



## Configuring IE2100 CNS Agents

This chapter describes how to configure the Intelligence Engine 2100 (IE2100) Series Cisco Networking Services (CNS) embedded agents on your Catalyst 3550switch.



### Note

For complete syntax and usage information for the commands used in this section, refer to the *Cisco Intelligence Engine 2100 Series Configuration Registrar Manual*, and select **Cisco IOS Software Release 12.2 > New Feature Documentation > 12.2(2)T** on Cisco.com.

This chapter consists of these sections:

- [Understanding IE2100 Series Configuration Registrar Software, page 5-1](#)
- [Understanding CNS Embedded Agents, page 5-5](#)
- [Configuring CNS Embedded Agents, page 5-6](#)
- [Displaying CNS Configuration, page 5-13](#)

## Understanding IE2100 Series Configuration Registrar Software

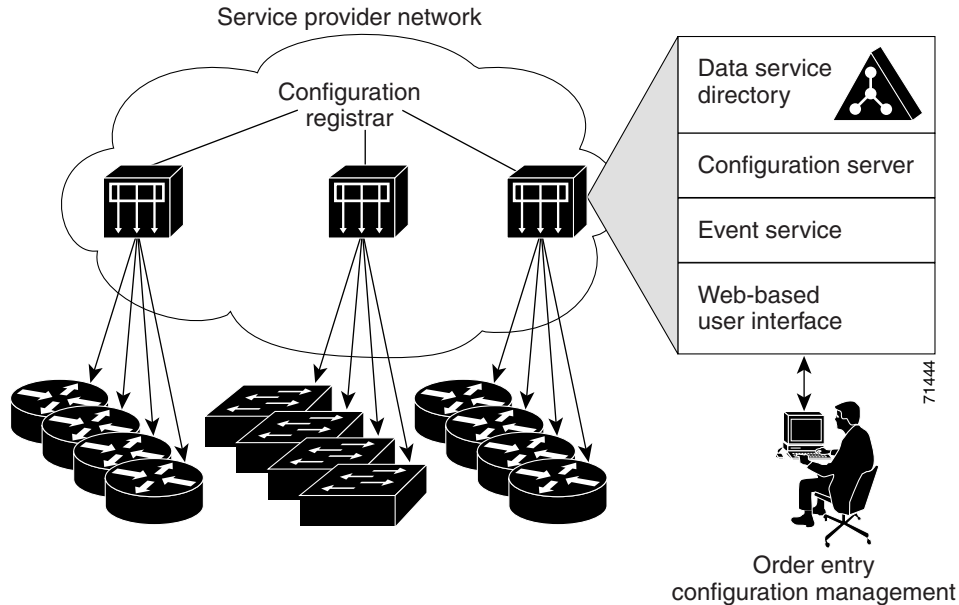
The IE2100 Series Configuration Registrar is a network management device that acts as a configuration service for automating the deployment and management of network devices and services (see [Figure 5-1](#)). Each Configuration Registrar manages a group of Cisco IOS devices (switches and routers) and the services that they deliver, storing their configurations and delivering them as needed. The Configuration Registrar automates initial configurations and configuration updates by generating device-specific configuration changes, sending them to the device, executing the configuration change, and logging the results.

The Configuration Registrar supports standalone and server modes and has these CNS components:

- Configuration service (web server, file manager, and namespace mapping server)
- Event service (event gateway)
- Data service directory (data models and schema)

In standalone mode, the Configuration Registrar supports an embedded CNS Directory Service. In this mode, no external directory or other data store is required. In server mode, the Configuration Registrar supports the use of a user-defined external directory.

Figure 5-1 Configuration Registrar Architectural Overview



These sections contain this conceptual information:

- [CNS Configuration Service, page 5-2](#)
- [CNS Event Service, page 5-3](#)
- [What You Should Know About ConfigID, DeviceID, and Host Name, page 5-3](#)

## CNS Configuration Service

The CNS Configuration Service is the core component of the Configuration Registrar. It consists of a configuration server that works with CNS configuration agents located on the switch. The CNS Configuration Service delivers device and service configurations to the switch for initial configuration and mass reconfiguration by logical groups. Switches receive their initial configuration from the CNS Configuration Service when they start up on the network for the first time.

The CNS Configuration Service uses the CNS Event Service to send and receive configuration change events and to send success and failure notifications.

