



Process Control

This chapter describes the components, configuration, and management utilities associated with the Cisco Info Center Process Control system. It includes the following sections:

- [Introduction to Process Control](#)
- [Process Control Components, page 7-3](#)
- [Creating a Process Control System Configuration, page 7-4](#)
- [Defining Processes, Services, and Hosts, page 7-6](#)
- [Example Process Agent Configuration File, page 7-12](#)
- [Process Control Agent Daemon Command Line Options, page 7-15](#)
- [Process Control Management, page 7-17.](#)

Introduction to Process Control

The Cisco Info Center Process Control system performs two primary tasks:

- execution of automations using external effects
- management of local and remote processes.



Note

When you install Cisco Info Center, you must select the Process Control component when installing components that use Process Control—such as the Info Server and Info Mediators components. For more detailed information, refer to the Cisco Info Center Installation and Configuration Guide, 3.6.

The information in this chapter is for advanced users who might want to modify the process control agents as installed and configured during the Cisco Info Center installation. If you are not familiar with the files and scripts that are used to implement process control, it is advisable to maintain the configuration using the Cisco Info Center configuration utility, **nco_config**.

Execution of External Effects in Automations

Process Control is responsible for the execution of external effects specified in automations. An automation does not execute programs by itself. It sends a request to a local Process Control agent, which forwards the request to the Process Control agent running on the specified host. The remote Process Control agent then executes the requested program.



Note

Process Control agents on UNIX machines can only connect to Process Control agents on other UNIX machines. External effects cannot pass between these different environments.

Configuring and Managing Cisco Info Center Components

Process Control allows you to configure remote processes to simplify the configuration and management of Cisco Info Center components such as Cisco Info Servers, Cisco Info Mediators, and gateways. It consists of the following elements:

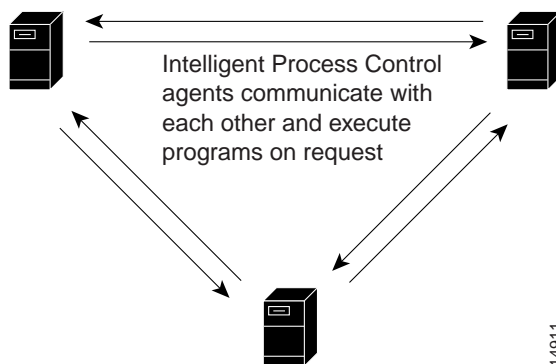
- Process Control agents, which are programs installed on each host with the responsibility of managing processes
- a set of command line utilities to provide an interface to process management.

The Process Control agents cooperate automatically and understand their own configuration. Process Control agents start remote processes and are capable of keeping those processes running. Processes can be defined as dependent on the starting of a previous process or have timed threshold dependencies. If a managed host is restarted, the Process Control agent restarts local components automatically.

Process Agents

Any Cisco Info Center participating UNIX host must be configured with the Process Control agent daemon. When you configure an Info Server or other Info Center components using the `nco_config` configuration utility, the Info Server and other components are automatically configured to use Process Control and the host is connected into the Process Control system. [Figure 7-1](#) shows a basic Cisco Info Center configuration consisting of three hosts running Process Control agents.

Figure 7-1 Basic Cisco Info Center Process Control Configuration



Joining the Network

A process agent joins the network by connecting to an existing member. If it tries to connect and fails, it retries after the number of seconds specified in the **MaxConnect** property in its properties file (`agent-name_PA.props`). If a process agent loses one of its connections in the network, it waits for the number of seconds specified in the **MinConnect** property in this file for a reconnection from another process agent. If this does not arrive, it tries to reconnect itself to the node to which it was last connected at the interval specified in the **MaxConnect** property.

When a process agent receives a request to join the existing network, it allows a proportion of the **MinConnect** period for the requesting agent to confirm it has joined. This proportion is specified as a percentage between 20% and 80% in the **AcceptRatio** property.

Removing a Process Agent From the Network

To remove a process agent from the network, stop the service from the Control Panel. Before it leaves the network, the agent notifies the other members and waits for acknowledgement for the time specified in the **MaxRingSpeed** property. If it does not receive acknowledgement, it leaves the network anyway. In this case, the network is not broken even if that was the intention in removing the agent.

The appropriate values for the **MinConnect**, **MaxRingSpeed**, and **AcceptRatio** properties are a function of network speed. They should be larger for slower networks to allow less stringent recovery criteria.

