



Configuring Smart Call Home

This chapter describes how to configure the Smart Call Home feature of the Cisco NX-OS devices.

This chapter contains the following sections:

- [Finding Feature Information, on page 1](#)
- [About Smart Call Home, on page 1](#)
- [Prerequisites for Smart Call Home, on page 8](#)
- [Guidelines and Limitations for Smart Call Home, on page 8](#)
- [Default Settings for Smart Call Home, on page 9](#)
- [Configuring Smart Call Home, on page 9](#)
- [Verifying the Smart Call Home Configuration, on page 23](#)
- [Configuration Examples for Smart Call Home, on page 24](#)
- [Additional References, on page 25](#)
- [Feature History for Smart Call Home, on page 38](#)

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at <https://tools.cisco.com/bugsearch/> and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information" chapter or the Feature History table in this chapter.

About Smart Call Home

Smart Call Home provides an email-based notification for critical system policies. A range of message formats are available for compatibility with pager services, standard email, or XML-based automated parsing applications. You can use this feature to page a network support engineer, email a Network Operations Center, or use Cisco Smart Call Home services to automatically generate a case with the Technical Assistance Center.

Smart Call Home offers the following features:

- Automatic execution and attachment of relevant CLI command output.
- Multiple message format options such as the following:
 - Short Text—Suitable for pagers or printed reports.

- Full Text—Fully formatted message information suitable for human reading.
- XML—Machine-readable format that uses Extensible Markup Language (XML) and Adaptive Messaging Language (AML) XML schema definition (XSD). The AML XSD is published on the Cisco.com website. The XML format enables communication with the Technical Assistance Center.
- Multiple concurrent message destinations. You can configure up to 50 email destination addresses for each destination profile.

Destination Profiles

A destination profile includes the following information:

- One or more alert groups—The group of alerts that trigger a specific Smart Call Home message if the alert occurs.
- One or more email destinations—The list of recipients for the Smart Call Home messages generated by alert groups assigned to this destination profile.
- Message format—The format for the Smart Call Home message (short text, full text, or XML).
- Message severity level—The Smart Call Home severity level that the alert must meet before Cisco NX-OS generates a Smart Call Home message to all email addresses in the destination profile. Cisco NX-OS does not generate an alert if the Smart Call Home severity level of the alert is lower than the message severity level set for the destination profile.

You can also configure a destination profile to allow periodic inventory update messages by using the inventory alert group that will send out periodic messages daily, weekly, or monthly.

Cisco NX-OS supports the following predefined destination profiles:

- CiscoTAC-1—Supports the Cisco-TAC alert group in XML message format. This profile is preconfigured with the callhome@cisco.com email contact, maximum message size, and message severity level 0. You cannot change any of the default information for this profile.
- full-text-destination—Supports the full text message format.
- short-text-destination—Supports the short text message format.

Smart Call Home Alert Groups

An alert group is a predefined subset of Smart Call Home alerts that are supported in all Cisco Nexus devices. Alert groups allow you to select the set of Smart Call Home alerts that you want to send to a predefined or custom destination profile. The device sends Smart Call Home alerts to email destinations in a destination profile only if that Smart Call Home alert belongs to one of the alert groups associated with that destination profile and if the alert has a Smart Call Home message severity at or above the message severity set in the destination profile.

The following table lists the supported alert groups and the default CLI command output included in Smart Call Home messages generated for the alert group.

| Alert Group | Description | Executed Commands |
|--------------------|--|---|
| Cisco-TAC | All critical alerts from the other alert groups destined for Smart Call Home. | Execute commands based on the alert group that originates the alert. |
| Configuration | Periodic events related to configuration. | show module show version |
| Diagnostic | Events generated by diagnostics. | show diagnostic result module all detail show diagnostic result module <i>number</i> detail show hardware show logging last 200 show module show sprom all show tech-support gold show tech-support ha show tech-support platform show version |
| EEM | Events generated by EEM. | show diagnostic result module all detail show diagnostic result module <i>number</i> detail show module show tech-support gold show tech-support ha show tech-support platform |
| Environmental | Events related to power, fan, and environment-sensing elements such as temperature alarms. | show environment show logging last 200 show module show version |

| Alert Group | Description | Executed Commands |
|---------------------|--|---|
| Inventory | Inventory status that is provided whenever a unit is cold booted or when FRUs are inserted or removed. This alert is considered a noncritical event, and the information is used for status and entitlement. | show inventory show license usage show module show sprom all show system uptime show version |
| License | Events related to licensing and license violations. | show logging last 200 |
| Linecard hardware | Events related to standard or intelligent switching modules. | show diagnostic result module all detail show diagnostic result module <i>number</i> detail show hardware show logging last 200 show module show sprom all show tech-support ethpm show tech-support gold show tech-support ha show tech-support platform show version |
| Supervisor hardware | Events related to supervisor modules. | show diagnostic result module all detail show hardware show logging last 200 show module show sprom all show tech-support ethpm show tech-support gold show tech-support ha show tech-support platform show version |

| Alert Group | Description | Executed Commands |
|-------------------|--|---|
| Syslog port group | Events generated by the syslog PORT facility. | show license usage show logging last 200 |
| System | Events generated by failure of a software system that is critical to unit operation. | show diagnostic result module all detail show hardware show logging last 200 show module show sprom all show tech-support ethpm show tech-support gold show tech-support ha show tech-support platform |
| Test | User-generated test message. | show module show version |

Smart Call Home maps the syslog severity level to the corresponding Smart Call Home severity level for syslog port group messages.