The configuration server is a web server that uses configuration templates and the device-specific configuration information stored in the embedded (standalone mode) or remote (server mode) directory.

Configuration templates are text files containing static configuration information in the form of CLI commands. In the templates, variables are specified using lightweight directory access protocol (LDAP) URLs that reference the device-specific configuration information stored in a directory.

The configuration agent can perform a syntax check on received configuration files and publish events to indicate the success or failure of the syntax check. The configuration agent can either apply configurations immediately or delay the application until receipt of a synchronization event from the configuration server.

## CNS Event Service

The Configuration Registrar uses the CNS Event Service for receipt and generation of configuration events. The CNS event agent resides on the switch and facilitates the communication between the switch and the event gateway on the Configuration Registrar.

The CNS Event Service is a highly-scalable publish-and-subscribe communication method. The CNS Event Service uses subject-based addressing to send messages to their destinations. Subject-based addressing conventions define a simple, uniform namespace for messages and their destinations.

## NameSpace Mapper

The Configuration Registrar includes the NameSpace Mapper (NSM) that provides a lookup service for managing logical groups of devices based on application, device ID or group ID, and event.

Cisco IOS devices recognize only event subject-names that match those configured in Cisco IOS software; for example, `cisco.cns.config.load`. You can use the namespace mapping service to designate events by using any desired naming convention. When you have populated your data store with your subject names, NSM resolves your event subject-name strings to those known by IOS.

For a subscriber, when given a unique device ID and event, the namespace mapping service returns a set of events to which to subscribe. Similarly, for a publisher, when given a unique group ID, device ID, and event, the mapping service returns a set of events on which to publish.

## What You Should Know About ConfigID, DeviceID, and Host Name

The Configuration Registrar assumes that a unique identifier is associated with each configured switch. This unique identifier can take on multiple synonyms, where each synonym is unique within a particular namespace. The event service uses namespace content for subject-based addressing of messages.

The Configuration Registrar intersects two namespaces, one for the event bus and the other for the configuration server. Within the scope of the configuration server namespace, the term `configID` is the unique identifier for a device. Within the scope of the event bus namespace, the term `deviceID` is the CNS unique identifier for a device.

Because the Configuration Registrar uses both the event bus and the configuration server to provide configurations to devices, you must define both `configID` and `deviceID` for each configured switch.

Within the scope of a single instance of the configuration server, no two configured switches can share the same value for `configID`. Within the scope of a single instance of the event bus, no two configured switches can share the same value for `deviceID`.

## ConfigID

Each configured switch has a unique `configID`, which serves as the key into the Configuration Registrar directory for the corresponding set of switch CLI attributes. The `configID` defined on the switch must match the `configID` for the corresponding switch definition on the Configuration Registrar.

The `configID` is fixed at boot time and cannot be changed until reboot, even when the switch host name is reconfigured.

## DeviceID

Each configured switch participating on the event bus has a unique deviceID, which is analogous to the switch source address so that the switch can be targeted as a specific destination on the bus. All switches configured with the **cns config partial** global configuration command must access the event bus. Therefore, the deviceID, as originated on the switch, must match the deviceID of the corresponding switch definition in the Configuration Registrar.

The origin of the deviceID is defined by the Cisco IOS host name of the switch. However, the deviceID variable and its usage reside within the event gateway, which is adjacent to the switch.

The logical Cisco IOS termination point on the event bus is embedded in the event gateway, which in turn functions as a proxy on behalf of the switch. The event gateway represents the switch and its corresponding deviceID to the event bus.

The switch declares its host name to the event gateway immediately after the successful connection to the event gateway. The event gateway couples the deviceID value to the Cisco IOS host name each time this connection is established. The event gateway caches this deviceID value for the duration of its connection to the switch.

## Host Name and DeviceID

The deviceID is fixed at the time of the connection to the event gateway and does not change even when the switch host name is reconfigured.

When changing the switch host name on the switch, the only way to refresh the deviceID is to break the connection between the switch and the event gateway. Enter the **no cns event** global configuration command followed by the **cns event** global configuration command.

When the connection is re-established, the switch sends its modified host name to the event gateway. The event gateway redefines the deviceID to the new value.