Process Control Components

Process Control consists of a Process Control agent daemon, an associated configuration file, and utilities to help you manage the Process Control agent and the processes and services configured in Process Control.

Process Control Agents

Process Control agents are programs installed on each host to manage Cisco Info Center processes. Any participating host must have a Process Control agent daemon and associated configuration file installed.

There can be any number of Process Control agents on any number of hosts. Process Control agents can manage any number of processes.

Processes

Processes are programs executed by a Process Control agent on the same machine. Processes must be defined within a service, described next.

A **PA Aware** process is part of the Process Control configuration and is aware of Process Control. All the features of Process Control, such as process dependencies, can be utilized. The Cisco Info Server, Cisco Info Mediators, and gateways are **PA Aware**. A process that is not **PA Aware** can be managed by Process Control, but cannot use all the features of Process Control a **PA Aware** process can use. The Desktop is not **PA Aware**. For information about how to set the process type to **PA Aware**, see [Defining Processes, page 7-6](#).

Sometimes the order in which applications are started is important. You can use Process Control to configure processes to be dependent on each other. For example, a process can be configured to start only after another process has started and completed various startup tasks.

A **PA Aware** process communicates with the Process Control agent. Once the process has reached the point in its startup when it recognizes itself to be running, it sends a message to the Process Control agent. When the Process Control agent receives this message, it starts dependent processes.

Services

You can use Process Control to configure and manage local and remote processes, and to provide automatic management of those processes. Processes must be defined within services. You can group related processes in a service to make them easier to manage. Once a service is correctly configured, it can be managed by Process Control.

A service is made up of one or more processes executed by Process Control agents. A service can be configured to start up automatically when the Process Control agent starts. Alternatively, it can be started manually.

A service can either be a master or a non-master. When started automatically by Process Control, master services are started before non-master services.

Process Control Utilities

Command line utilities are provided to do the following:

- retrieve and display the status of services and processes being controlled by the Process Control agent
- start a service or process located anywhere within the configuration
- stop a service or process located anywhere within the configuration
- shut down a Process Control agent
- add a service or process entry while a Process Control agent is running.

Creating a Process Control System Configuration

This section describes how to create a new Process Control configuration.

Before You Configure Process Control

Before you start to configure Process Control:

- Ensure the Process Control component is installed. See the *Cisco Info Center Installation and Configuration Guide*, 3.6 for additional information.
- Determine which Cisco Info Center components are installed and where they are located. Ensure you have taken into account all components and any failover or backup systems. The Cisco Info Center Desktops are not managed by Process Control.

Once you have determined the complete Cisco Info Center configuration requirements, follow the instructions described in the following sections.

Creating User Groups

By default, the Process Control system uses UNIX user names and passwords to grant access to the Process Control system. Other supported authorization modes can be specified using the **-authorization** command line option, described in [Process Control Management](#), page 7-17.

To control who can log in, any user who needs access to the Process Control system must be made a member of a UNIX user group called **ncoadmin**. If this group does not exist, create it and add Process Control users. If you run NIS, NIS+, or some other global information service, this process must be performed by the administrator of that service. See the documentation provided with your operating system for details of creating user groups.

Configuring Process Control Agents

For each Process Control agent you must:

- add the server name in the Server Editor
- start the Process Control agent.

Adding Server Names in the Server Editor

Using the Server Editor, you must assign a unique server name to each Process Control agent for each host machine. For each Process Control agent:

1. Add an entry for each Process Control agent using the Server Editor window, as described in the *Cisco Info Center Installation and Configuration Guide, 3.6*.

The name must end with **_PA** to identify the server as a Process Control agent in the server editor. For example, if you are installing the Process Control agent on a host machine named **sfosys1**, the Process Control agent could be named **SFOSYS1_PA**. By default, the first Process Control agent installed in a configuration is named **NCO_PA**.

2. Distribute updated interfaces files to all host machines in the configuration. See the *Cisco Info Center Installation and Configuration Guide, 3.6* for information about how to generate interfaces files for all platforms.

This enables all Process Control agents on all host machines to connect to each other.

Starting the Process Control Agent Manually

The Process Control agent can be started manually at the command line with the command:

```
$OMNIHOME/bin/nco_pad -name <process_agent>
```

where *<process_agent>* is the name of the Process Control agent.

Starting Process Control Automatically on Reboot

When you install Cisco Info Center, you are prompted to configure automatic startup on system boot. If you choose to configure automatic startup, then the name of a startup script is added to the startup script for the host.