You can customize predefined alert groups to execute additional CLI **show** commands when specific events occur and send that **show** output with the Smart Call Home message.

You can add **show** commands only to full text and XML destination profiles. Short text destination profiles do not support additional **show** commands because they only allow 128 bytes of text.

Smart Call Home Message Levels

Smart Call Home allows you to filter messages based on their level of urgency. You can associate each predefined or user-defined destination profile with a Smart Call Home threshold from 0 (least urgent) to 9 (most urgent). The default is 0 (all messages are sent).

Syslog severity levels are mapped to the Smart Call Home message level.



Note Smart Call Home does not change the syslog message level in the message text.

The following table lists each Smart Call Home message level keyword and the corresponding syslog level for the syslog port alert group.

Table 1: Severity and Syslog Level Mapping

| Smart Call Home Level | Keyword | Syslog Level | Description |
|-----------------------|--------------|-----------------|--|
| 9 | Catastrophic | N/A | Network-wide catastrophic failure. |
| 8 | Disaster | N/A | Significant network impact. |
| 7 | Fatal | Emergency (0) | System is unusable. |
| 6 | Critical | Alert (1) | Critical conditions that indicate that immediate attention is needed. |
| 5 | Major | Critical (2) | Major conditions. |
| 4 | Minor | Error (3) | Minor conditions. |
| 3 | Warning | Warning (4) | Warning conditions. |
| 2 | Notification | Notice (5) | Basic notification and informational messages. Possibly independently insignificant. |
| 1 | Normal | Information (6) | Normal event signifying return to normal state. |
| 0 | Debugging | Debug (7) | Debugging messages. |

Obtaining Smart Call Home

If you have a service contract directly with Cisco, you can register for the Smart Call Home service. Smart Call Home analyzes Smart Call Home messages and provides background information and recommendations. For known issues, particularly online diagnostics failures, Automatic Service Requests are generated with the Cisco TAC.

Smart Call Home offers the following features:

- Continuous device health monitoring and real-time diagnostic alerts.
- Analysis of Smart Call Home messages and, if needed, Automatic Service Request generation, routed to the correct TAC team, including detailed diagnostic information to speed problem resolution.
- Secure message transport directly from your device, through an HTTP proxy server, or a downloadable Transport Gateway (TG). You can use a TG aggregation point to support multiple devices or in cases where security dictates that your devices may not be connected directly to the Internet.
- Web-based access to Smart Call Home messages and recommendations, inventory, and configuration information for all Smart Call Home devices. This feature provides access to associated field notices, security advisories, and end-of-life information.

You need the following information to register:

- The SMARTnet contract number for your device
- Your email address

- Your Cisco.com ID

For more information about Smart Call Home, see the following Smart Call Home page:
https://supportforums.cisco.com/community/netpro/solutions/smart_services/smartcallhome

Distributing Smart Call Home Using CFS

You can use Cisco Fabric Services (CFS) to distribute a Smart Call Home configuration to all CFS-enabled devices in the network. The entire Smart Call Home configuration is distributed except the device priority and the sysContact names.

For more information about CFS, see the “Configuring CFS” section.

Database Merge Guidelines

When you merge two Smart Call Home databases, the following guidelines apply:

- The merged database contains the following information:
 - A superset of all the destination profiles from the merging devices.
 - The destination profile email addresses and alert groups.
 - Other configuration information (for example, message throttling, or periodic inventory) present in the managing device.
- Destination profile names cannot be duplicated within the merging devices—even though the configurations are different, the names cannot be duplicated. If a profile name is duplicated, one of the duplicate profiles must first be deleted or the merger fails.

High Availability

Both stateful and stateless restarts are supported for Smart Call Home.

Virtualization Support

One instance of Smart Call Home is supported per virtual device context (VDC). Smart Call Home uses the contact information from the first registered VDC as the administrator contact for all VDCs on the physical device. For example, if you want the Smart Call Home to use the contact information from the default VDC, you should register using that VDC. You can update this information at the Smart Call Home web site at the following URL:

<http://www.cisco.com/go/smartcall/>

Smart Call Home registers the contacts for all other VDCs as users that can see all the Smart Call Home data for the physical device but cannot act as administrators. All registered users and the registered administrator receive all Smart Call Home notifications from all VDCs on the physical device.

By default, you are placed in the default VDC. In the default VDC, you can test Smart Call Home using the **callhome send** and **callhome test** commands. In a nondefault VDC, only the **callhome test** command is available. For more information on VDCs, see the Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide.

Smart Call Home is virtual routing and forwarding (VRF) aware. You can configure Smart Call Home to use a particular VRF to reach the Smart Call Home SMTP server.

Prerequisites for Smart Call Home

Smart Call Home has the following prerequisites:

- To send messages to an email address, you must first configure an email server. To send messages using HTTP, you must have access to an HTTPS server and have a valid certificate installed on the Cisco Nexus device.
- Your device must have IP connectivity to an email server or HTTPS server.
- You must first configure the contact name (SNMP server contact), phone, and street address information. This step is required to determine the origin of messages received.
- If you use Smart Call Home, you need an active service contract for the device that you are configuring.
- If you configure VDCs, install the appropriate license. See the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide* for configuration information and the *Cisco NX-OS Licensing Guide* for licensing information.