### Caution

---

When using the Configuration Registrar user interface, you must first set the deviceID field to the host name value that the switch acquires *after*—not *before*—you use the **cns config initial** global configuration command at the switch. Otherwise, subsequent **cns config partial** global configuration command operations malfunction.

---

## Using Host Name, DeviceID, and ConfigID

In standalone mode, when a host name value is set for a switch, the configuration server uses the host name as the deviceID when an event is sent on host name. If the host name has not been set, the event is sent on the `cn=<value>` of the device.

In server mode, the host name is not used. In this mode, the unique deviceID attribute is always used for sending an event on the bus. If this attribute is not set, you cannot update the switch.

These and other associated attributes (tag value pairs) are set when you run **Setup** on the Configuration Registrar.



### Note

---

For more information about running the setup program on the Configuration Registrar, refer to the *Cisco Intelligence Engine 2100 Series Configuration Registrar Manual*.

---

# Understanding CNS Embedded Agents

The CNS event agent feature allows the switch to publish and subscribe to events on the event bus and works with the CNS configuration agent. The CNS configuration agent feature supports the switch by providing:

- Initial configurations
- Incremental (partial) configurations
- Synchronized configuration updates

## Initial Configuration

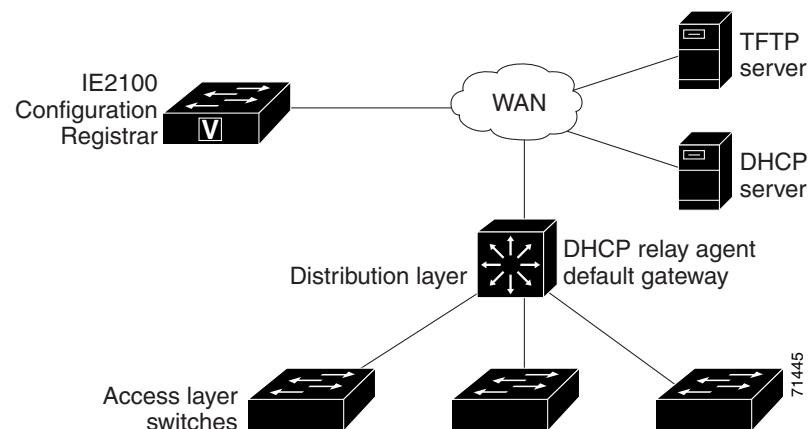
When the switch first comes up, it attempts to get an IP address by broadcasting a Dynamic Host Configuration Protocol (DHCP) request on the network. Assuming there is no DHCP server on the subnet, the distribution switch acts as a DHCP relay agent and forwards the request to the DHCP server. Upon receiving the request, the DHCP server assigns an IP address to the new switch and includes the Trivial File Transfer Protocol (TFTP) server IP address, the path to the bootstrap configuration file, and the default gateway IP address in a unicast reply to the DHCP relay agent. The DHCP relay agent forwards the reply to the switch.

The switch automatically configures the assigned IP address on interface VLAN 1 (the default) and downloads the bootstrap configuration file from the TFTP server. Upon successful download of the bootstrap configuration file, the switch loads the file in its running configuration.

The embedded CNS agents initiate communication with the IE2100 Configuration Registrar by using the appropriate configID and eventID. The Configuration Registrar maps the configID to a template and downloads the full configuration file to the switch.

Figure 5-2 shows a sample network configuration for retrieving the initial bootstrap configuration file by using DHCP-based autoconfiguration.

**Figure 5-2 Initial Configuration Overview**



## Incremental (Partial) Configuration

After the network is running, new services can be added by using the CNS configuration agent. Incremental (partial) configurations can be sent to the switch. The actual configuration can be sent as an event payload by way of the event gateway (push operation) or as a signal event that triggers the switch to initiate a pull operation.

The switch can check the syntax of the configuration before applying it. If the syntax is correct, the switch applies the incremental configuration and publishes an event that signals success to the configuration server. If the switch does not apply the incremental configuration, it publishes an event showing an error status. When the switch has applied the incremental configuration, it can write it to nonvolatile RAM (NVRAM) or wait until signaled to do so.

## Synchronized Configuration

When the switch receives a configuration, it can defer application of the configuration upon receipt of a write-signal event. The write-signal event tells the switch not to save the updated configuration into its NVRAM. The switch uses the updated configuration as its running configuration. This ensures that the switch configuration is synchronized with other network activities before saving the configuration in NVRAM for use at the next reboot.

