



Operating and Maintaining Cisco SPS

Operating and maintaining Cisco SIP proxy server (Cisco SPS) involves starting and stopping the system and database, working with system logs, and backing up and restoring the system. This chapter contains the following sections:

- [How to Operate Cisco SPS, page 3-2](#)
- [How to Operate MySQL, page 3-2](#)
- [How to Manage Debug Log Files, page 3-3](#)
- [How to Back Up and Restore Cisco SPS, page 3-8](#)
- [How to Restore a MySQL Database, page 3-10](#)
- [How to Replace, Upgrade, or Delete a Cisco SPS License, page 3-14](#)



Note

- For background information on concepts relevant to this chapter, see [Chapter 1, “Cisco SPS Overview.”](#)
- For additional information on use of the GUI-based provisioning system, see [Chapter 2, “Configuring Cisco SPS.”](#)
- For troubleshooting information, see [Appendix A, “Troubleshooting”](#) and [Appendix F, “SIP Call-Flow Scenarios.”](#)
- For information on Apache directives, see the The Apache Software Foundation website at www.apache.org.



Tip

To access the Cisco SPS GUI-based provisioning system, use this procedure:

1. Go to the following (default) directory or your Windows desktop:

Linux: `/usr/local/sip/gui/`

Solaris: `/opt/sip/gui/`

2. Enter the **CiscoSPS** command or double-click the CiscoSPS icon to open the Cisco SPS GUI.
3. Enter your password (default is `cpsuser`).
4. During installation, did you enter the correct value for the `pserver` location?
 - If yes, click **OK**.

- If no, click **more>>**, enter the pserver host name and port number, and click **OK** (the default port is 26005).

The Cisco SPS main menu appears. The pserver host name and port number automatically reappear at the next login.

How to Operate Cisco SPS

You can start and stop Cisco SPS in any of three ways:

- Using the GUI-based provisioning system (recommended and described below)
- Using CIAgent and the CIAgent Dr-Web interface (described in [Chapter 4, “Managing Databases”](#))
- Using the sip script that is generated when you run the SPS setup (sps_setup) script (described in [Appendix C, “Manual Operation and Maintenance”](#))



Note

The results of any operation that you perform outside of the GUI-based provisioning system (such as using the sip script) are not reflected in the GUI. If such an operation was performed outside of the GUI, you might need to update the required operation manually.



Tip

Cisco SPS provides a means not only to stop and restart, but also to gracefully restart the proxy server. Graceful restart causes the system to use a new SIP directives (sipd.conf) file. The SIP proxy server (sipd) daemon (parent process) remains alive, rereads the sipd.conf file, tears down child processes as they become idle, and spawns new child processes with the new configuration. Call processing is not interrupted as a result. Graceful restart is therefore a useful way to activate a new configuration without dropping calls.

Detailed Steps

- Step 1** From the Cisco SPS main menu (see [Tip](#) for how to access), choose **Proxy Control**.
The screen displays all system proxy servers and their current running state.
- Step 2** Click the Select box next to the one or more proxy servers whose operation you want to control.
- Step 3** Click **Stop**, **Restart**, or **Graceful restart** (see tip above).
- Step 4** Click **Yes**.

How to Operate MySQL

The MySQL database starts automatically when you run the SPS setup (sps_setup) script. You can also start the database manually as described below.

User and root passwords are set when you run the script. You can, however, change the root password.

**Note**

The start and stop procedures might yield the error message “/etc/init.d/mysql: @HOSTNAME@: not found.” Ignore the message; MySQL starts and stops properly.

