



Troubleshooting SESM Installation and Configuration

This chapter provides some help with troubleshooting problems in a Cisco Subscriber Edge Services Manager (SESM) deployment. It includes the following topics:

- [Diagnosing Problems, page 10-1](#)
- [Troubleshooting Aids, page 10-4](#)
- [Troubleshooting Tips, page 10-7](#)

Diagnosing Problems

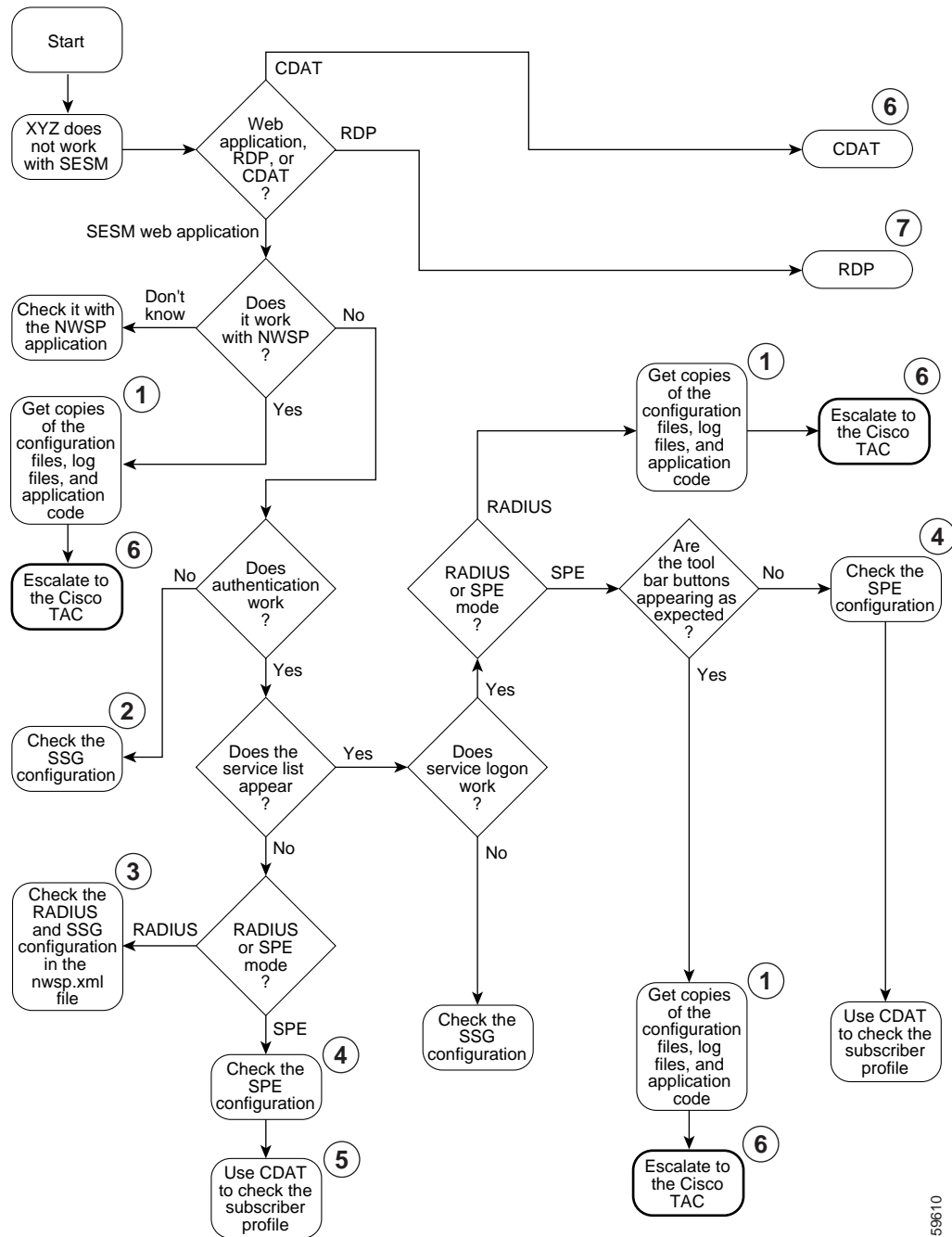
This section contains procedural charts that show you how to research a problem and identify the general area of the problem before escalating it to the Cisco Technical Assistance Center. The section includes the following procedures:

- [Procedures for Troubleshooting an SESM Web Application, page 10-1](#)
- [Procedures for Troubleshooting RDP, page 10-3](#)

Procedures for Troubleshooting an SESM Web Application

[Figure 10-1](#) shows a procedure for analyzing problems in an SESM web application. The numbered callouts are keyed to the table that follows the figure.