## Configuring CNS Embedded Agents

The CNS agents embedded in the switch IOS software allow the switch to be connected and automatically configured as described in the “[Enabling Automated CNS Configuration](#)” section on [page 5-6](#). If you want to change the configuration or install a custom configuration, see these sections for instructions:

- [Enabling the CNS Event Agent, page 5-8](#)
- [Enabling the CNS Configuration Agent, page 5-9](#)

## Enabling Automated CNS Configuration

To enable automated CNS configuration of the switch, you must first complete the prerequisites in [Table 5-1](#). When you complete them, power on the switch. At the **setup** prompt, do nothing: The switch begins the initial configuration as described in the “[Initial Configuration](#)” section on [page 5-5](#). When the full configuration file is loaded on your switch, you need to do nothing else.

**Table 5-1 Prerequisites for Enabling Automatic Configuration**

Device	Required Configuration
Access switch	Factory default (no configuration file)
Distribution switch	<ul style="list-style-type: none"> <li>• IP helper address</li> <li>• Enable DHCP relay agent</li> <li>• IP routing (if used as default gateway)</li> </ul>

**Table 5-1 Prerequisites for Enabling Automatic Configuration (continued)**

Device	Required Configuration
DHCP server	<ul style="list-style-type: none"> <li>• IP address assignment</li> <li>• TFTP server IP address</li> <li>• Path to bootstrap configuration file on the TFTP server</li> <li>• Default gateway IP address</li> </ul>
TFTP server	<ul style="list-style-type: none"> <li>• Create a bootstrap configuration file that includes the CNS configuration commands that enable the switch to communicate with the IE2100 Configuration Registrar.</li> <li>• Configure the switch to use either the switch MAC address or the serial number (instead of the default host name) to generate the configID and eventID.</li> <li>• Configure the CNS event agent to push the configuration file to the switch.</li> </ul>
IE2100 Configuration Registrar	Create one or more templates for each type of device, and map the configID of the device to the template.

**Note**

For more information about running the setup program and creating templates on the Configuration Registrar, refer to the *Cisco Intelligence Engine 2100 Series Configuration Registrar Manual*.

## Enabling the CNS Event Agent


**Note**

You must enable the CNS event agent on the switch before you enable the CNS configuration agent.

Beginning in privileged EXEC mode, follow these steps to enable the CNS event agent on the switch:

	Command	Purpose
Step 1	<code>configure terminal</code>	Enter global configuration mode.
Step 2	<code>cns event {ip-address   hostname} [port-number] [backup] [init-retry retry-count] [keepalive seconds retry-count] [source ip-address]</code>	<p>Enable the event agent, and enter the gateway parameters.</p> <ul style="list-style-type: none"> <li>For <code>{ip-address   hostname}</code>, enter either the IP address or the host name of the event gateway.</li> <li>(Optional) For <code>port number</code>, enter the port number for the event gateway. The default port number is 11011.</li> <li>(Optional) Enter <b>backup</b> to show that this is the backup gateway. (If omitted, this is the primary gateway.)</li> <li>(Optional) For <code>init-retry retry-count</code>, enter the number of initial retries before switching to backup. The default is 3.</li> <li>(Optional) For <code>keepalive seconds</code>, enter how often the switch sends keepalive messages. For <code>retry-count</code>, enter the number of unanswered keepalive messages that the switch sends before the connection is terminated. The default for each is 0.</li> <li>(Optional) For <code>source ip-address</code>, enter the source IP address of this device.</li> </ul> <p><b>Note</b> Though visible in the command-line help string, the <b>encrypt</b> and <b>force-fmt1</b> keywords are not supported.</p>
Step 3	<code>end</code>	Return to privileged EXEC mode.
Step 4	<code>show cns event connections</code>	Verify information about the event agent.
Step 5	<code>show running-config</code>	Verify your entries.
Step 6	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

To disable the CNS event agent, use the **no cns event** *{ip-address | hostname}* global configuration command.

This example shows how to enable the CNS event agent, set the IP address gateway to 10.180.1.27, set 120 seconds as the keepalive interval, and set 10 as the retry count.

```
Switch(config)# cns event 10.180.1.27 keepalive 120 10
```

## Enabling the CNS Configuration Agent

After enabling the CNS event agent, start the CNS configuration agent on the switch. You can enable the configuration agent with these commands:

- the **cns config initial** global configuration command enables the configuration agent and initiates an initial configuration on the switch.
- the **cns config partial** global configuration command enables the configuration agent and initiates a partial configuration on the switch. You can then remotely send incremental configurations to the switch from the Configuration Registrar.