Detailed Steps

Step 1 Log in to Cisco SPS as root.

```
> su root
```

Step 2 To start MySQL, do either of the following:

- Use the **start** command:

```
# /etc/init.d/mysql start
```

- Use the safe MySQL daemon (safe mysqld) script:

```
Linux: # /usr/local/mysql/bin/safe_mysqld &
```

```
Solaris: # /opt/mysql/bin/safe_mysqld &
```

Step 3 To stop MySQL, use the **stop** command:

```
# /etc/init.d/mysql stop
```

Step 4 To change the MySQL administrator password, enter the following sequence of commands:

```
Linux: # /usr/local/mysql/bin/safe_mysqld --user=mysql &
# /usr/local/mysql/bin/mysqladmin -u root -p<old_password> password <new_password>
# /usr/local/mysql/bin/mysqladmin -p reload
```

```
Solaris: # /opt/mysql/bin/safe_mysqld --user=mysql &
# /opt/mysql/bin/mysqladmin -u root -p<old_password> password <new_password>
# /opt/mysql/bin/mysqladmin -p reload
```

How to Manage Debug Log Files

This section contains the following information:

- [Information About Log Files, page 3-4](#)
- [Setting Up Debug Logs, page 3-5](#)
- [Rotating sipd Logs, page 3-6](#)

Information About Log Files

During Cisco SPS operation, each system component writes to one or more log files (see [Table 3-1](#)).

Table 3-1 System Components and Their Log and Associated Configuration Files

Component	Component Name	Log File ¹	Associated Configuration File ²
SIP proxy server	sipd	access_log agent_log error_log refer_log stats_log	sipd.conf
SIP provisioning agent (provisioning client for sipd)	spa	spa_log	spa.conf
Provisioning server	pserver	pserver_log	ps.conf
License manager	lm	licenseMgr_log	lm.conf
Installation script	sps_setup	sps_setup_log	Cannot be configured

1. Located in the <ServerRoot>/logs directory.
2. Located in the <ServerRoot>/conf/ directory. Where possible, configure debugging using the GUI-based provisioning system rather than by manually editing these files.

Log files are text files that you can view with any text editor. They contain detailed event information in hexadecimal code. In most cases, you can configure their content and level of detail by setting the debug level or verbosity. Although you should do so using the GUI-based provisioning system as described below, you can also do so by manually editing the DebugLevel directive in the associated configuration file.

Error logs are either lengthy or abbreviated, according to whether or not debug flags are turned on.

- Lengthy format prints when a debug flag is turned on.

Debug Output Example

```
[Fri Apr 13 22:29:37 2001] sip_protocol.c(4322) Received 291 bytes UDP packets from
10.80.36.85:50117
REGISTER sip:64.102.93.77 SIP/2.0
Via:SIP/2.0/UDP 10.80.36.85:5060
From:sip:IPphone-2@64.102.93.77
To:sip:IPphone-2@64.102.93.77
Call-ID:c3943000-ee2f9c88-23f9821e-382e3031@10.80.36.85
CSeq:101 REGISTER
Contact:<sip:IPphone-2@10.80.36.85:5060>
Expires:3600
```

- Abbreviated format prints when a debug flag is turned off (default).

Debug Output Example

```
[Fri Apr 20 21:44:51 2001] [notice] A new sipd child process (27413) has started.
```

Setting Up Debug Logs

Detailed Steps

- Step 1** From the Cisco SPS main menu (see [Tip](#) for how to access), choose **Farm/Proxies > Advanced > Debug and Logs**.
- Step 2** In the Debug Flags area, click the desired debug-flag check box.
- Step 3** In the Error Log entry box, enter the path for a log file (see [Table 3-1](#)).
- Step 4** In the Log Level dropdown list, select a log level. Valid values are the following, in decreasing order of verbosity: debug (most verbose), info, notice, warn, error, crit, alert, and emerg (least verbose).
- Step 5** In the Custom Log area, add the custom log file.

A custom log is a table in the GUI that allows you to specify your own log filename in Apache style. It looks something like the following, where the entries shown are the defaults. You can change the value in the Name column to anything that you want.