Guidelines and Limitations for Smart Call Home

Smart Call Home has the following configuration guidelines and limitations:

- If there is no IP connectivity or if the interface in the virtual routing and forwarding (VRF) instance to the profile destination is down, the device cannot send Smart Call Home messages.
- Smart Call Home operates with any SMTP server.
- You can configure up to five SMTP servers for Smart Call Home.
- If you distribute the Smart Call Home configuration using CFS, then the entire Smart Call Home configuration is distributed except device priority and the sysContact names.
- Currently CoPP does not protect packets for Smart Call Home using HTTP/HTTPS or SMTP method when connectivity is required inband. Return traffic for these services is subject to class-default CoPP class and leads to little to no connectivity.
- A system configured for Smart Call Home (SCH) feature where connectivity may fail during reporting if an explicit class for either the HTTPS method or the SMTP method is not defined in control-plane policing and there is continual violations in the CoPP class-default class. This issue is only seen when the configured destination from SCH is known inband.
- A syslog should be printed if inband is used for SCH where non-standard destination ports are used warning the user to add these ports. Consideration should also be made with a syslog warning when using either a HTTP or HTTPS proxy server on a non-administrative port to allow reachability to Cisco's web servers.
- In a mixed fabric environment with CFS enabled, Cisco devices running Cisco NX-OS Release 5.x can distribute 5.x configurations (multiple SMTP server support, HTTP VRF support, and HTTP proxy support) to other 5.x devices in the fabric over CFS. However, if an existing device upgrades to 5.x, these

new configurations are not distributed to that device because a CFS merge is not triggered upon an upgrade. Therefore, we recommend applying the new configurations only when all the devices in the fabric support them or performing an empty commit from an existing 5.x device (not the newly upgraded device) that has the new configurations.

Default Settings for Smart Call Home

This table lists the default settings for Smart Call Home parameters.

Table 2: Default Smart Call Home Parameters

| Parameters | Default |
|--|--|
| Destination message size for a message sent in full text format | 2,500,000 |
| Destination message size for a message sent in XML format | 2,500,000 |
| Destination message size for a message sent in short text format | 4000 |
| SMTP server port number if no port is specified | 25 |
| SMTP server priority if no priority is specified | 50 |
| Alert group association with profile | All for full-text-destination and short-text-destination profiles. The cisco-tac alert group for the CiscoTAC-1 destination profile. |
| Format type | XML |
| Smart Call Home message level | 0 (zero) |
| HTTP proxy server use | Disabled and no proxy server configured |

Configuring Smart Call Home



Note Be aware that the Cisco NX-OS commands may differ from the Cisco IOS commands.

We recommend that you complete the Smart Call Home configuration procedures in the following sequence:

1. [Configuring Contact Information, on page 10](#)
2. [Creating a Destination Profile, on page 12](#)
3. [Associating an Alert Group with a Destination Profile, on page 15](#)
4. (Optional) [Adding Show Commands to an Alert Group, on page 15](#)
5. [Enabling or Disabling Smart Call Home, on page 22](#)
6. (Optional) [Testing the Smart Call Home Configuration, on page 22](#)

Configuring Contact Information

You must configure the email, phone, and street address information for Smart Call Home. You can optionally configure the contract ID, customer ID, site ID, and switch priority information.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | configure terminal Example: <pre>switch# configure terminal switch(config)#</pre> | Enters global configuration mode. |
| Step 2 | snmp-server contact <i>sys-contact</i> Example: <pre>switch(config)# snmp-server contact personname@companyname.com</pre> | Configures the SNMP sysContact. |
| Step 3 | callhome Example: <pre>switch(config)# callhome switch(config-callhome)#</pre> | Enters Smart Call Home configuration mode. |
| Step 4 | email-contact <i>email-address</i> Example: <pre>switch(config-callhome)# email-contact admin@Mycompany.com</pre> | Configures the email address for the person primarily responsible for the device. The <i>email-address</i> can be up to 255 alphanumeric characters in email address format. Note You can use any valid email address. The address cannot contain spaces. |
| Step 5 | phone-contact <i>international-phone-number</i> Example: <pre>switch(config-callhome)# phone-contact +1-800-123-4567</pre> | Configures the phone number in international phone number format for the person primarily responsible for the device. The <i>international-phone-number</i> can be up to 17 alphanumeric characters and must be in international phone number format. Note The phone number cannot contain spaces. Use the plus (+) prefix before the number. |
| Step 6 | streetaddress <i>address</i> Example: <pre>switch(config-callhome)# streetaddress 123 Anystreet st. Anytown,AnyWhere</pre> | Configures the street address as an alphanumeric string with white spaces for the person primarily responsible for the device. The <i>address</i> can be up to 255 alphanumeric characters. Spaces are accepted. |

