



Manual Operation and Maintenance

This appendix contains information about the following:

- [How to Manage Cisco SPS Licenses, page C-1](#)
- [How to Start and Stop Cisco SPS, page C-3](#)
- [How to Change the MySQL Password, page C-10](#)

You can perform all of these operations using the Cisco SIP proxy server (Cisco SPS) GUI-based provisioning system and, unless your situation is highly unusual, you should do so.

How to Manage Cisco SPS Licenses

Cisco SPS licenses are of two types: evaluation and permanent. An evaluation license has an expiration date. A permanent license has no expiration date.

Your license is delivered to you as a license key. A license key is a sequence of text characters that Cisco SPS must read and validate at startup before it can run.



Note

To add a license, refer to the *Cisco SIP Proxy Server Installation Guide*.

Detailed Steps

Step 1 Access the Cisco SPS licensing window by one of the following methods:

- (Recommended) Open the Cisco SPS licensing window.
- Do the following:
 - a. Go to the script directory:
Linux: `usr/local/sip/bin`
Solaris: `opt/sip/bin`
 - b. (Non-Windows 2000): Run the Cisco SPS license (`sysadmin_csp_license`) script.

- c. (Windows 2000) Copy the `licensing.jar` from the Linux or Solaris system to the following Windows system directory: `C > Program Files > Cisco SPS`. From a dos/cmd/ command-line window, double-click the text-file icon with the batch-file extension (`.bat`) that contains the following text line (or enter the line):

```
# java -jar c:/"Program Files"/CSPS/licensing.jar
```

Step 2 Enter the password (default is `cspuser`) and do one of the following:

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

The licensing window appears. The pserver host-name and port-number entries are automatically saved and reappear at each login.

Step 3 Reformat the licensing window as needed:

- To hide a column: Click **Column** on top of the window and follow instructions.
- To change a column width: Place the cursor on a vertical line dividing column headers and drag the cursor.
- To change column order: Place the cursor on a header and drag it to a new position.

Step 4 To display a license, do the following:

- a. Click **Operations > Search**.
- b. Click the down arrows to select search query and operators.
- c. Enter a search string. Do not include quotation marks.
- d. Click **OK**.

Step 5 To upgrade a license, do the following:

- a. Copy the new license key, in preparation for pasting in a subsequent step. Do not include quotation marks around the license key.
- b. Select the license that you wish to upgrade.
- c. Click **Edit**.
- d. Paste the new license key over the old license key.
- e. Edit license information and settings for the new license.
- f. Click **Submit > OK**.

Step 6 To delete a license, do the following:

- a. Select the license.
- b. Click **Delete**.

Step 7 When done, click **Reload All**.

**Tip**

- Troubleshoot any problems by checking for license-validation debug messages in the error-log (error_log) file.
- Be cautious if you cut and paste a license key from one system to another. You might introduce an incorrect end-of-line character sequence that prevents the system from recognizing the key.

How to Start and Stop Cisco SPS

This section contains information about the following:

- [Starting Cisco SPS, page C-3](#)
- [Stopping Cisco SPS, page C-4](#)
- [Restarting Cisco SPS, page C-5](#)
- [Gracefully Restarting Cisco SPS, page C-5](#)
- [Output Examples, page C-6](#)

You can start, stop, restart, and gracefully restart Cisco SPS by means of the sip script. This script is created when you install Cisco SPS using the SPS setup (sps_setup) script.

**Caution**

Scripts to start and stop each individual component exist in the same directory. Do not use the component scripts individually. Rather, use the sip script to invoke component scripts in the appropriate order and with the necessary precondition checking.

**Note**

- If errors occur when Cisco SPS starts, stops, restarts, or gracefully restarts, error messages display (Linux) in /var/log/messages or (Solaris) onscreen. For details about the log files for each process, see [Chapter 3, “Operating and Maintaining Cisco SPS.”](#)
- The sip script provides the functions in the sipdctl script in Cisco SPS 1.x.