File	Name
logs/access_log	access_log
logs/refer_log	refer_log
logs/agent_log	agent_log

- Step 6** In the Log Format area, add a log format as needed.
- Step 7** Enter settings for the following:
- SIP Stats Log
 - SIP Stats Intervals
 - Shared Memory Stats Log
 - Shared Memory Stats Interval



Note TransferLog is not provisionable from the GUI, but you can configure CustomLog to have the same functionality.

- Step 8** Click **Submit**.
- Step 9** Repeat the procedure as needed for additional logs.
- Step 10** As appropriate, reset log levels in each of the following files:
- Provisioning server (ps.conf) file
 - SIP provisioning agent (spa.conf) file
 - License manager (lm.conf) file

Look for the “DebugLevel” line and change the parameter from LOG_ERR to LOG_DEBUG to increase log verbosity.

- Step 11** Restart your proxy servers.

Rotating sipd Logs

Log Rotation

On even a moderately busy server, the quantity of information stored in the log files is very large. The access log file typically grows 1 MB or more per 10,000 requests. The error log file can grow even faster, depending on its configured verbosity.

It is consequently necessary to cause the system periodically to cease writing to old log files, move or delete those files, and begin writing to new log files—that is, to rotate log files. You cannot rotate files while the server is running, however, because Cisco SPS continues writing to log files as long as they are open. Instead, you must gracefully restart the server to cause it to continue to write to the old log files while it finishes serving old requests, and then to open new log files. The system must wait for some time after the graceful restart before processing the new log files.

Default names for active and rotated log files are shown in [Table 3-2](#).

Table 3-2 Rotated Log Files

Log	Active Log File	Rotated Log File
Access log	access_log	access_log_yyyymmddhhmmss
Error log	error_log	error_log_yyyymmddhhmmss

A typical scenario that simply rotates logs and moves them off the local drive is as follows:

```
# cd <server_root>/logs
# mv access_log access_log.old
# mv error_log error_log.old
# sip graceful
# sleep 600
# mv access_log.old <some remote location>
# mv error_log.old <some remote location>
```

Rather than rotate and move logs in this way, however, we recommend that you use pipes to set up periodic rotation as described below.

Piped Logs

In addition to writing directly to a file, you can direct Cisco SPS to write error and access log files through a pipe to another process, which allows you to rotate logs without restarting the server. Cisco SPS includes a simple program called `rotatlogs` for this purpose.

To write logs to a pipe, you simply replace the filename with the pipe character "|", followed by the name of the executable that is to accept log entries on its standard input.

Detailed Steps



Note

- You can rotate only error-log and access-log files that are written by the SIP proxy server (sipd).
- With periodic log rotation, new log files are not created unless traffic or logging occurs. If there is nothing to write, new log files are not created.

Step 1 From the Cisco SPS main menu (see [Tip](#) for how to access), choose **Farm/Proxies > Advanced > Debug and Logs**.

Step 2 To set up periodic rotation of error logs (in this case, every 86400 seconds or 24 hours), do either of the following:

- (GUI) In the ErrorLog field, enter the following (including the quotes) and click **Submit**:

Linux: `"|/usrlocal/sip/bin/rotatelog /usr/local/sip/logs/error_log 86400"`

Solaris: `"|/opt/sip/bin/rotatelog /opt/sip/logs/error_log 86400"`



Note You can rotate logs according to size rather than time. For example, in place of 86400 above, you can enter 5M.

- (Manual) In the SIP directives (sipd.conf) file, do the following:
 - a. Locate the line shown for GUI above and remove the comment marker at the beginning of the line.
 - b. Locate the corresponding default line that does not mention rotatelog and add a comment marker so that just one such line is left active (see [Tip](#) below).
 - c. Remove the comment marker in the ErrorLog logs/error_log line.