Figure 10-1 Procedures for Troubleshooting an SESM Web Application



59610

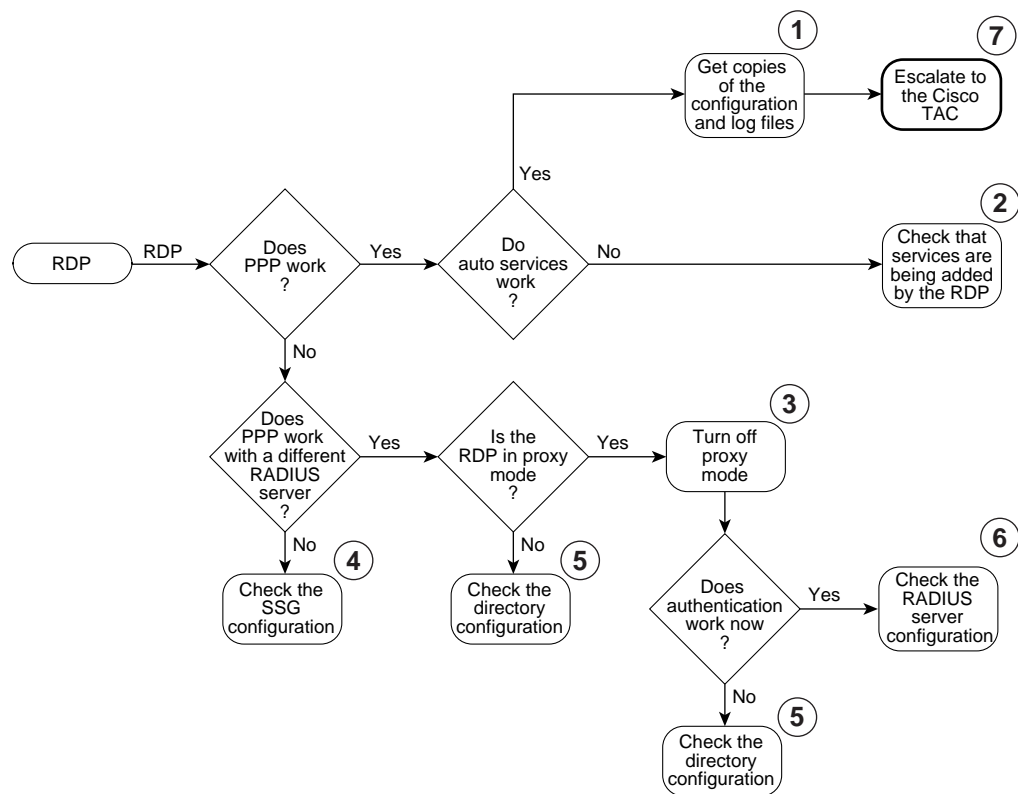
1	See Configuration File Summary , page 6-3 and Logging and Debugging Mechanisms , page 10-4.
2	See Configuring the SSG , page B-1.
3	See the SSG and AAA objects in Table 6-4 on page 6-16.
4	See Configuring SPE , page 6-37.

5	Make sure the subscriber is subscribed to services and has the proper privileges to access those services. See the URL for CDAT documentation: http://www.cisco.com/univercd/cc/td/doc/solution/sesm/sesm_311/toolgd/index.htm
6	See Obtaining Technical Assistance , page xvi.
7	See Procedures for Troubleshooting RDP , page 10-3.

Procedures for Troubleshooting RDP

Figure 10-2 shows a procedure for analyzing problems in RDP. The numbered callouts are keyed to the table that follows the figure.