| | Command or Action | Purpose |
|----------------|---|---|
| Step 7 | (Optional) contract-id <i>contract-number</i> Example: switch(config-callhome)# contract-id Contract5678 | Configures the contract number for this device from the service agreement. The <i>contract-number</i> can be up to 255 alphanumeric characters in free format. |
| Step 8 | (Optional) customer-id <i>customer-number</i> Example: switch(config-callhome)# customer-id Customer123456 | Configures the customer number for this device from the service agreement. The <i>customer-number</i> can be up to 255 alphanumeric characters in free format. |
| Step 9 | (Optional) site-id <i>site-number</i> Example: switch(config-callhome)# site-id Site1 | Configures the site number for this device. The <i>site-number</i> can be up to 255 alphanumeric characters in free format. |
| Step 10 | (Optional) switch-priority <i>number</i> Example: switch(config-callhome)# switch-priority 3 | Configures the switch priority for this device. The range is from 0 to 7, with 0 being the highest priority and 7 the lowest. The default is 7. Note Switch priority is used by the operations personnel or TAC support personnel to decide which Call Home message should be responded to first. You can prioritize Call Home alerts of the same severity from each switch. |
| Step 11 | commit Example: switch(config-callhome)# commit | Commits the Smart Call Home configuration commands. |
| Step 12 | (Optional) show callhome Example: switch(config-callhome)# show callhome | Displays a summary of the Smart Call Home configuration. |
| Step 13 | (Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config | Copies the running configuration to the startup configuration. |

What to do next

Create a destination profile.

Creating a Destination Profile

You can create a user-defined destination profile and configure its message format.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | configure terminal Example: <pre>switch# configure terminal switch(config)#</pre> | Enters global configuration mode. |
| Step 2 | callhome Example: <pre>switch(config)# callhome switch(config-callhome)#</pre> | Enters Smart Call Home configuration mode. |
| Step 3 | destination-profile <i>name</i> Example: <pre>switch(config-callhome)# destination-profile Noc101</pre> | Creates a new destination profile. The name can be any alphanumeric string up to 31 characters. |
| Step 4 | destination-profile <i>name</i> format {XML full-txt short-txt} Example: <pre>switch(config-callhome)# destination-profile Noc101 format full-txt</pre> | Sets the message format for the profile. The name can be any alphanumeric string up to 31 characters. |
| Step 5 | commit Example: <pre>switch(config-callhome)# commit</pre> | Commits the Smart Call Home configuration commands. |
| Step 6 | (Optional) show callhome destination-profile [<i>profile name</i>] Example: <pre>switch(config-callhome)# show callhome destination-profile profile Noc101</pre> | Displays information about one or more destination profiles. |
| Step 7 | (Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre> | Copies the running configuration to the startup configuration. |

What to do next

Associate one or more alert groups with a destination profile.

Modifying a Destination Profile

You can modify the following attributes for a predefined or user-defined destination profile:

- Destination email address—The actual address, pertinent to the transport mechanism, to which the alert should be sent.
- Destination URL—The HTTP or HTTPS URL that defines where alerts should be sent.
- Transport method—The email or HTTP transport that determines which type of destination addresses are used.
- Message formatting—The message format used for sending the alert (full text, short text, or XML).
- Message level—The Smart Call Home message severity level for this destination profile.
- Message size—The allowed length of a Smart Call Home message sent to the email addresses in this destination profile.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | configure terminal Example: switch# configure terminal switch(config)# | Enters global configuration mode. |
| Step 2 | callhome Example: switch(config)# callhome switch(config-callhome)# | Enters Smart Call Home configuration mode. |
| Step 3 | destination-profile <i>{name CiscoTAC-1 full-txt-destination short-txt-destination}</i> email-addr <i>address</i> Example: switch(config-callhome)# destination-profile full-txt-destination email-addr person@place.com | Configures an email address for a user-defined or predefined destination profile. You can configure up to 50 email addresses in a destination profile. |
| Step 4 | destination-profile <i>{name CiscoTAC-1 full-txt-destination short-txt-destination}</i> http <i>address</i> Example: switch(config-callhome)# destination-profile CiscoTAC-1 http http://site.com/service/callhome | Configures an HTTP or HTTPS URL for a user-defined or predefined destination profile. The URL can be up to 255 characters. |
| Step 5 | destination-profile <i>{name CiscoTAC-1 full-txt-destination short-txt-destination}</i> transport-method <i>{email http}</i> | Configures an email or HTTP transport method for a user-defined or predefined destination profile. The type of transport method that you |

| | Command or Action | Purpose |
|----------------|--|---|
| | Example: <pre>switch(config-callhome)# destination-profile CiscoTAC-1 transport-method http</pre> | choose determines the configured destination addresses of that type. |
| Step 6 | destination-profile <i>{name CiscoTAC-1 full-txt-destination short-txt-destination}</i> message-level <i>number</i> Example: <pre>switch(config-callhome)# destination-profile full-txt-destination message-level 5</pre> | Configures the Smart Call Home message severity level for this destination profile. Cisco NX-OS sends only alerts that have a matching or higher Smart Call Home severity level to destinations in this profile. The range is from 0 to 9, where 9 is the highest severity level. |
| Step 7 | destination-profile <i>{name CiscoTAC-1 full-txt-destination short-txt-destination}</i> message-size <i>number</i> Example: <pre>switch(config-callhome)# destination-profile full-txt-destination message-size 100000</pre> | Configures the maximum message size for this destination profile. The range is from 0 to 5000000. The default is 2500000. |
| Step 8 | commit Example: <pre>switch(config-callhome)# commit</pre> | Commits the Smart Call Home configuration commands. |
| Step 9 | (Optional) show callhome destination-profile <i>[profile name]</i> Example: <pre>switch(config-callhome)# show callhome destination-profile profile full-text-destination</pre> | Displays information about one or more destination profiles. |
| Step 10 | (Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre> | Copies the running configuration to the startup configuration. |

What to do next

Associate one or more alert groups with a destination profile.