Before your start, make a note of the directory that contains the sip script. When a step instructs you to access the sip script, this is the directory to which you go to do so. The default location is as follows:

Linux: /usr/local/sip/bin

Solaris: /opt/sip/bin

Starting Cisco SPS

Detailed Steps

**Tip**

For sample screen and log outputs for these steps, see the [“Output Examples” section on page C-6](#).

Step 1 Run the sip script with the start argument:

Linux: # /usr/local/sip/bin sip start

Solaris: # /opt/sip/bin sip start

Step 2 Verify that Cisco SPS is running properly by viewing all Cisco SPS processes:

ps -ef | grep sps

You should have the number of processes shown in [Table C-1](#).

Table C-1 Cisco SPS Processes

Process Type	Linux	Solaris
Provisioning server (pserver)	1	1
License manager (licenseMgr)	3	1
SIP provisioning agent (spa)	3	1
sipd	7 (default)	7 (default)

Stopping Cisco SPS

Detailed Steps



Tip

For sample screen and log outputs for these steps, see the [“Output Examples” section on page C-6](#).

Step 1 Run the sip script with the stop argument:

Linux: # /usr/local/sip/bin sip stop

Solaris: # /opt/sip/bin sip stop

All Cisco SPS processes stop and the server can no longer process calls.

Step 2 Verify that Cisco SPS processes are stopped:

Linux: # /usr/local/sip/bin ps -ef | grep sps

Solaris: # /opt/sip/bin ps -ef | grep sps

Step 3 If any of the following processes still are running, stop them manually by using the UNIX command **kill**:

- SIP proxy server (sipd)
- SIP provisioning agent (spa)
- License manager (licenseMgr)
- Provisioning server (pserver)

Restarting Cisco SPS

Detailed Steps

**Tip**

For sample screen and log outputs for these steps, see the “[Output Examples](#)” section on page C-6.

Step 1 Run the sip script with the restart argument:

Linux: # /usr/local/sip/bin sip restart

Solaris: # /opt/sip/bin sip restart

Step 2 Verify that Cisco SPS is running properly by viewing all Cisco SPS processes:

```
# ps -ef | grep sps
```

You should have the number of processes shown in [Table C-1](#).

**Note**

Process IDs that display upon system startup differ from those in effect beforehand. During the time between stop and start, the server cannot process calls.

Gracefully Restarting Cisco SPS

Graceful restart provides a mechanism to prompt spa to write a new sipd.conf file. During a graceful restart, the sipd daemon (parent process) remains alive, rereads the configuration file (sipd.conf), tears down child sipd processes as they become idle, and spawns new child processes with the new configuration. Call processing is not interrupted as a result.

If a server configuration changes, perform a graceful restart to activate the new configuration without dropping calls.

If a Cisco SPS TCP I/O process becomes unresponsive, the parent sipd performs its own graceful restart (up to five times) to activate the process.

Detailed Steps

**Tip**

For sample screen and log outputs for these steps, see the “[Output Examples](#)” section on page C-6.

Step 1 Run the sip script with the graceful argument:

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

Solaris: # /opt/sip/bin sip graceful

Step 2 Verify that Cisco SPS is running properly by viewing all Cisco SPS processes (the number of processes should be as in [Table C-1](#)):

```
# ps -ef | grep sps
```

Step 3 If the TCP I/O process fails to activate, wait 1 minute and perform a graceful restart manually.

Output Examples

This section contains the following output examples:

- [Screen and Log Output: Starting Cisco SPS, page C-6](#)
- [Screen and Log Output: Stopping Cisco SPS, page C-7](#)
- [Screen and Log Output: Restarting Cisco SPS, page C-7](#)
- [Screen and Log Output: Gracefully Restarting Cisco SPS, page C-9](#)

Screen and Log Output: Starting Cisco SPS

Start Screen Output

Upon startup, the screen displays output similar to the following:

```
Starting pserver:           [ OK ]
Starting license manager:  [ OK ]
Starting spa:              [ OK ]
Starting sipd:             [ OK ]
```