Figure 10-2 Procedures for Troubleshooting RDP



59611

1	See Configuration File Summary , page 6-3 and Logging and Debugging Mechanisms , page 10-4.
2	See the RDPPacketFactory object in Table 6-6 on page 6-32.
3	See RDP Modes , page 6-30.
4	See Configuring the SSG , page B-1.
5	See Configuring SPE , page 6-37.
6	See the AAA object in Table 6-4 on page 6-16.
7	See Obtaining Technical Assistance , page xvi.

Troubleshooting Aids

This section describes some facilities that might be useful in troubleshooting SESM installation and configuration problems. It includes the following topics:

- [Logging and Debugging Mechanisms, page 10-4](#)
- [Java Command Line Options, page 10-6](#)
- [SESM Management Console, page 10-6](#)
- [Obtaining License and Version Information, page 10-7](#)

Logging and Debugging Mechanisms

This section describes the logging and debugging options available to help troubleshoot SESM applications and deployment. The logging and debugging mechanisms are MBeans which are configured in the MBean configuration files. By changing the configuration of the logging and debugging mechanisms, you can change the amount of detail reported and specify message filtering. The topics in this section are:

- [Log File Locations, page 10-4](#)
- [Logging and Debugging in SESM Web Applications, page 10-4](#)
- [Switching Debugging On and Off at Run Time, page 10-5](#)
- [Logging and Debugging in RDP, page 10-5](#)
- [Logging and Debugging in CDAT, page 10-6](#)

Log File Locations

The log files for the SESM web applications, RDP, and CDAT, are located in log directories. All of the configuration files use the `application.log` Java system property to set the location of the log directory. Java system properties are set by the application startup scripts.

See [Table 7-1 on page 7-5, “Java System Properties in Startup Scripts”](#) for information about how the application startup scripts set `application.log`, the default values of `application.log`, and how to change the default.

Logging and Debugging in SESM Web Applications

You can use the SESM application’s log files to troubleshoot problems. Two of the log files have debugging mechanisms that you can configure along with the logging features.

An SESM web application and its Jetty container write to the following log files:

- Jetty container’s HTTP request log—This log file records all incoming HTTP requests. You can use this log file to analyze volume and traffic patterns for the web server.
The default name for this file is `date.request.log`. For information on configuring this log, including file name and retention period, see the “Jetty Log Sink” object in [Table 6-3 on page 6-9](#).
- Jetty container’s application log—This log file records logging and debugging messages. The logging messages record the startup of the Jetty server and all ongoing activity, such as errors trapped by the Jetty server and HTTP errors. If the SESM application fails to start, look at this log.

Make sure you monitor this log file for illegal HTTP requests that might indicate attempts to subvert the web server. If you enable debugging, the log file also includes more detailed debugging messages.

The default name for this file is *date.jetty.log*. For information on configuring this log, including file name, retention period, and contents, see:

- The “Log” object in [Table 6-3 on page 6-9](#), to configure the container’s log file
- The “DebugMBean” object in [Table 6-3 on page 6-9](#), to configure the container’s debugging features
- SESM web application log—This log file records logging and debugging messages. The logging tool logs SESM web application activity. The debugging mechanism produces messages useful to developers in debugging applications.

The default name for this file is *date.application.log*. See the “Logger” object in [Table 6-4 on page 6-16](#) for information on configuring this file, including its file name and retention period, whether debugging is turned on or off, and the content of logging and debugging messages.

Configure the container logs in the container MBean configuration file for the application:

```
jetty
  config
    nwsp.jetty.xml
    wap.jetty.xml
```

Configure the application log in the application MBean configuration file:

```
nwsp
  config
    nwsp.xml
```

Switching Debugging On and Off at Run Time

The debug attribute in the Logger MBean in the application configuration file controls whether debug is on or off. The attribute is a Java system property:

```
nwsp.debug
```

You can switch debugging on or off at the command line to override the default value in the XML file or use the `-D` option on the command line when you start the portal application.

Logging and Debugging in RDP

The RDP application writes to the RDP application log file, which records logging and debugging messages. The logging feature logs RDP application activity. The debugging mechanism might be useful in debugging packet handling. The default name for this log file is *date.application.log*.

RDP uses the same logging facility that the SESM web applications use. Both use a Logger MBean to configure the logging and debugging features. For RDP, you configure the Logger MBean in the RDP configuration file:

```
rdp
  config
    rdp.xml
```

See the “Logger” object in [Table 6-4 on page 6-16](#) for information about configuring the Logger MBean.