If you choose not to configure automatic startup, then the installation utility places a startup script called **nco** in the **/etc/init.d** directory. You can then use the **nco** script to start and stop the installed Info Server components manually.

When Cisco Info Server starts up automatically or you start it manually, process control is started up and any components that run under process control are run under the control of the process control system.

For information on the **nco** command, refer to the [“Starting a Cisco Info Server Manually”](#) section on page 1-2, and the [“Stopping a Cisco Info Server Manually”](#) section on page 1-2.

Defining Processes, Services, and Hosts

To run under Process Control, processes, services, and hosts must be added to the following configuration file:

\$OMNIHOME/etc/process_agent_name.conf

The Process Control agent reads this file when it is started to establish configuration settings. The file is made up of definitions, each of which contains various attributes and associated values, for each process, service, and host.

Edit this file directly to add or modify definitions. Maintain the configuration files on all of your hosts to ensure the host configuration information stays synchronized across all of the agents in the configuration.



Note

To prevent unauthorized users from gaining access, operating system security must be set appropriately for files such as configuration files that may contain usernames and passwords.

Defining Processes

You must define the list of processes in the configuration file. An example process definition in the configuration file is shown below:

```
nco_process 'Info Server'
{
    Command '$OMNIHOME/bin/nco_objserv -name NCOMS -pa SFOSYS1_PA' run as 0
    Host='sfosys1'
    Managed=true
    RestartMsg='The Info Server has been restarted'
    AlertMsg='The Info Server has gone down'
    RetryCount=0
    ProcessType=PaPA_AWARE
}
```

Process Definition Description

[Table 7-1](#) contains a description of the process definition information contained in the configuration file.

Table 7-1 Process Definition Description

Configuration Information	Description
nco_process 'Info Server'	<p>Defines the name of the process. This example is for a Cisco Info Server.</p> <p>The names for the processes must be unique within the complete Process Control network.</p>
Command	<p>The command string that starts the process as it would be entered on the command line. Use the full path for the command. For example, to configure a Cisco Info Server named NCOMS, enter:</p> <pre>'\$OMNIHOME/bin/nco_objserv -name NCOMS -pa SFOSYS1_PA' run as 0</pre> <p>or:</p> <pre>'\$OMNIHOME/bin/nco_objserv -name NCOMS -pa SFOSYS1_PA' run as 'root'</pre> <p>The run as option instructs the host machine to run the Cisco Info Server as the specified user. Enter either the user ID (typically 0) or the user name in quotes (typically root). When a username is entered, the process agent looks up the user ID to use.</p> <p>If the Process Control agent is not running as root, this option is ignored, and the process is run as the user who is running the Process Control agent.</p> <p>For information about the Cisco Info Server command line options, see Specifying Cisco Info Server Command Line Options, page 1-3.</p>
Host	The name of the host on which the process should be executed. Process Control automatically resolves the name of the Process Control agent when required.
Managed	Can have the value True (the process is restarted automatically if it exits) or False (the process is not restarted automatically if it exits).
RestartMsg	Contains the message to be sent to syslog if the process is restarted. For example, The Cisco Info Server has been restarted.
AlertMsg	Contains the message to be sent to syslog if the process exits. For example, The Cisco Info Server has gone down.
RetryCount	Specifies the number of restart attempts to be made if the process exits within the time specified by the process agent daemon -retrytime command line option. If set to 0 , there is no limit to the number of restart attempts. The default is 0 .
ProcessType	Can have the value PaPA_AWARE for PA aware processes and PaNOT_PA_AWARE for processes not PA aware. See Processes, page 7-3 for additional information.

Expansion Keywords

You can include expansion keywords in the **RestartMsg** and **AlertMsg** entries in the configuration file. Expansion keywords act as variables and contain information about the process that has restarted. The expansion keywords are shown in [Table 7-2](#).

Table 7-2 Expansion Keywords

Expansion Keyword	Description
<code>\${NAME}</code>	The name of the process.
<code>\${HOST}</code>	The name of the host running the process.
<code>\${EUID}</code>	The effective user ID under which the process is running.
<code>\${COMMAND}</code>	The command that defines the process.

Alert and Restart Syslog Messages

When an alert or restart message is generated by `nco_pad`, it is passed to the syslog system. Cisco Info Center has a Syslog Cisco Info Mediator that can monitor these messages and convert them into Cisco Info Server alerts. For more information about the Syslog Cisco Info Mediator, see the documentation available for each Cisco Info Mediator on the Cisco Systems Support Site.