Step 3 To set up periodic rotation of access logs (in this case, every 86400 seconds or 24 hours), do the following:

- (GUI) In the CustomLog field, set the access_log filename to the following (including the quotes) and click **Submit**:

Linux: `"|/usr/local/sip/bin/rotatelog /usr/local/sip/logs/access_log 86400"`

Solaris: `"|/opt/sip/bin/rotatelog /opt/sip/logs/access_log 86400"`



Note You can rotate logs according to size rather than time. For example, in place of 86400 above, you can enter 5M.

- (Manual) In the SIP directives (sipd.conf) file, do the following:
 - a. Locate the line shown for GUI above and remove the comment marker at the beginning of the line.
 - b. Locate the corresponding default line that does not mention rotatelog and add a comment marker so that just one such line is left active (see [Tip](#) below).
 - c. Remove the comment marker in the CustomLog logs/access_log common line.

Step 4 Periodically remove old log files from the local hard disk. (Piped logs, as described above, do not automatically remove themselves.) This prevents logs from growing until they use up the entire hard disk and cause the proxy server to stop functioning properly.

If you use piped logs, simply move inactive logs off the hard disk without renaming, sleeping, or gracefully restarting the server as follows:

```
# cd <server_root>/logs
# mv access_log.<all except the active one> <some remote location>
# mv error_log.<all except the active one> <some remote location>
```

**Tip**

The following manual-configuration examples show the result of deleting and adding comment markers so as to leave just one active log-instruction line:

```
Correct:      #ErrorLog logs/error_log
                ErrorLog "|/opt/sip/bin/rotatelogs /opt/sip/logs/error_log 86400"

Correct:      ErrorLog logs/error_log
                #ErrorLog "|/opt/sip/bin/rotatelogs /opt/sip/logs/error_log 86400"

Incorrect:    ErrorLog logs/error_log
                ErrorLog "|/opt/sip/bin/rotatelogs /opt/sip/logs/error_log 86400"

Incorrect:    #ErrorLog logs/error_log
                #ErrorLog "|/opt/sip/bin/rotatelogs /opt/sip/logs/error_log 86400"
```

How to Back Up and Restore Cisco SPS

It is important that you back up Cisco SPS data on a regular basis so that you can recover quickly from catastrophic failures on the part of one or more servers.

The following sections describe procedures for backing up and restoring data:

- [Backing Up Data, page 3-8](#)
- [Restoring Backed-Up Data, page 3-9](#)

Backing Up Data

Prerequisites

- Make available a separate data-storage system on which to store backed-up data.
- Determine and adhere to a regular backup schedule.

Detailed Steps

-
- Step 1** If MySQL is run for the provisioning system or subscriber features or both, save all the data to a flat file using the following command on the system where MySQL is run:
- ```
mysqldump -u guest -p --databases sip > <outside_directory/file>
Enter password: <default password is "nobody">
```
- Step 2** Export any registries to a computer-separated value (csv) file as described in the [“How to Import and Export Bulk Data” section on page 2-9](#).
- Step 3** Export any static routes as described in the [“How to Import and Export Bulk Data” section on page 2-9](#).
- Step 4** Copy the license (license.conf), persistent TCP (persistent\_tcp.conf), and SIP directives (sipd.conf) files and store the copies in an alternate location.



**Note** If you use the GUI-based provisioning-system, do not back up the sipd.conf file. It regenerates from the information stored in MySQL.

**Linux:** # cp /usr/local/sip/conf/license.conf <outside\_directory/file>  
# cp /usr/local/sip/conf/persistent\_tcp.conf <outside\_directory/file>  
# cp /usr/local/sip/conf/sipd.conf <outside\_directory/file>

**Solaris:** # cp /opt/sip/conf/license.conf <outside\_directory/file>  
# cp /opt/sip/conf/persistent\_tcp.conf <outside\_directory/file>  
# cp /opt/sip/conf/sipd.conf <outside\_directory/file>

## Restoring Backed-Up Data

### Prerequisites

- If Cisco SPS was installed on your system when the system was delivered to you, before restoring Cisco SPS, uninstall Cisco SPS so that the system is in a known state and then reinstall it using the SPS setup (sps\_setup) script. For details on uninstalling and installing Cisco SPS, refer to the *Cisco SIP Proxy Server Installation Guide*.
- During reinstallation, when the license-key prompt appears, refer to the saved license (license.conf) file if the license key from initial installation is not readily available.