Logging and Debugging in CDAT

CDAT is a web application running in a J2EE container, similar to the SESM web applications. The CDAT application and its Jetty container write to the same types of log files as the SESM web applications, as described previously in the “[Logging and Debugging in SESM Web Applications](#)” section on page 10-4.

Configure the container logs in the CDAT-specific container MBean configuration file:

```
jetty
  config
    cdat.jetty.xml
```

Configure the application log in the CDAT MBean configuration file:

```
cdat
  config
    cdat.xml
```

Java Command Line Options

When you execute a startup script that includes the java command, you can specify any Java option on the command line. To specify Java options, use `-jvm` as an option on the command line. For example, you can add the following option to the command line when you execute the SESM application startup script:

```
-jvm -Djava.compiler=NONE
```

SESM Management Console

The Sun example JMX server, which is the JMX server installed with the Jetty component in the SESM installation package, includes a JMX HTML adaptor. SESM uses the adaptor to produce a management console that shows the current value of all MBean attributes in all of the MBean configuration files.



Note

This JMX HTML adaptor is not production quality. For example, configuration changes made using this console are not persistent. You should remove it from your configuration files before transitioning the SESM application to public use. See the “ManagementConsole” object in [Table 6-4 on page 6-16](#) for information about configuring and removing this adaptor.

You can access the SESM management console on a web browser at the following URL:

```
http://SESMserver:managementPortNumber/
```

Where:

SESMserver—Host name or IP address of the workstation on which SESM is installed.

managementPortNumber—The port configured in the `HtmlAdaptorServer` MBean in the application configuration file. The default management port number used by the SESM installation program is:

```
applicationPortNumber + 100
```

For example, for NWSP, the installer uses a default application port number of 8080 and a corresponding management port number of 8180.

Management Console User Name and Password

Before you gain access to the management console, you must enter a valid user name and password. Enter the values that match the values in the ManagementConsole MBean in the application's configuration file. See the "ManagementConsole" object in [Table 6-4 on page 6-16](#) for more information.

Obtaining License and Version Information

If you purchased SESM, your license number is available on the License Certificate shipped with the product. If you have not purchased SESM, you can install an evaluation copy of the software without a license number. An evaluation installation provides full software functionality. Although the evaluation options do not have an expiration period, you must obtain a license before deploying SESM in a production environment.

The installation program records the license number and the software version you installed in the licensenum.txt file under the installation directory.

Troubleshooting Tips

This section contains some hints that might help you identify and fix problems in SESM. The problems are divided into the following topics:

- [JRE and JDK Troubleshooting, page 10-7](#)
- [Installation Troubleshooting, page 10-10](#)
- [Configuration File Location Troubleshooting, page 10-11](#)
- [SESM Configuration Troubleshooting, page 10-11](#)
- [RADIUS Configuration Troubleshooting, page 10-12](#)
- [SSG Configuration Troubleshooting, page 10-12](#)

JRE and JDK Troubleshooting

If the installer does not find an appropriate JRE, it installs the bundled JRE Version 1.2.2.

This section contains the following topics:

- [Warning and Error Messages after JRE Installation, page 10-7](#)
- [Searching for an Existing JDK or JRE, page 10-8](#)
- [Using a Pre-installed JRE or JDK, page 10-9](#)
- [Recompiling a Customized JSP, page 10-9](#)

Warning and Error Messages after JRE Installation

The JRE installation might produce warning messages and nonfatal error messages. These messages are expected and normal.

- The warning message states that JSPs will not be compiled. You do not need to recompile JSPs to run the NWSP application.

If you are a Web developer expecting to write new JSPs or change the NWSP JSPs, you must load the Java Development Kit (JDK). To obtain a recent JDK, go to:

<http://java.sun.com/products/j2se>

- The nonfatal JIT relocation error message is the result of a problem within the bundled JVM obtained from Sun Microsystems. It does not affect SESM operation. You can ignore this message and all supporting information.

Searching for an Existing JDK or JRE

The installer does the following when searching for a valid JDK or JRE:

1. It searches for a JDK Version 1.2.2 that is already installed.
2. Failing that, it searches for a JRE Version 1.2.2 or later that is already installed.
3. Failing that, it installs and uses the bundled JRE Version 1.2.2.