The alert and restore messages are sent to syslog as warnings. The message is formatted as:

HOSTNAME : `<ALERT_OR_RESTART_MSG>` : `<MSG>`

where **HOSTNAME** is the name of the host which has reported the problem, `<ALERT_OR_RESTART_MSG>` describes the type of message. `<MSG>` is the text defined in the configuration file for that process or host.

Defining Services

You can define services to group together the functional elements of the system. A group of processes run together in a specified way provide a service. The processes must already be defined in the list of processes.

An example of a service definition in the configuration file is shown below.

```
nco_service 'Omnibus'
{
    ServiceType=Master
    ServiceStart=Non-Auto
    process 'Info Server' NONE
    process 'Proxy' 'Info Server'
    process 'Info Mediator' 'Proxy'
    process 'Info Mediator-1' 'Info Server'
    process 'Sleep' 5
}
```

Service Definition Description

Table 7-3 contains a description of the service definition information contained in the configuration file.

Table 7-3 Service Definition Description

Configuration Information	Description
nco_service ‘Omnibus’	Defines the name of the service (for example, Omnibus). Each name must be unique within the complete Process Control network.
ServiceType	Defines whether this service should be started before all other services and handled as the master service upon which other services depend. This can be set as either Master or Non-Master .
ServiceStart	This can be set to Auto to start the service as soon as nco_pa has started and Non-Auto if the service must be started manually with the nco_pa_start command.
process	Defines a process that must be run as part of the service.

Specifying Process Dependencies

The process attribute allows you to define the process that should be run as part of the service. You can add dependencies on each of the processes in the service. The format of the process attribute is as follows:

```
process ‘<processname>’ <dependency>
```

where <processname> is the name of the process defined in the list of processes and <dependency> can be a numeric value, a string value, or **NONE**.

The dependency type numeric allows you to specify a time dependency, in seconds, for starting the dependent process. For example, if you enter 5, the process starts five seconds after the service has started.

The dependency type string allows you to specify another **PA aware** process within the same service. See [Table 7-1 on page 7-7](#) for more information about the process types.

In the example Omnibus service, the Cisco Info Server process starts first because it has no dependencies. Five seconds after the Cisco Info Server starts, the Sleep process starts. Once the Cisco Info Server is running successfully, Proxy and Cisco Info Mediator-1 start. When the proxy server is running, the Cisco Info Mediator process starts.

The dependency type **NONE** specifies no dependency.

Defining Secure Hosts

You can specify only certain hosts can connect to process agents by adding a security definition to the configuration file between the service and routing definitions, described in the next section.

If you do not create a security definition, any process can connect from any host.

You can create a security definition with no hosts specified, as follows:

```
nco_security
{
}
```

When no hosts are specified, only processes running on the current host or any host listed in the routing definition can connect.

Processes running on hosts not listed in the routing definition can only connect if their host is listed in the security definition.

The process agent compares the IP address of the incoming connection with the IP address of each entry in the security and routing definitions and the local host. Only the main address of the host running the process agent daemon is automatically added to the security definition. You must add the loopback address (127.0.0.1) and secondary interfaces, if required.



Note

When a process connecting to the process agent is run on a host with multiple interfaces, you should add the address of the interface closest to the process agent daemon. This does not need to be the main address of that host, nor, in the case of the Cisco Info Server (**nco_objserv**) or the process agent daemon (**nco_pad**), does it need to be the address in the server editor (**nco_xigen**).

You can specify the following types of entries in the security definition:

- a host name, in which a case lookup is performed to find the corresponding IP address
- a full IP address in dotted decimal format
- an IP address in dotted decimal format can contain the following wildcards:
 - ? matches one character
 - * matches any number of characters.

The following security definition allows connections from processes on hosts alpha, 192.9.200.34, and any host on the subnet 193.42.52.0.

```
nco_security
{
  host 'alpha'
  host '192.9.200.34'
  host '193.42.52.*'
}
```

Defining Routing Hosts, Usernames, and Passwords

To specify the hosts the process agents should contact to reach each other, you must define the process agent host names.

Each host field defines the name of the host (for example, **sfosys1**) and the name of the process agent to be used in the Process Control system (for example, **SFOSYS1_PA**).

When using Process Control in secure mode, the routing entry must also have a username and password.