Associating an Alert Group with a Destination Profile

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | configure terminal Example: switch# configure terminal switch(config)# | Enters global configuration mode. |
| Step 2 | callhome Example: switch(config)# callhome switch(config-callhome)# | Enters Smart Call Home configuration mode. |
| Step 3 | destination-profile { <i>name</i> CiscoTAC-1 full-txt-destination short-txt-destination} alert-group {All Cisco-TAC Configuration Diagnostic EEM Environmental Inventory License Supervisor-Hardware Syslog-group-port System Test} Example: switch(config-callhome)# destination-profile Noc101 alert-group All | Associates an alert group with this destination profile. Use the All keyword to associate all alert groups with the destination profile. |
| Step 4 | commit Example: switch(config-callhome)# commit | Commits the Smart Call Home configuration commands. |
| Step 5 | (Optional) show callhome destination-profile [<i>profile name</i>] Example: switch(config-callhome)# show callhome destination-profile profile Noc101 | Displays information about one or more destination profiles. |
| Step 6 | (Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config | Copies the running configuration to the startup configuration. |

What to do next

Optionally add **show** commands to an alert group and then configure the SMTP email server.

Adding Show Commands to an Alert Group

You can assign a maximum of five user-defined CLI **show** commands to an alert group.



Note You cannot add user-defined CLI **show** commands to the CiscoTAC-1 destination profile.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | configure terminal Example: <pre>switch# configure terminal switch(config)#</pre> | Enters global configuration mode. |
| Step 2 | callhome Example: <pre>switch(config)# callhome switch(config-callhome)#</pre> | Enters Smart Call Home configuration mode. |
| Step 3 | alert-group {Configuration Diagnostic EEM Environmental Inventory License Supervisor-Hardware Syslog-group-port System Test} user-def-cmd <i>show-cmd</i> Example: <pre>switch(config-callhome)# alert-group Configuration user-def-cmd show ip route</pre> | Adds the show command output to any Smart Call Home messages sent for this alert group. Only valid show commands are accepted. |
| Step 4 | commit Example: <pre>switch(config-callhome)# commit</pre> | Commits the Smart Call Home configuration commands. |
| Step 5 | (Optional) show callhome user-def-cmds Example: <pre>switch(config-callhome)# show callhome user-def-cmds</pre> | Displays information about all user-defined show commands added to alert groups. |
| Step 6 | (Optional) copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre> | Copies the running configuration to the startup configuration. |

What to do next

Configure Smart Call Home to connect to the SMTP email server.

Configuring the Email Server

You must configure the SMTP server address for the Smart Call Home functionality to work. You can also configure the from and reply-to email addresses.

You can configure up to five SMTP servers for Smart Call Home. The servers are tried based on their priority. The highest priority server is tried first. If the message fails to be sent, the next server in the list is tried until the limit is exhausted. If two servers have equal priority, the one that was configured earlier is tried first.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | configure terminal Example: switch# configure terminal switch(config)# | Enters global configuration mode. |
| Step 2 | callhome Example: switch(config)# callhome switch(config-callhome)# | Enters Smart Call Home configuration mode. |
| Step 3 | transport email mail-server ip-address [port number] [priority number] [use-vrf vrf-name] Example: switch(config-callhome)# transport email mail-server 192.0.2.1 use-vrf Red | Configures the SMTP server as the domain name server (DNS) name, IPv4 address, or IPv6 address. Optionally configures the port number. The port range is from 1 to 65535. The default port number is 25. Also optionally configures the priority of the SMTP server. The priority range is from 1 to 100, with 1 being the highest priority and 100 the lowest. If you do not specify a priority, the default value of 50 is used. Also optionally configures the VRF to use when communicating with this SMTP server. The VRF specified is not used to send messages using HTTP. |
| Step 4 | (Optional) transport email from email-address Example: switch(config-callhome)# transport email from person@company.com | Configures the email from field for Smart Call Home messages. |
| Step 5 | (Optional) transport email reply-to email-address Example: switch(config-callhome)# transport email reply-to person@company.com | Configures the email reply-to field for Smart Call Home messages. |

| | Command or Action | Purpose |
|---------------|---|---|
| Step 6 | commit Example: switch(config-callhome)# commit | Commits the Smart Call Home configuration commands. |
| Step 7 | (Optional) show callhome transport Example: switch(config-callhome)# show callhome transport | Displays the transport-related configuration for Smart Call Home. |
| Step 8 | (Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config | Copies the running configuration to the startup configuration. |

What to do next

Optionally use VRFs to send Smart Call Home messages over HTTP.