## Enabling an Initial Configuration

Beginning in privileged EXEC mode, follow these steps to enable the CNS configuration agent and initiate an initial configuration on the switch:

	Command	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>cns config connect-intf</b> <i>interface-prefix</i> [ <b>ping-interval</b> <i>seconds</i> ] [ <b>retries</b> <i>num</i> ]	Enter the connect-interface-config submode, and specify the interface for connecting to the Configuration Registrar. <ul style="list-style-type: none"> <li>• Enter the <i>interface-prefix</i> for the connecting interface. You must specify the interface type but need not specify the interface number.</li> <li>• (Optional) For <b>ping-interval</b> <i>seconds</i>, enter the interval between successive ping attempts. The range is 1 to 30 seconds. The default is 10 seconds.</li> <li>• (Optional) For <b>retries</b> <i>num</i>, enter the number of ping retries. The range is 1 to 30. The default is 5.</li> </ul>

	Command	Purpose
Step 3	<b>config-cli</b> or <b>line-cli</b>	Enter <b>config-cli</b> to connect to the Configuration Registrar through the interface defined in <b>cns config connect-intf</b> . Enter <b>line-cli</b> to connect to the Registrar through modem dialup lines.  <b>Note</b> The <b>config-cli</b> interface configuration command accepts the special directive character <b>&amp;</b> that acts as a placeholder for the interface name. When the configuration is applied, the <b>&amp;</b> is replaced with the interface name. For example, to connect through FastEthernet0/0, the command <b>config-cli ip route 0.0.0.0 0.0.0.0 &amp;</b> generates the command <b>ip route 0.0.0.0 0.0.0.0 FastEthernet0/0</b> .
Step 4	<b>exit</b>	Return to global configuration mode.
Step 5	<b>hostname</b> <i>name</i>	Enter the host name for the switch.
Step 6	<b>ip route</b> <i>network-number</i>	Establish a static route to the Configuration Registrar whose IP address is <i>network-number</i> .
Step 7	<b>cns id</b> <i>interface num</i> { <b>dns-reverse</b>   <b>ipaddress</b>   <b>mac-address</b> } [ <b>event</b> ] or <b>cns id</b> { <b>hardware-serial</b>   <b>hostname</b>   <b>string</b> <i>string</i> } [ <b>event</b> ]	Set the unique eventID or configID used by the Configuration Registrar. <ul style="list-style-type: none"> <li>For <i>interface num</i>, enter the type of interface—for example, Ethernet, Group-Async, Loopback, or Virtual-Template. This setting specifies from which interface the IP or MAC address should be retrieved to define the unique ID.</li> <li>For {<b>dns-reverse</b>   <b>ipaddress</b>   <b>mac-address</b>} enter <b>dns-reverse</b> to retrieve the host name and assign it as the unique ID, enter <b>ipaddress</b> to use the IP address, or enter <b>mac-address</b> to use the MAC address as the unique ID.</li> <li>(Optional) Enter <b>event</b> to set the ID to be the event-id value used to identify the switch.</li> <li>For {<b>hardware-serial</b>   <b>hostname</b>   <b>string</b> <i>string</i>}, enter <b>hardware-serial</b> to set the switch serial number as the unique ID, enter <b>hostname</b> (the default) to select the switch host name as the unique ID, or enter an arbitrary text string for <b>string</b> <i>string</i> as the unique ID.</li> </ul>