Routing Definition Example

An example of a routing definition in the configuration file is shown next:

```
nco_routing
{
host 'sfosys1' 'SFOSYS1_PA' 'username' 'password'
host 'sfosys2' 'SFOSYS2_PA' 'username' 'password'
}
```

If the process agent is using UNIX authentication (the default), the username must be an operating system user that is a member of the **ncoadmin** group, as described in [Creating User Groups, page 7-4](#). A process agent daemon running in secure mode must be run by the **root** user.

**Note**

If you run the process agent in secure mode, user names and passwords are required in the routing entries. If you are not running the process agent in secure mode, user names and passwords are optional in the routing entries.

Password Encryption Example

You can use the **nco_pa_crypt** utility to encrypt plain text login passwords stored in the configuration file. Passwords encrypted using **nco_pa_crypt** are decrypted by the remote Process Control agent.

To encrypt a plain text password:

Step 1 Enter the following:

```
$OMNIHOME/bin/nco_pa_crypt <password>
```

where <password> is the unencrypted form of the password. The **nco_pa_crypt** utility displays an encrypted version of the password.

Step 2 Copy the encrypted password into the appropriate routing entry.

If the username is specified without a password, the system prompts for a password. If the password is specified without a username, the name of the user entering the command is used.

**Note**

To prevent unauthorized users from gaining access, operating system security must be set appropriately for files containing usernames and passwords.

You can also specify the username and password with the **-username** and **-password** command line options. This overrides any entries in the properties file.

**Note**

Even if the password is specified on the command line, it does not appear in **ps** command output.

Example Process Agent Configuration File

The following example shows the settings for a simple process agent configuration file:

```
#NCO_PA3
#
# Process Agent Daemon Configuration File 1.1
#
#
# List of processes
#
nco_process 'Info Server'
{
Command '$OMNIHOME/bin/nco_objserv -name NCOMS -pa SFOSYS1_PA' run as 0
Host = 'sfosys1'
Managed = True
RestartMsg = 'The Info Server has been restarted'
AlertMsg = 'The Info Server has gone down'
RetryCount = 0
ProcessType = PaPA_AWARE
}

nco_process 'Proxy'
{
Command '$OMNIHOME/bin/nco_proxyserv -name SCNCOMS -server NCOMS
-debug' run as 0
Host='sfosys1'
Managed = True
RestartMsg = ` `
AlertMsg = ` `
RetryCount = 0
ProcessType = PaPA_AWARE
}

nco_process 'Info Mediator'
{
Command '$OMNIHOME/probes/nco_p_simnet -server SCNCOMS' run as 0
Host = 'sfosys1'
Managed = True
RestartMsg = ` `
AlertMsg = ` `
RetryCount = 0
ProcessType = PaNOT_PA_AWARE
}

nco_process 'Info Mediator-1'
```

```
{
Command '$OMNIHOME/probes/nco_p_simnet -server NCOMS' run as 0
Host = 'sfosys1'
Managed = True
RestartMsg = ' '
AlertMsg = ' '
RetryCount = 0
ProcessType = PaNOT_PA_AWARE
}

nco_process 'Sleep'
{
Command '/usr/bin/sleep 500' run as 60003
Host = 'sfosys1'
Managed = True
RestartMsg = 'STARTED'
AlertMsg = 'STOPPED'
RetryCount = 0
ProcessType = PaNOT_PA_AWARE
}

nco_process 'Sleep-1'
{
Command '/usr/bin/sleep 500' run as 305
Host = 'sfosys2'
Managed = True
RestartMsg = ' '
AlertMsg = ' '
RetryCount = 0
ProcessType = PaNOT_PA_AWARE
}

nco_process 'Sleep-2'
{
Command '/usr/bin/sleep 500' run as 305
Host = 'sfosys2'
Managed = True
RestartMsg = ' '
AlertMsg = ' '
RetryCount = 0
ProcessType = PaNOT_PA_AWARE
}

nco_process 'Sleep-3'
{
Command '/usr/bin/sleep 500' run as 305
```

```
Host = 'sfosys4'
Managed = True
RestartMsg = ' '
AlertMsg = ' '
RetryCount = 0
ProcessType = PaNOT_PA_AWARE
}

# List of Services

nco_service 'Omnibus'
{
ServiceType = Master
ServiceStart = Non-Auto
process 'Info Server' NONE
process 'Proxy' 'Info Server'
process 'Info Mediator' 'Proxy'
process 'Info Mediator-1' 'Info Server'
process 'Sleep' 5
}

nco_service 'Core'
{
ServiceType = Master
ServiceStart = Auto
process 'Master Object Server' NONE
}

nco_security
{
}

# ROUTING TABLE
#
nco_routing
{
host 'sfosys1' 'SFOSYS1_PA' 'ssmith' 'sspass'
host 'sfosys2' 'SFOSYS2_PA' 'ssmith' 'sspass'
}
```

Process Control Agent Daemon Command Line Options

This section describes the command line options for the Process Control agent daemon, **nco_pad**. [Table 7-4](#) describes the Process Control agent command line options.