Configuring VRFs To Send Messages Using HTTP

You can use VRFs to send Smart Call Home messages over HTTP. If HTTP VRFs are not configured, the default VRF is used to transport messages over HTTP.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | configure terminal Example: switch# configure terminal switch(config)# | Enters global configuration mode. |
| Step 2 | callhome Example: switch(config)# callhome switch(config-callhome)# | Enters Smart Call Home configuration mode. |
| Step 3 | transport http use-vrf <i>vrf-name</i> Example: switch(config-callhome)# transport http use-vrf Blue | Configures the VRF used to send email and other Smart Call Home messages over HTTP. |
| Step 4 | commit Example: switch(config-callhome)# commit | Commits the Smart Call Home configuration commands. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 5 | (Optional) show callhome Example: switch(config-callhome)# show callhome | Displays information about Smart Call Home. |
| Step 6 | (Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config | Copies the running configuration to the startup configuration. |

What to do next

Optionally configure Smart Call Home to send HTTP messages through an HTTP proxy server.

Configuring an HTTP Proxy Server

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | configure terminal Example: switch# configure terminal switch(config)# | Enters global configuration mode. |
| Step 2 | callhome Example: switch(config)# callhome switch(config-callhome)# | Enters Smart Call Home configuration mode. |
| Step 3 | transport http proxy server ip-address [port number] Example: switch(config-callhome)# transport http proxy server 192.0.2.1 | Configures the HTTP proxy server domain name server (DNS) name, IPv4 address, or IPv6 address. Optionally configures the port number. The port range is from 1 to 65535. The default port number is 8080. |
| Step 4 | transport http proxy enable Example: switch(config-callhome)# transport http proxy enable | Enables Smart Call Home to send all HTTP messages through the HTTP proxy server. Note You can execute this command only after the proxy server address has been configured. Note The VRF used for transporting messages through the proxy server is the same as that configured using the transport http use-vrf command. |

| | Command or Action | Purpose |
|---------------|---|---|
| Step 5 | commit Example: switch(config-callhome)# commit | Commits the Smart Call Home configuration commands. |
| Step 6 | (Optional) show callhome transport Example: switch(config-callhome)# show callhome transport | Displays the transport-related configuration for Smart Call Home. |
| Step 7 | (Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config | Copies the running configuration to the startup configuration. |

What to do next

Optionally configure your device to periodically send inventory notifications.

Configuring Periodic Inventory Notifications

You can configure the device to periodically send a message with an inventory of all software services currently enabled and running on the device along with hardware inventory information. The device generates two Smart Call Home notifications: periodic configuration messages and periodic inventory messages.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | configure terminal Example: switch# configure terminal switch(config)# | Enters global configuration mode. |
| Step 2 | callhome Example: switch(config)# callhome switch(config-callhome)# | Enters Smart Call Home configuration mode. |
| Step 3 | periodic-inventory notification [interval days] [timeofday time] Example: switch(config-callhome)# periodic-inventory notification interval 20 | Configures periodic inventory messages. The interval range is from 1 to 30 days, and the default is 7 days. The <i>time</i> argument is in HH:MM format. It defines at what time of the day every <i>X</i> days an update is sent (where <i>X</i> is the update interval). |
| Step 4 | commit Example: | Commits the Smart Call Home configuration commands. |

| | Command or Action | Purpose |
|---------------|--|--|
| | <code>switch(config-callhome)# commit</code> | |
| Step 5 | (Optional) show callhome Example: <code>switch(config-callhome)# show callhome</code> | Displays information about Smart Call Home. |
| Step 6 | (Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code> | Copies the running configuration to the startup configuration. |

What to do next

Optionally disable duplicate message throttling.

Disabling Duplicate Message Throttling

You can limit the number of duplicate messages received for the same event. By default, the device limits the number of duplicate messages received for the same event. If the number of duplicate messages sent exceeds 30 messages within a 2-hour time frame, the device discards further messages for that alert type.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code> | Enters global configuration mode. |
| Step 2 | callhome Example: <code>switch(config)# callhome</code> <code>switch(config-callhome)#</code> | Enters Smart Call Home configuration mode. |
| Step 3 | no duplicate-message throttle Example: <code>switch(config-callhome)# no duplicate-message throttle</code> | Disables duplicate message throttling for Smart Call Home. Duplicate message throttling is enabled by default. |
| Step 4 | commit Example: <code>switch(config-callhome)# commit</code> | Commits the Smart Call Home configuration commands. |
| Step 5 | (Optional) copy running-config startup-config Example: | Copies the running configuration to the startup configuration. |

| | Command or Action | Purpose |
|--|---|---------|
| | <code>switch(config)# copy running-config startup-config</code> | |

What to do next

Enable Smart Call Home.

Enabling or Disabling Smart Call Home

Once you have configured the contact information, you can enable the Smart Call Home function.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | configure terminal Example: <code>switch# configure terminal</code> <code>switch(config)#</code> | Enters global configuration mode. |
| Step 2 | callhome Example: <code>switch(config)# callhome</code> <code>switch(config-callhome)#</code> | Enters Smart Call Home configuration mode. |
| Step 3 | [no] enable Example: <code>switch(config-callhome)# enable</code> | Enables or disables Smart Call Home. Smart Call Home is disabled by default. |
| Step 4 | commit Example: <code>switch(config-callhome)# commit</code> | Commits the Smart Call Home configuration commands. |
| Step 5 | (Optional) copy running-config startup-config Example: <code>switch(config)# copy running-config startup-config</code> | Copies the running configuration to the startup configuration. |

What to do next

Optionally generate a test message.