In some cases, even though a JRE is installed, the installer may not find it or finds a different JRE

On Windows NT, the installer looks in the NT Registry for the location of a JDK or JRE. It uses Java Version 1.2.2 in preference to Version 1.3.

On Solaris, the installer looks in the following well-known locations before installing the bundled JRE:

usr/jre	/opt/jre	/usr/jre1.3	/opt/jre1.3
/usr/jre1.2.2	/opt/jre1.2.2	/usr/jre1.3.0	/opt/jre1.3.0
/usr/java1.2	/opt/java	/usr/java1.3	/opt/java
/usr/java	/opt/java1.2	/usr/java	/opt/java1.3
/usr/java1.2.2	/opt/java1.2.2	/usr/java1.3.0	/opt/java1.3.0
/usr/jdk	/opt/jdk	/usr/jdk	/opt/jdk
/usr/jdk1.2	/opt/jdk1.2	/usr/jdk1.3	/opt/jdk1.3
/usr/jdk1.2.2	/opt/jdk1.2.2	/usr/jdk1.3.0	/opt/jdk1.3.0

On Linux, the installer looks in the following well-known locations before installing the bundled JRE:

sun.java.1.2.2.jvm:	sun.java.1.3.x.jvm:	blackdownjdk122:
/usr/jre	/usr/jre1.3	/usr/jdk1.2.2
/usr/jre1.2.2	/usr/jre1.3.1	/usr/local/jdk1.2.2
/usr/java1.2	/usr/java1.3	/opt/jdk1.2.2
/usr/java	/usr/java	
/usr/java1.2.2	/usr/java1.3.1	blackdownjdk13.jvm:
/usr/jdk	/usr/jdk	/usr/j2sdk1.3
/usr/jdk1.2	/usr/jdk1.3	/usr/local/j2sdk1.3
/usr/jdk1.2.2	/usr/jdk1.3.1	/opt/j2sdk1.3
/opt/jre	/opt/jre1.3	
/opt/jre1.2.2	/opt/jre1.3.1	ibmjdk13.jvm:
/opt/java	/opt/java	/opt/IBMJava2-13/
/opt/java1.2	/opt/java1.3	/usr/IBMJava2-13/
/opt/java1.2.2	/opt/java1.3.1	/usr/local/IBMJava2-13/
/opt/jdk	/opt/jdk	/opt/jdk13
/opt/jdk1.2	/opt/jdk1.3	/usr/jdk13
/opt/jdk1.2.2	/opt/jdk1.3.1	/usr/local/jdk13

Using a Pre-installed JRE or JDK

On either platform, you can specify the location of a pre-installed JRE or JDK by starting the installation process on a command line and specifying the `javahome` parameter, as follows:

```
installImageName -is:javahome location
```

Where:

installImageName is the name of the SESM downloaded image.

location is the path name for the JRE or JDK.

Recompiling a Customized JSP

If you do not see changes that you make to a JSP, follow these procedures:

-
- Step 1** Install a JDK (Version 1.2.2 or later).
 - Step 2** Edit the application start script so that it uses the JDK, rather than the JRE (for example, edit `jetty/bin/start.sh`).
 - Step 3** Ensure that `JDK_HOME` points to the directory into which you installed the JDK.
 - Step 4** Stop the SESM application.
 - Step 5** Change directories to the application's WEB-INF directory. For example, enter:

```
cd installDir/nwsp/docroot/WEB-INF
```

- Step 6** In the WEB-INF directory, back up the web.xml file by renaming it. For example, enter:
- ```
cp web.xml web.xml.bak
```
- Step 7** In the WEB-INF directory, copy the web.recompile.xml file over web.xml. For example, enter:
- ```
cp web.recompile.xml web.xml
```
- Step 8** Restart the SESM application.
-

The installed web.xml file points to precompiled versions of the JSPs. It does *not* reference the JSPs in /nwsp/docroot. Thus, changing the JSPs in docroot has no effect if you use the installed web.xml file.

The web.recompile.xml file references the JSPs in /nwsp/docroot, rather than using the precompiled JSPs.