Table 7-4 Process Control Agent Daemon Command Line Options

Option	Parameter	Description
-apicheck	N/A	If specified, Sybase API checking is enabled.
-authenticate	string	<p>Specifies the authentication mode to use to verify the credentials of a user or remote Process Control agent daemon. The options are UNIX, PAM, KERBEROS, and HPTCB.</p> <p>The default authentication mode is UNIX, which means the Posix getpwnam or getspnam function is used to verify user credentials on UNIX platforms. Depending on system setup, passwords are verified using the /etc/password file, /etc/shadow shadow password file, NIS, or NIS+.</p> <p>If PAM is specified as the authentication mode, Pluggable Authentication Modules are used to verify user credentials. The service name used by the gateway when the PAM interface is initialized is netcool. PAM authentication is available on Linux, Solaris, and HP-UX 11 platforms only.</p> <p>If KERBEROS is specified, Kerberos IV authentication server is used to verify user credentials. This is only available on Solaris systems with a Kerberos IV authentication server installed.</p> <p>If HPTCB is specified as the authentication mode, this HP-UX password protection system is used. This is only available on HP trusted (secure) systems.</p>
-configfile	string	Use string as the configuration file rather than the default file \$OMNIHOME/etc/nco_pa.conf .
-debug	numeric	Enables debug messages mode. The numeric specifies the amount of debug information required; available levels are 1 (Debug), 2 (Information), 3 (Warning), 4 (Error), and 5 (Fatal). The default is Warning.
-DNS	string	Specifies a value to override the host name in DNS environments. This must be the same as the entry in the configuration file.
-help	N/A	Displays help information about the Process Control agent and exits.
-killprocessgroup	N/A	If specified, when the process agent daemon stops a process, it also sends a signal to kill any processes in the same operating system process group.
-logfile	string	Use string as the log file rather than the default file \$OMNIHOME/log/<paname>.log , where <paname> is the name of the Process Control agent specified with -name .
-logsize	numeric	Maximum log file size in KB. The default is 1024KB , and the minimum size is 16KB .

Table 7-4 Process Control Agent Daemon Command Line Options (continued)

Option	Parameter	Description
-msgpoolsize	numeric	Specifies the number of messages available to the Process Control agent.
-name	string	Use string as the name of the server for this Process Control agent. If not specified, the default Process Control agent name is NCO_PA .
-newlog	N/A	Start a new log file. Without this option, Process Control appends to the end of any existing log file. With this option, the log is cleared first.
-noauto	N/A	If specified, the Process Control agent does not start the automatic start services.
-noconfig	N/A	If specified, the Process Control agent does not read the configuration file. This forces Process Control to start with no configuration information.
-nodaemon	N/A	By default, Process Control forks into the background to run as a <i>daemon process</i> . When -nodaemon is specified, the process runs in the <i>foreground</i> .
-password	string	Specifies the password used to log into a remote Process Control agent. If the -user option is not also specified, the user name used to make the connection is the user executing the command.
-pidfile	string	Specifies the path, relative to \$OMNIHOME , to the file in which the Process Control daemon PID is stored. Each process agent daemon must have its own PID file. The default is \$OMNIHOME/var/nco_pa.pid . This makes it possible to run more than one process agent daemon on the same machine.
-pidmsgpool	numeric	Specifies the size of the signal-handling message pool.
-redirectfile	string	Specifies a file to which the stderr and stdout messages of processes started by the process agent is directed.
-retrytime	numeric	Specifies the number of seconds a process started by Process Control must run to be considered a successful start. The default retrytime is 5. The Process Control agent attempts to restart a process if the process exits. If the process exits after retrytime seconds, the process agent attempts to restart the process immediately. If the process exits before retrytime seconds, the process agent attempts to restart the process at exponential rate of 2, 4, 8, 16, 32, ..., 256 seconds. The process agent resets the timing interval after eight attempts to start the process. If the process fails to run for more than the retrytime seconds, the RetryCount (specified in the process definition) for that process is also decremented. If the process runs successfully for at least retrytime seconds, the RetryCount is set back to its original value. If the RetryCount is 0, there is no limit to the number of restart attempts.