Testing the Smart Call Home Configuration

You can generate a test message to test your Smart Call Home communications.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | configure terminal Example: switch# configure terminal switch(config)# | Enters global configuration mode. |
| Step 2 | callhome Example: switch(config)# callhome switch(config-callhome)# | Enters Smart Call Home configuration mode. |
| Step 3 | callhome send [configuration diagnostic] Example: switch(config-callhome)# callhome send diagnostic | Sends the specified Smart Call Home test message to all configured destinations. |
| Step 4 | callhome test Example: switch(config-callhome)# callhome test | Sends a test message to all configured destinations. |
| Step 5 | (Optional) copy running-config startup-config Example: switch(config)# copy running-config startup-config | Copies the running configuration to the startup configuration. |

Verifying the Smart Call Home Configuration

To display Smart Call Home configuration information, perform one of the following tasks:

| Command | Purpose |
|--|---|
| show callhome | Displays the Smart Call Home configuration. |
| show callhome destination-profile <i>name</i> | Displays one or more Smart Call Home destination profiles. |
| show callhome merge | Displays the status of the last CFS merger for Smart Call Home. |
| show callhome pending | Displays the Smart Call Home configuration changes in the pending CFS database. |
| show callhome pending-diff | Displays the differences between the pending and running Smart Call Home configuration. |

| Command | Purpose |
|---|--|
| show callhome session-status | Displays the status of the last CFS commit or abort operation. |
| show callhomestatus | Displays the CFS distribution state (enabled or disabled) for Smart Call Home. |
| show callhome transport | Displays the transport-related configuration for Smart Call Home. |
| show callhome user-def-cmds | Displays CLI commands added to any alert groups. |
| show running-config callhome [all] | Displays the running configuration for Smart Call Home. |
| show startup-config callhome | Displays the startup configuration for Smart Call Home. |
| show tech-support callhome | Displays the technical support output for Smart Call Home. |

Configuration Examples for Smart Call Home

This example shows how to create a destination profile called Noc101, associate the Configuration alert group to that profile, configure contact and email information, and specify the VRF used to send Smart Call Home messages over HTTP:

```
configure terminal
snmp-server contact person@company.com
callhome
distribute
email-contact admin@Mycompany.com
phone-contact +1-800-123-4567
streetaddress 123 Anystreet st. Anytown,AnyWhere
destination-profile Noc101 format full-txt
destination-profile full-text-destination email-addr person@company.com
destination-profile full-text-destination message-level 5
destination-profile Noc101 alert-group Configuration
alert-group Configuration user-def-cmd show ip route
transport email mail-server 192.0.2.10 priority 1
transport http use-vrf Blue
enable
commit
```

This example shows how to configure multiple SMTP servers for Smart Call Home messages:

```
configure terminal
callhome
transport email mail-server 192.0.2.10 priority 4
transport email mail-server 172.21.34.193
transport email smtp-server 10.1.1.174
transport email mail-server 64.72.101.213 priority 60
transport email from person@company.com
```



```
transport email reply-to person@company.com
commit
```

Based on the configuration above, the SMTP servers would be tried in this order:

10.1.1.174 (priority 0)

192.0.2.10 (priority 4)

172.21.34.193 (priority 50, which is the default)

64.72.101.213 (priority 60)



Note The **transport email smtp-server** command has a priority of 0, which is the highest. The server specified by this command is tried first followed by the servers specified by the **transport email mail-server** commands in order of priority.

This example shows how to configure Smart Call Home to send HTTP messages through an HTTP proxy server:

```
configure terminal
callhome
transport http proxy server 10.10.10.1 port 4
transport http proxy enable
commit
```

Additional References

Event Triggers

The following table lists the event triggers and their Smart Call Home message severity levels.

| Alert Group | Event Name | Description | Smart Call Home Severity Level |
|---------------|-------------------------|--|--------------------------------|
| Configuration | PERIODIC_CONFIGURATION | Periodic configuration update message. | 2 |
| Diagnostic | DIAGNOSTIC_MAJOR_ALERT | GOLD generated a major alert. | 7 |
| | DIAGNOSTIC_MINOR_ALERT | GOLD generated a minor alert. | 4 |
| | DIAGNOSTIC_NORMAL_ALERT | Smart Call Home generated a normal diagnostic alert. | 2 |