	Command	Purpose
Step 8	<b>cns config initial</b> { <i>ip-address</i>   <i>hostname</i> } [ <i>port-number</i> ] [ <i>event</i> ] [ <i>no-persist</i> ] [ <i>page page</i> ] [ <i>source ip-address</i> ] [ <i>syntax-check</i> ]	<p>Enable the configuration agent, and initiate an initial configuration.</p> <ul style="list-style-type: none"> <li>For {<i>ip-address</i>   <i>hostname</i>}, enter the IP address or the host name of the configuration server.</li> <li>(Optional) For <i>port-number</i>, enter the port number of the configuration server. The default port number is 80.</li> <li>(Optional) Enable <b>event</b> for configuration success, failure, or warning messages when the configuration is finished.</li> <li>(Optional) Enable <b>no-persist</b> to suppress the automatic writing to NVRAM of the configuration pulled as a result of entering the <b>cns config initial</b> global configuration command. If the <b>no-persist</b> keyword is not entered, using the <b>cns config initial</b> command causes the resultant configuration to be automatically written to NVRAM.</li> <li>(Optional) For <b>page</b> <i>page</i>, enter the web page of the initial configuration. The default is <i>/Config/config/asp</i>.</li> <li>(Optional) Enter <b>source</b> <i>ip-address</i> to use for source IP address.</li> <li>(Optional) Enable <b>syntax-check</b> to check the syntax when this parameter is entered.</li> </ul> <p><b>Note</b> Though visible in the command-line help string, the <b>encrypt</b> keyword is not supported.</p>
Step 9	<b>end</b>	Return to privileged EXEC mode.
Step 10	<b>show cns config connections</b>	Verify information about the configuration agent.
Step 11	<b>show running-config</b>	Verify your entries.

To disable the CNS configuration agent, use the **no cns config initial** {*ip-address* | *hostname*} global configuration command.

This example shows how to configure an initial configuration on a remote switch. The switch host name is the unique ID. The CNS Configuration Registrar IP address is 172.28.129.22.

```
Switch(config)# cns config connect-intf serial ping-interval 1 retries 1
Switch(config-cns-conn-if)# config-cli ip address negotiated
Switch(config-cns-conn-if)# config-cli encapsulation ppp
Switch(config-cns-conn-if)# config-cli ip directed-broadcast
Switch(config-cns-conn-if)# config-cli no keepalive
Switch(config-cns-conn-if)# config-cli no shutdown
Switch(config-cns-conn-if)# exit
Switch(config)# hostname RemoteSwitch
RemoteSwitch(config)# ip route 10.1.1.1 255.255.255.255 11.11.11.1
RemoteSwitch(config)# cns id Ethernet 0 ipaddress
RemoteSwitch(config)# cns config initial 10.1.1.1 no-persist
```

## Enabling a Partial Configuration

Beginning in privileged EXEC mode, follow these steps to enable the CNS configuration agent and to initiate a partial configuration on the switch:

	Command	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>cns config partial</b> { <i>ip-address</i>   <i>hostname</i> } [ <i>port-number</i> ] [ <b>source</b> <i>ip-address</i> ]	<p>Enable the configuration agent, and initiate a partial configuration.</p> <ul style="list-style-type: none"> <li>For {<i>ip-address</i>   <i>hostname</i>}, enter the IP address or the host name of the configuration server.</li> <li>(Optional) For <i>port-number</i>, enter the port number of the configuration server. The default port number is 80.</li> <li>(Optional) Enter <b>source</b> <i>ip-address</i> to use for the source IP address.</li> </ul> <p><b>Note</b> Though visible in the command-line help string, the <b>encrypt</b> keyword is not supported.</p>
Step 3	<b>end</b>	Return to privileged EXEC mode.
Step 4	<b>show cns config stats</b> or <b>show cns config outstanding</b>	Verify information about the configuration agent.
Step 5	<b>show running-config</b>	Verify your entries.
Step 6	<b>copy running-config startup-config</b>	(Optional) Save your entries in the configuration file.

To disable the CNS configuration agent, use the **no cns config partial** {*ip-address* | *hostname*} global configuration command. To cancel a partial configuration, use the **cns config cancel** privileged EXEC command.

# Displaying CNS Configuration

You can use the privileged EXEC commands in [Table 5-2](#) to display CNS Configuration information.

**Table 5-2** *Displaying CNS Configuration*

<b>Command</b>	<b>Purpose</b>
<b>show cns config connections</b>	Displays the status of the CNS configuration agent connections.
<b>show cns config outstanding</b>	Displays information about incremental (partial) CNS configurations that have started but are not yet completed.
<b>show cns config stats</b>	Displays statistics about the CNS configuration agent.
<b>show cns event connections</b>	Displays the status of the CNS event agent connections.
<b>show cns event stats</b>	Displays statistics about the CNS event agent.
<b>show cns event subject</b>	Displays a list of event agent subjects that are subscribed to by applications.