Table 7-4 Process Control Agent Daemon Command Line Options (continued)

Option	Parameter	Description
-roguetimeout	numeric	Specifies the time in seconds to wait for the process to shut down. The default is 30 and the minimum is 5 seconds.
-secure	N/A	All clients need to authenticate themselves with a valid username and password, specified with the -user and -password command line options. If specified, when the Process Control agent connects to another Process Control agent, login information is automatically encrypted in transmission. Do not use this option when the Process Control agent is connecting to a Process Control agent prior to version Cisco Info Center 3.5.
-stacksize	numeric	Specifies the size of the thread stack.
-ticketdir	string	Directory for Kerberos tickets if -authenticate is set to Kerberos.
-tracevtq	N/A	Enables tracing of event queue activity.
-tracemsgq	N/A	Enables tracing of message queue activity.
-tracemtx	N/A	Enables the tracing of mutex locks.
-tracenet	N/A	Enables the net library tracing.
-user	string	Specifies the user name used to log into another Process Control agent. This must be specified if connecting to a Process Control agent running in secure mode (using the -secure option).
-version	N/A	Displays version information about the Process Control agent and exits.
-walkhosttab	N/A	Specifies to search the hosts table to verify all aliases.

**Note**

A new instance of the Process Control agent cannot manage processes that were started by another instance and are still running. When the Process Control agent is stopped and restarted, it has no knowledge of such processes, and therefore starts new instances of them. The previous instances are left running.

Process Control Management

The Process Control system provides various utilities to manage and change the Cisco Info Center configuration. This section describes the following management utilities:

- **nco_pa_status**
- **nco_pa_start**
- **nco_pa_stop**
- **nco_pa_shutdown**
- **nco_pa_addentry**.

**Note**

Each utility prompts for your password.

Displaying Service Status - nco_pa_status

The **nco_pa_status** utility retrieves the status of services in the configuration. For each service, **nco_pa_status** returns a list of defined processes, the status of each process, and the UNIX process identifier. To display the service status, enter the command:

```
$OMNIHOME/bin/nco_pa_status -server <string>
```

In this command, *<string>* is the process agent name. The following is example output:

```
-----
Service Name      Process Name      Hostname  User    Status  PID
-----
Master Service    Info Server       SFOSYS1   root    RUNNING 16751
                  Proxy             SFOSYS1   root    RUNNING 16752
                  Sleep             SFOSYS1   root    RUNNING 16753
                  Info Mediator     SFOSYS1   root    RUNNING 16754
-----
```

[Table 7-5](#) describes each of the status levels.

Table 7-5 Service Status Descriptions

Status Level	Description
RUNNING	The process is running.
STARTING	A start request has been issued.
PENDING	The process is waiting for a dependency to start. This status can also indicate the process has failed to start properly (whether or not it has any dependencies).
WAITING	The process is waiting for a time dependency to complete.
DEAD	The process is not running.
ERROR	It was not possible to retrieve a status from the process agent.

If a process agent is instructed to run a process by a process agent running on a separate machine, the remote process agent does not retain a record of the process. If the remote process agent stops, the process will continue to run. When the remote process agent restarts, it has no record of the process, and therefore the process status for this orphan process is listed as *DEAD*.

You can manually restart the process using the **nco_pa_start** utility. [Table 7-6](#) describes the Process Control service **nco_pa_status** command line options.

Command Line Options

Table 7-6 Process Control Service Status Command Line Options

Option	Parameter	Description
-help	N/A	Displays help on the command line options and exits.
-nosecure	N/A	You must use this option to connect to Process Control agents for releases prior to Cisco Info Center version 3.5.
-password	string	Password.
-server	string	Name of Process Control agent to contact.
-user	string	User name. The default is the user running the command.
-version	N/A	Displays software version information and exits.

Starting a Service or Process - nco_pa_start

The **nco_pa_start** utility starts a service or process at any location in the configuration. You can only specify a single service or process. If the service or process has already been started, the command is ignored.

Command Line Options

[Table 7-7](#) describes the Process Control service **nco_pa_start** command line options.