Start-Verification Screen Output

Upon system-start verification, the screen displays output similar to the following. In this example, the first sipd process, with parent process ID 1, is the parent sipd. The other sipd processes are the TCP I/O process and five child processes. Sip_Services is an additional process required to maintain synchronization among local and remote farm members.

```
csps      4040      1  0 17:38 ?          00:00:00 /usr/local/sip/bin/pserver -c /u
csps      4054      1  0 17:38 ?          00:00:00 /usr/local/sip/bin/licenseMgr /u
csps      4064  4054  0 17:38 ?          00:00:00 /usr/local/sip/bin/licenseMgr /u
csps      4065  4064  0 17:38 ?          00:00:00 /usr/local/sip/bin/licenseMgr /u
csps      4068      1  0 17:38 ?          00:00:00 /usr/local/sip/bin/spa /usr/loca
csps      4074      1  0 17:38 ?          00:00:00 /usr/local/sip/bin/Sip_Services
csps      4080  4068  0 17:38 ?          00:00:00 /usr/local/sip/bin/spa /usr/loca
csps      4081  4080  0 17:38 ?          00:00:00 /usr/local/sip/bin/spa /usr/loca
csps      4092      1  0 17:38 pts/1        00:00:00 /usr/local/sip/bin/sipd
csps      4094  4092  0 17:38 pts/1        00:00:00 /usr/local/sip/bin/sipd
csps      4096  4092  0 17:38 pts/1        00:00:00 /usr/local/sip/bin/sipd
csps      4097  4092  0 17:38 pts/1        00:00:00 /usr/local/sip/bin/sipd
csps      4100  4092  0 17:38 pts/1        00:00:00 /usr/local/sip/bin/sipd
csps      4101  4092  0 17:38 pts/1        00:00:00 /usr/local/sip/bin/sipd
csps      4102  4092  0 17:38 pts/1        00:00:00 /usr/local/sip/bin/sipd
root      4107  1387  0 17:39 pts/1        00:00:00 grep sps
```

Screen and Log Output: Stopping Cisco SPS

Stop Screen Output

Upon system stop, the screen displays output similar to the following:

```
Stopping sipd:           [ OK ]
Stopping spa:           [ OK ]
Stopping license manager: [ OK ]
Stopping pserver:      [ OK ]
```

Stop Log Output

Upon system stop, the log (Linux /var/log/messages file or Solaris screen) displays output similar to the following:

```
sipdctl: Waiting for process to stop.
sipdctl: .
sipdctl: /usr/local/sip/bin/sipdctl stop: sipd stopped
sipdctl: Waiting for process to stop.
sipdctl: .
sipdctl: /usr/local/sip/bin/sipdctl stop: Sip_Services stopped
sip: Stopping sipd: succeeded
spactl: Waiting for process to stop.
spactl: .
spactl: /usr/local/sip/bin/spactl stop: spa stopped
sip: Stopping spa: succeeded
lmctl: Waiting for process to stop.
lmctl: .
lmctl: /usr/local/sip/bin/lmctl stop: licenseMgr stopped
sip: Stopping license manager: succeeded
pserverctl: Waiting for process to stop.
pserverctl: .
pserverctl: /usr/local/sip/bin/pserverctl stop: pserver stopped
sip: Stopping pserver: succeeded
```

Stop-Verification Screen Output

Upon system-stop verification, the screen displays output similar to the following:

```
csps 16421 15876 0 09:47 pts/0 00:00:00 grep sps
```

Screen and Log Output: Restarting Cisco SPS

Restart Screen Output

Upon system restart, the screen displays output similar to the following:

```
Stopping sipd:           [ OK ]
Stopping spa:           [ OK ]
Stopping license manager: [ OK ]
Stopping pserver:      [ OK ]
Starting pserver:      [ OK ]
Starting license manager: [ OK ]
Starting spa:          [ OK ]
Starting sipd:         [ OK ]
```