### Detailed Steps

**Step 1** Delete the existing sip database.

```
mysql -u guest -p
Enter password: <default password is "nobody">
at mysql> prompt type:
 drop database sip;
quit;
```

**Step 2** Restore the previously saved sip database.

```
mysql -u guest -p < <mysql_backup_file>
Enter password: <default password is "nobody">
```

**Step 3** (GUI) Delete any old provisioning-server connection data:

```
mysql -u guest -p
Enter password: <default password is "nobody">
at mysql> prompt type:
 use sip;
 delete from DBSubscriberTable;
quit;
```

**Step 4** Import any saved registries to shared memory from the backup file (csv format), as described in the [“How to Import and Export Bulk Data”](#) section on page 2-9.




---

**Note** If you have a proxy-server farm, perform this operation on the first farm member only. If an active member already exists with a current registry database, skip this step, because members update each other automatically.

---

**Step 5** Import any saved static routes as described above.

**Step 6** Restore the following files:

- license (license.conf)
- persistent TCP (persistent\_tcp.conf)
- SIP directives (sipd.conf) files

**Linux:** # cp <license.conf\_backup\_file> /usr/local/sip/conf/license.conf  
 # cp <persistent\_tcp.conf\_backup\_file> /usr/local/sip/conf/persistent\_tcp.conf  
 # cp <sipd.conf\_backup\_file> /usr/local/sip/conf/sipd.conf

**Solaris:** # cp <license.conf\_backup\_file> /opt/sip/conf/license.conf  
 # cp <persistent\_tcp.conf\_backup\_file> /opt/sip/conf/persistent\_tcp.conf  
 # cp <sipd.conf\_backup\_file> /opt/sip/conf/sipd.conf




---

**Note** If you used the SPS setup (sps\_setup) script to install Cisco SPS, do not restore the license (license.conf) file. It is generated from the information entered during setup.

---




---

**Note** If you use the provisioning-system GUI, do not restore the SIP directives (sipd.conf) file. It regenerates from the information stored in MySQL at the next start, restart, or graceful restart.

---

**Step 7** Restart Cisco SPS with the new SIP directives (sipd.conf) file.

**Linux:** # /usr/local/sip/bin/sip graceful

**Solaris:** # /opt/sip/bin/sip graceful

---

## How to Restore a MySQL Database

If you enable MySQL replication when configuring your system—that is, if you use two synchronized MySQL databases—updates to either MySQL server are replicated to the other. The SPS setup (sps\_setup) script configures each replicated MySQL server as both master and slave to each other. Replication then works as follows:

1. The master MySQL logs all changes.
2. The slave MySQL reads from these logs and keeps track of where it has read from last.

The location of debug information is shown in [Table 3-3](#).

**Table 3-3** Debug Location

| Debug Information                                            | Location                                                                                   |
|--------------------------------------------------------------|--------------------------------------------------------------------------------------------|
| For MySQL replication (when the MySQL debug flag is enabled) | Main error log (<server_root>/logs/error_log) file                                         |
| For base MySQL replication                                   | <b>Linux:</b> /var/lib/mysql/<host>.err<br><b>Solaris:</b> /opt/mysql/data/<host>.err file |

For more information about MySQL replication, refer to the MySQL website at [www.mysql.com](http://www.mysql.com).

**Note**

Do not attempt to write to the MySQL server simultaneously using both the GUI-based provisioning system and the `sysadmin_mysql_users` script. Changes made via one are not seen by the other and there is a potential to both make conflicting changes to the same user, or add the same user, in which case the first change is overwritten by the second. The same would be the case if you used two GUIs.