Installation Troubleshooting

This section describes some potential problems that you might encounter during installation.

No X Server for a Solaris Installation

To install SESM on a Solaris server with no X server, use the Silent or Console installation modes.

Incorrect Permissions

The SESM installation program writes to parts of the file system or Windows registry that are only accessible to a privileged user (that is, root on Solaris, or a member of the Administrators group on Windows NT). An SESM installation must be performed by a privileged user who has access to these resources. Otherwise, the outcome of the installation is unpredictable.

Files Not Found

If you receive Java error messages indicating missing files in system level directories (for example, /var, on Solaris), you do not have correct permissions to perform the installation. See the preceding [“Incorrect Permissions”](#) section.

Incomplete Installation or Files Installed in Incorrect Directory

On a Solaris system, if you remove the contents of the SESM installation directory using the **rm** command instead of uninstalling SESM using the uninstall utility, then subsequent installations of SESM into the same directory might be adversely affected.

Uninstall SESM using uninstall.bin. If uninstalling is not possible, before reinstalling SESM, delete the vpd.properties file from the home directory of the person who is performing the installation.



Note

If you deploy multiple SESM installations, and you delete the vpd.properties file to recover from removing one of the installations, then you cannot use uninstall.bin to uninstall any of the other installations.

Configuration File Location Troubleshooting

The SESM installation program places the J2EE web server and SESM configuration files in the correct directories as defined in the startup scripts. If the configuration files are moved for any reason, then you must edit the web.xml file to reflect the new locations.

SESM Configuration Troubleshooting

If the SESM software is installed correctly, and all of the configuration files are in the proper location, but the SESM web application does not function, then examine the configuration values in the SESM application's MBean configuration file (for example, nwsp/config/nwsp.xml).

Communication with SSG

If the SSG port number or shared secret specified in the SESM application's MBean configuration file does not match actual SSG configuration (as performed on the SSG host), the SSG cannot see the SESM requests or is unable to decrypt the requests because the shared secret does not match. When the shared secret does not match, the SSG returns an Access Reject message.

For more information on SSG configuration, see [Appendix B, "Configuring the SSG."](#)

Communication with RADIUS Server

If incorrect IP addresses or port numbers are specified in the SESM application's MBean configuration file for the primary and secondary RADIUS servers, the RADIUS servers cannot see the SESM requests.

If the IP addresses and port numbers are correct, the RADIUS server returns an Access Reject when either of the following errors is present:

- The shared secret specified for the RADIUS server in the application's MBean configuration file is not correct.
- The SESM web application is not properly configured as a RADIUS client.

For more information on RADIUS configuration, see [Appendix D, "Configuring RADIUS."](#)

Out of Memory Exceptions

Out of memory exceptions might indicate that there is not enough Java virtual memory reserved to handle the number of users currently logged on.

The generic startup script sets the Java virtual memory (VM) size to 64 MB. To change this value, stop the application, edit the generic start script (start.sh or start.cmd), and restart the application.

Web Server Unavailable

Messages stating that the web server is unavailable might indicate that there is not enough Java virtual memory reserved to handle the number of users currently logged on. Follow the instructions in the ["Out of Memory Exceptions" section on page 10-11](#) to increase Java virtual memory.

RADIUS Configuration Troubleshooting

The RADIUS server must be configured to recognize the following two clients:

- SESM web application
- SSG

If either of these configuration items is incorrect, then the RADIUS server sends Access Reject messages in response to all requests. See the [“Configuring RADIUS Clients” section on page D-2](#) for information on configuring these RADIUS clients.

For service profile requests, the password for service and service group profiles must match those defined for the SSG and the SESM application. This password is used in Access Request messages for profiles, where the profile name is the service or service group name and the password is as defined in the following two locations:

- The servicePassword attribute in the AAA section of the SESM application’s MBean configuration file
- The service-password parameter for the SSG

SSG Configuration Troubleshooting

The SSG must have a default network location defined, from which the SESM web application is accessible. Otherwise, client requests never reach the SESM application, and the client browser eventually times out.

The SSG must have the radius-helper parameters configured with the correct port numbers and shared secret so that the SSG can see SESM messages and decrypt them. Because the SSG carries out authentication against the RADIUS server, it must also have the correct values defined for the radius-server parameters.