Table 7-7 Process Control Service Start Command Line Options

Option	Parameter	Description
-help	N/A	Displays help about the command line options and exits.
-nosecure	N/A	You must use this option to connect to Process Control agents for releases prior to Cisco Info Center version 3.5.
-password	string	Password.
-process	string	Name of the process to start.
-server	string	Name of Process Control agent to contact.
-service	string	Name of the service to start.
-user	string	User name. The default is the user running the command.
-version	N/A	Displays software version information and exits.

Stopping a Service or Process - nco_pa_stop

The **nco_pa_stop** utility stops a service or process at any location in the configuration. You can only specify a single service or process. If the service or process has already been stopped, the command is ignored.

Command Line Options

Table 7-8 describes the Process Control **nco_pa_stop** command line options.

Table 7-8 Process Control Stop Command Line Options

Option	Parameter	Description
-force	N/A	If specified, no warning is output if the process or service is not running.
-help	N/A	Displays help about the command line options and exits.
-nosecure	N/A	You must use this option to connect to Process Control agents for releases prior to Cisco Info Center version 3.5.
-password	string	Password.
-process	string	Name of the process to stop.
-server	string	Name of Process Control agent to contact.
-service	string	Name of the service to stop.
-user	string	User name. The default is the user running the command.
-version	N/A	Displays software version information and exits.

Shutting Down a Process Control Agent - nco_pa_shutdown

The **nco_pa_shutdown** utility shuts down a complete Process Control agent and optionally stops associated services and processes.

Command Line Options

Table 7-9 describes the Process Control **nco_pa_shutdown** command line options.

Table 7-9 Process Control Shutdown Command Line Options

Option	Parameter	Description
-help	N/A	Displays help about the command line options and exits.
-nosecure	N/A	You must use this option to connect to Process Control agents for releases prior to Cisco Info Center version 3.5.
-option	string	Specifies how the shutdown is completed. Can be STOP to shut down all managed processes or LEAVE to leave the managed processes running after the shutdown. If -option is not specified on the command line, the utility displays a menu with the shutdown options and prompts for which type of shutdown to perform.

Table 7-9 Process Control Shutdown Command Line Options (continued)

Option	Parameter	Description
-password	string	Password.
-server	string	Name of Process Control agent to shutdown.
-user	string	User name. The default is the user running the command.
-version	N/A	Displays software version information and exits.

Adding a New Service or Process to a Running PAD - nco_pa_addentry

The `nco_pa_addentry` utility enables you to add a new service or process to a running Process Control agent.



Note

The new service or process is not added to the process agent configuration file.

Command Line Options

Table 7-10 describes the Process Control `nco_pa_addentry` command line options.

Table 7-10 Process Control Add Entry Command Line Options

Option	Parameter	Description
-alert_msg	string	Specifies the message to send to syslog if the process exits.
-auto -nonauto	N/A	If auto , the service or process is started as soon as the Process Control agent is started. By default, the service must be started manually with the nco_pa_start command.
-command	string	Specifies the process command line.
-delay	numeric	Specifies the time delay in seconds before the specified process is started.
-depend	string	Specifies the process on which the specified process depends.
-help	N/A	Displays help about the command line options and exits.
-host	string	Specifies the host on which to run the process.
-managed -unmanaged	N/A	If managed (the default), the process is restarted automatically if it exits.
-master -nonmaster	N/A	If master (the default), the service type is set to master.
-nosecure	N/A	You must use this option to connect to Process Control agents for releases prior to Cisco Info Center version 3.5.
-pa_aware -not_pa_aware	N/A	If pa_aware , the ProcessType is set to PaPA_AWARE . By default, the process is not PA aware.
-parentservic	string	Specifies the service to which to add the process.

Table 7-10 Process Control Add Entry Command Line Options (continued)

Option	Parameter	Description
-password	string	Specifies the password to use when connecting to the Process Control agent.
-process	string	Name of the process to add.
-restart_msg	string	Specifies the message to send to the syslog if the process is restarted.
-retrycount	numeric	Specifies the number of restart attempts to be made if the process exits within the time specified by the process agent daemon -retrytime command line option. If set to 0, there is no limit to the number of restart attempts. The default is 0.
-runas	numeric	Specifies the user ID to run the process as.
-server	string	Name of Process Control agent to contact. The default is NCO_PA .
-service	string	Name of the service to add.
-user	string	Specifies the user name to use when connecting to the Process Control agent. The default is the user running the command.
-version	N/A	Displays software version information and exits.