**Tip**

Loss of connection during an update usually leaves databases out of sync. When the connection is back up, the databases do not automatically synchronize. To cause them to synchronize, stop and restart the MySQL that did not update. You can determine whether your databases are out of sync by running the `mysql_sync_check.sh` script. You can also configure SNMP to run the script every evening.

**Detailed Steps**

- Step 1** To check for differences between the two MySQL databases, run the MySQL synchronicity check (<server\_root>/bin/mysql\_sync\_check.sh) script (run as cron if needed).

**Note**

For 20,000 users, this script takes about 10 seconds to run, during which it does not lock the databases from writes. A write during this time might fail if the slave database is not yet updated.

**Solaris:** # /opt/sip/bin/mysql\_sync\_check.sh

**Linux:** # /usr/local/sip/bin/mysql\_sync\_check.sh

- Step 2** To restore a corrupted MySQL database (corrupted, for example, by a hard-drive crash), do one of the following:

- (Recommended) If you can tolerate a stop to call processing and can therefore stop your remaining MySQL server, follow the *Cisco SIP Proxy Server Installation Guide* instructions for upgrading from a one-member to a two-member farm.

**Note**

Upgrading involves bringing down the remaining MySQL server. Call processing stops during this upgrade time.

- If you cannot tolerate a stop to call processing and must keep your remaining MySQL server up and running, use the following procedure:

**Note**

The following procedure is long and complex. If possible, follow the recommended procedure above rather than this one.

1. Back up your database (see the “[How to Back Up and Restore Cisco SPS](#)” section on page 3-8).

**On the Existing MySQL Server...**

2. On the existing MySQL server, log in as root:

**Linux:** `/usr/bin/mysql -uroot -p<root_password> sip`

**Solaris:** `/opt/mysql/bin/mysql -uroot -p<root_password> sip`

3. Stop the slave process. Ignore any error messages.

```
mysql> slave stop;
```

4. Lock the table from writes:

```
mysql> flush tables with read lock;
```

5. In a new window, change directories:

**Linux:** `cd /var/lib/mysql`

**Solaris:** `cd /opt/mysql/data`

6. Tar the MySQL data:

```
tar -cf <tmp_dir>/sip.tar sip
```

7. In the original window, enter the following **show** command:

```
mysql> show master status;
```

8. Make a note of the file and position information that displays and keep it in a safe place.

9. Unlock the table from writes:

```
mysql> unlock tables;
```

**On the New MySQL Server...**

10. On the new MySQL server, uninstall Cisco SPS (if it is already installed) and then reinstall it (refer to the *Cisco SIP Proxy Server Installation Guide*).

```
rpm -i <CSPS.rpm>
```

or

```
pkg -d <CSPS.pkg>
```

11. Install whichever member is corrupted:

```
<server_root>/bin/sps_setup
```

12. Stop MySQL:

```
/etc/init.d/mysql stop
```

13. Open the my.cnf file with a text editor and remove all “master\*” lines.

```
vi /etc/my.cnf
```

14. Change directories:

**Linux:** `cd /var/lib/mysql`

**Solaris:** `cd /opt/mysql/data`

15. Delete the current database:

```
rm -rf sip
```

16. If the database that you stored from the existing MySQL in the steps above is not accessible from this machine, copy it to a local temporary directory.