Restart Log Output

Upon system restart, the log (Linux /var/log/messages file or Solaris screen) displays output similar to the following:

```
sipdctl: Waiting for process to stop.
sipdctl: .
```

```

sipdctl: /usr/local/sip/bin/sipdctl stop: sipd stopped
sipdctl: Waiting for process to stop.
sipdctl: .
sipdctl: /usr/local/sip/bin/sipdctl stop: Sip_Services stopped
sip: Stopping sipd: succeeded
spactl: Waiting for process to stop.
spactl: .
spactl: /usr/local/sip/bin/spactl stop: spa stopped
sip: Stopping spa: succeeded
lmctl: Waiting for process to stop.
lmctl: .
lmctl: /usr/local/sip/bin/lmctl stop: licenseMgr stopped
sip: Stopping license manager: succeeded
pserverctl: Waiting for process to stop.
pserverctl: .
pserverctl: /usr/local/sip/bin/pserverctl stop: pserver stopped
sip: Stopping pserver: succeeded
pserverctl: /usr/local/sip/bin/pserverctl start: pserver started
sip: Starting pserver: succeeded
lmctl: /usr/local/sip/bin/lmctl start: licenseMgr started
sip: Starting license manager: succeeded
spactl: /usr/local/sip/bin/spactl start: spa started
spactl: /usr/local/sip/bin/spactl start: Waiting for sipd.conf from spa..
spactl: .
spactl: /usr/local/sip/bin/spactl start: sipd.conf written
sip: Starting spa: succeeded
sipdctl: Version of CSPS      : 2.1.x.x
sipdctl: Version in Config file : 2.1.x.x
sipdctl: Software release version of CSPS validated successfully with your license
sipdctl: License validated successfully
sipdctl: This is Permanent license, with Infrastructure functionality
sipdctl: /usr/local/sip/bin/sipdctl start: sipd started
sip: Starting sipd: succeeded

```

Restart-Verification Screen Output

Upon system-restart verification, the screen displays output similar to the following:

```

[/usr/local/sip/bin]# ps -ef | grep sps
csps      4216      1  0 17:58 ?        00:00:00 /usr/local/sip/bin/pserver -c /u
csps      4225      1  0 17:58 ?        00:00:00 /usr/local/sip/bin/licenseMgr /u
csps      4232  4225  0 17:58 ?        00:00:00 /usr/local/sip/bin/licenseMgr /u
csps      4233  4232  0 17:58 ?        00:00:00 /usr/local/sip/bin/licenseMgr /u
csps      4241      1  0 17:58 ?        00:00:00 /usr/local/sip/bin/spa /usr/loca
csps      4245      1  0 17:58 ?        00:00:00 /usr/local/sip/bin/Sip_Services
csps      4252  4241  0 17:58 ?        00:00:00 /usr/local/sip/bin/spa /usr/loca
csps      4253  4252  0 17:58 ?        00:00:00 /usr/local/sip/bin/spa /usr/loca
csps      4264      1  0 17:58 pts/1    00:00:00 /usr/local/sip/bin/sipd
csps      4266  4264  0 17:58 pts/1    00:00:00 /usr/local/sip/bin/sipd
csps      4268  4264  0 17:58 pts/1    00:00:00 /usr/local/sip/bin/sipd
csps      4269  4264  0 17:58 pts/1    00:00:00 /usr/local/sip/bin/sipd
csps      4270  4264  0 17:58 pts/1    00:00:00 /usr/local/sip/bin/sipd
csps      4271  4264  0 17:58 pts/1    00:00:00 /usr/local/sip/bin/sipd
csps      4276  4264  0 17:58 pts/1    00:00:00 /usr/local/sip/bin/sipd
root      4279  1387  0 18:00 pts/1    00:00:00 grep sps

```


Screen and Log Output: Gracefully Restarting Cisco SPS

Graceful-Restart Screen Output

Upon graceful restart, the screen displays output similar to the following:

```
Gracefully restarting pserver:           [ OK ]
Gracefully restarting license manager:   [ OK ]
Gracefully restarting spa:               [ OK ]
Gracefully restarting sipd:              [ OK ]
```

