uninstalling VCS Agents, page 4-43](#)

## Uninstalling the Subscriber Manager

### uninstall-sm.sh Script

To execute the **uninstall-sm.sh** script, from your machine shell prompt, enter the following command:

```
uninstall-sm.sh [command-options]
```

Table 4-15 lists the command options:

**Table 4-15** Options for *uninstall-sm.sh* Script

| Options | Description                |
|---------|----------------------------|
| -n      | Do not remove SM database. |
| -h      | Shows the help message     |

The script performs the following steps:

- Stops the SM.
- Stops the replication agent (in cluster setups) if the **-n** flag is not used.
- Destroys the data-stores if the **-n** flag is not used.
- Uninstalls the TimesTen database.
- Removes the SM directories and boot files.
- Removes the Java that was installed as part of the SM installation.

- 
- Step 1** If using a cluster setup, stop the VCS monitoring of the SM.  
Stop the VCS monitoring by running the following VCS CLU command from **/opt/VRTSvcs/bin**:
- ```
#./hastop -local
```
- Step 2** Run the **uninstall-sm.sh** script from the distribution root directory.
- ```
#./uninstall-sm.sh
```
- For further information, see [uninstall-sm.sh Script, page 4-42](#)
- Step 3** If using a cluster setup, start the VCS monitoring.  
Start the VCS monitoring by running the following VCS CLU command from **/opt/VRTSvcs/bin**:
- ```
#./hastart
```
- Step 4** If using a cluster setup, remove the Veritas Cluster agents.
Removal of the Veritas Cluster agents is described in [Uninstalling VCS Agents, page 4-43](#).
Remove the following resource names: OnOnlyProcess, SubscriberManager, and TimesTenRep.
- Step 5** Remove the **pcube** user, by running the **userdel** command.
- ```
userdel -r pcube
```



**Note** If you chose to keep TimesTen installed, do not remove the **pcube** user.

---

## Uninstalling VCS Agents

Repeat the following procedure for each Veritas Cluster agent that you wish to remove.

- Step 1** Remove the VCS agents by using the Veritas Cluster Manager or by using the **hares** CLU.

The VCS agents can be removed using the Veritas Cluster Manager or the following CLU: (The resource names in your system might have different names, use **hares -list** to see the existing resource names).

```
hares -delete TimesTenDaemon
hares -delete SM
hares -delete ReplicationAgent
hares -delete Network-NICs
hares -delete Network-VIP
```

- Step 2** Remove the VCS resource types by using the **hatype** CLU.

The type names in your system might have different names, use **hatype -list** to see the existing type names.

```
hatype -delete OnOnlyProcess
hatype -delete SubscriberManager
hatype -delete TimesTenRep
```

- Step 3** Delete the VCS agent from the disk.

Use the following command to delete the VCS agent:

```
rm -rf /opt/VRTSvcs/bin/OnOnlyProcess
rm -rf /opt/VRTSvcs/bin/SubscriberManager
rm -rf /opt/VRTSvcs/bin/TimesTenRep
```