17. Restore the database with that from the existing MySQL server:

```
tar -xf <tmp_dir>/sip.tar
```

18. Start MySQL:

```
/etc/init.d/mysql start
```

19. Log in to the MySQL server as root:

**Linux:** `/usr/bin/mysql -uroot -p<root_password> sip`

**Solaris:** `/opt/mysql/bin/mysql -uroot -p<root_password> sip`

20. Stop the slave process:

```
mysql> slave stop;
```

21. Update the master information using the information that you wrote down earlier:

```
mysql> change master to master_host='<EXISTING_MYSQL_host>';
mysql> change master to master_user='guest';
mysql> change master to master_password='nobody';
mysql> change master to master_port=3306;
mysql> change master to master_log_file='vvs-finland-bin.002';
mysql> change master to master_log_pos=1428;
mysql> show master status;
```

22. Make a note of the file and position information and keep it in a safe place.

23. Start the slave process:

```
mysql> slave start;
```

#### On the Existing MySQL Server...

24. Log in to the existing MySQL server as root:

**Linux:** `/usr/bin/mysql -uroot -p<root_password> sip`

**Solaris:** `/opt/mysql/bin/mysql -uroot -p<root_password> sip`

25. Reverse master and slave, using the information that you wrote down above:

```
mysql> change master to master_log_file='vvs-iceland-bin.002';
mysql> change master to master_log_pos=73;
```

26. If you are using a different host, enter the following:

```
mysql> change master to master_host='<NEW_MYSQL_host>';
```

27. Start the new slave process:

```
mysql> slave start;
```

**Note**

Writes are blocked only during the appropriate times for the existing MySQL server. It is assumed that no writes occur to the new MySQL server until the master information is updated on the new MySQL server.

## How to Replace, Upgrade, or Delete a Cisco SPS License

Cisco SPS licenses are of two types: evaluation and permanent. You can replace one evaluation license with another, upgrade from an evaluation license to a permanent license, or delete a license.

Your license is delivered to you in the form of a license key—that is, a sequence of text characters that the Cisco SPS must read and validate at startup before it can run.

You can access your licenses either using the GUI-based provisioning system or manually. Both methods are presented below.

**Tip**

- To resize a column, place the cursor on the vertical line dividing column headers and drag it to a desired position. To rearrange column order, place the cursor on a header and drag it to a desired position.
- To display only specific licenses, use the search tool (field, operator, search string) at page top.
- To display all licenses, use the search tool with the search string set to \*.
- To display licenses in a particular order, use the column-heading sort arrows.

## GUI Method

### Detailed Steps

**Step 1** Access the Cisco SPS license window as follows:

- a. Go to the following (default) directory or your Windows desktop:

**Linux:** /usr/local/sip/gui/

**Solaris:** /opt/sip/gui/

- b. Double-click the license GUI.
- c. Enter your password (default is cspuser) and then do either of the following:

- From the pserver, click **OK**.
- From a server other than the pserver, click **more>>**, enter the pserver host name and port number, and click **OK**.

The license window appears. The pserver host name and port number automatically reappear at login.

**Step 2** To replace a license, do the following:

- a. Copy the new license key, in preparation for pasting.
- b. Locate the license (see the tips above) and click to select it.

- c. Click **Edit**.
- d. Double-click and delete the old license key.
- e. Paste in the new license key.
- f. Edit other fields as needed for the upgraded license.
- g. Click **Submit**.

**Step 3** To delete a license, do the following:

- a. Locate the license (see the tips above) and click to select it.
  - b. Click **Delete > Yes**.
- 

## Manual Method

### Detailed Steps

- 
- Step 1** Copy the new license key, in preparation for pasting.
- Step 2** Open the license (license.conf) file:
- Linux:** `/usr/local/sip/conf/license.conf`
- Solaris:** `/opt/sip/conf/license.conf`
- Step 3** Comment out the existing “LicenseKey” line and leave it in the file for backup.
- Step 4** Add a new LicenseKey line with the new key value.
- Step 5** Save and close the file.
- 

## Troubleshooting Tips

- Do not include quotation marks around the license key.
- Be cautious if you cut and paste the license key from one operating system to another. You might introduce an incorrect end-of-line character sequence that causes the system not to recognize the key.
- License-validation messages that display (Linux) on screen or (Solaris) in the `/var/log/messages` file are stored in the error log (`error_log`) file for your later reference.