Graceful-Restart Log Output

Upon graceful restart, the log (Linux /var/log/messages file or Solaris screen) displays output similar to the following:

```
pserverctl: /usr/local/sip/bin/pserverctl graceful: pserver (pid 3749 3769 3770 3775)
already running
sip: Gracefully restarting pserver: succeeded
lmctl: /usr/local/sip/bin/lmctl graceful: licenseMgr (pid 3764 3766 3767) already running
sip: Gracefully restarting license manager: succeeded
spactl: Waiting for process to stop.
spactl: .
spactl: /usr/local/sip/bin/spactl stop: spa stopped
spactl: Wait 3 seconds before restarting the application...
spactl: /usr/local/sip/bin/spactl start: spa started
spactl: /usr/local/sip/bin/spactl start: Waiting for sipd.conf from spa..
spactl: .
spactl: /usr/local/sip/bin/spactl start: sipd.conf written
sip: Gracefully restarting spa: succeeded
sipdctl: /usr/local/sip/bin/sipdctl graceful: sipd gracefully restarted
sip: Gracefully restarting sipd: succeeded
```

Graceful-Restart-Verification Screen Output

Upon graceful-restart verification, the screen displays output similar to the following. In this example, the original pserver and licenseMgr processes are not affected. The spa processes are restarted to force the writing of a new SIP directives (sipd.conf) file. The parent sipd process, the original Sip_Services, and the TCP I/O sipd process remain the same as for the previous start of the server. All other sipd child processes have been restarted and have new process IDs.

```
csps      4216      1  0 17:58 ?          00:00:00 /usr/local/sip/bin/pserver -c /u
csps      4225      1  0 17:58 ?          00:00:00 /usr/local/sip/bin/licenseMgr /u
csps      4232  4225  0 17:58 ?          00:00:00 /usr/local/sip/bin/licenseMgr /u
csps      4233  4232  0 17:58 ?          00:00:00 /usr/local/sip/bin/licenseMgr /u
csps      4235  4216  0 17:58 ?          00:00:00 /usr/local/sip/bin/pserver -c /u
csps      4236  4235  0 17:58 ?          00:00:00 /usr/local/sip/bin/pserver -c /u
csps      4240  4235  0 17:58 ?          00:00:00 /usr/local/sip/bin/pserver -c /u
csps      4264      1  0 17:58 pts/1      00:00:00 /usr/local/sip/bin/sipd
csps      4266  4264  0 17:58 pts/1      00:00:00 /usr/local/sip/bin/sipd
csps      4334      1  0 18:02 ?          00:00:00 /usr/local/sip/bin/spa /usr/loca
csps      4337      1  0 18:02 ?          00:00:00 /usr/local/sip/bin/Sip_Services
csps      4344  4334  0 18:02 ?          00:00:00 /usr/local/sip/bin/spa /usr/loca
csps      4345  4344  0 18:02 ?          00:00:00 /usr/local/sip/bin/spa /usr/loca
csps      4357  4264  0 18:02 pts/1      00:00:00 /usr/local/sip/bin/sipd
csps      4358  4264  0 18:02 pts/1      00:00:00 /usr/local/sip/bin/sipd
csps      4359  4264  0 18:02 pts/1      00:00:00 /usr/local/sip/bin/sipd
csps      4360  4264  0 18:02 pts/1      00:00:00 /usr/local/sip/bin/sipd
csps      4361  4264  0 18:02 pts/1      00:00:00 /usr/local/sip/bin/sipd
root      4370  1387  0 18:13 pts/1      00:00:00 grep sps
```

How to Change the MySQL Password

User and root passwords are set when you run the SPS setup (sps_setup) script. You can, however, change the root password. If a user has forgotten a password, assign a new one.

Detailed Steps

Step 1 Log in to the MySQL database.

Step 2 Enter the following 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`