| Alert Group | Event Name | Description | Smart Call Home Severity Level |
|------------------------------------|-------------------------|---|--------------------------------|
| Environmental and CISCO_TAC | FAN_FAILURE | Cooling fan has failed. | 5 |
| | POWER_SUPPLY_ALERT | Power supply warning has occurred. | 6 |
| | POWER_SUPPLY_FAILURE | Power supply has failed. | 6 |
| | POWER_SUPPLY_SHUTDOWN | Power supply has shut down. | 6 |
| | TEMPERATURE_ALARM | Thermal sensor going bad. | 6 |
| | TEMPERATURE_MAJOR_ALARM | Thermal sensor indicates temperature has reached operating major threshold. | 6 |
| | TEMPERATURE_MINOR_ALARM | Thermal sensor indicates temperature has reached operating minor threshold. | 4 |
| Inventory and CISCO_TAC | COLD_BOOT | Switch is powered up and reset to a cold boot sequence. | 2 |
| | HARDWARE_INSERTION | New piece of hardware has been inserted into the chassis. | 2 |
| | HARDWARE_REMOVAL | Hardware has been removed from the chassis. | 2 |
| | PERIODIC_INVENTORY | Periodic inventory message has been generated. | 2 |
| License | LICENSE_VIOLATION | Feature in use is not licensed and is turned off after grace period expiration. | 6 |
| Line module Hardware and CISCO_TAC | LINEmodule_FAILURE | Module operation has failed. | 7 |
| Supervisor Hardware and CISCO_TAC | SUP_FAILURE | Supervisor module operation has failed. | 7 |
| Syslog-group-port | PORT_FAILURE | syslog message that corresponds to the port facility has been generated. | 6 |
| | SYSLOG_ALERT | syslog alert message has been generated. | 5 |

| Alert Group | Event Name | Description | Smart Call Home Severity Level |
|----------------------|------------------------|---|--------------------------------|
| System and CISCO_TAC | SW_CRASH | Software process has failed with a stateless restart, indicating an interruption of a service. Messages are sent for process crashes on supervisor modules. | 5 |
| | SW_SYSTEM_INCONSISTENT | Inconsistency has been detected in software or file system. | 5 |
| Test and CISCO_TAC | TEST | User generated test has occurred. | 2 |

Message Formats

Smart Call Home supports the following message formats:

Short Text Message Format

The following table describes the short text formatting option for all message types.

| Data Item | Description |
|-------------------------|--|
| Device identification | Configured device name |
| Date/time stamp | Time stamp of the triggering event |
| Error isolation message | Plain English description of triggering event |
| Alarm urgency level | Error level such as that applied to system message |

Common Event Message Fields

The following table describes the first set of common event message fields for full text or XML messages.

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|--------------------------------|---|--------------------|
| Timestamp | Date and time stamp of event in ISO time notation: YYYY-MM-DD HH:MM:SS GMT+HH:MM. | /aml/header/time |
| Message name | Name of message. | /aml/header/name |
| Message type | Name of message type, such as reactive or proactive. | /aml/header/type |
| Message group | Name of alert group, such as syslog. | /aml/header/group |
| Severity level | Severity level of message. | /aml/header/level |

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|--------------------------------|--|--------------------------|
| Source ID | Product type for routing, such as the Catalyst 6500 series switch. | /aml/header/source |
| Device ID | <p>Unique device identifier (UDI) for the end device that generated the message. This field should be empty if the message is nonspecific to a device. The format is <i>type@Sid@serial</i>.</p> <ul style="list-style-type: none"> • <i>type</i> is the product model number from the backplane IDPROM. • @ is a separator character. • <i>Sid</i> is C, identifying the serial ID as a chassis serial number. • <i>serial</i> is the number identified by the Sid field. <p>An example is WS-C6509@C@12345678.</p> | /aml/ header/deviceId |
| Customer ID | Optional user-configurable field used for contract information or other ID by any support service. | /aml/ header/customerID |
| Contract ID | Optional user-configurable field used for contract information or other ID by any support service. | /aml/ header /contractId |
| Site ID | Optional user-configurable field used for Cisco-supplied site ID or other data meaningful to alternate support service. | /aml/ header/siteId |
| Server ID | <p>If the message is generated from the device, this ID is the unique device identifier (UDI) of the device. The format is <i>type@Sid@serial</i>.</p> <ul style="list-style-type: none"> • <i>type</i> is the product model number from the backplane IDPROM. • @ is a separator character. • <i>Sid</i> is C, identifying the serial ID as a chassis serial number. • <i>serial</i> is the number identified by the Sid field. <p>An example is WS-C6509@C@12345678.</p> | /aml/header/serverId |

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|--------------------------------|---|-------------------------------------|
| Message description | Short text that describes the error. | /aml/body/msgDesc |
| Device name | Node that experienced the event (hostname of the device). | /aml/body/sysName |
| Contact name | Name of person to contact for issues associated with the node that experienced the event. | /aml/body/sysContact |
| Contact email | Email address of person identified as the contact for this unit. | /aml/body/sysContactEmail |
| Contact phone number | Phone number of the person identified as the contact for this unit. | /aml/body/sysContactPhone Number |
| Street address | Optional field that contains the street address for RMA part shipments associated with this unit. | /aml/body/sysStreetAddress |
| Model name | Model name of the device (the specific model as part of a product family name). | /aml/body/chassis/name |
| Serial number | Chassis serial number of the unit. | /aml/body/chassis/serialNo |
| Chassis part number | Top assembly number of the chassis. | /aml/body/chassis/partNo |

Alert Group Message Fields

The following table describes the fields specific to alert group messages for full text and XML. These fields may be repeated if multiple CLI commands are executed for an alert group.

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|--------------------------------|---|------------------------------------|
| Command output name | Exact name of the issued CLI command. | /aml/attachments/attachment/name |
| Attachment type | Specific command output. | /aml/attachments/attachment/type |
| MIME type | Either plain text or encoding type. | /aml/attachments/attachment/mime |
| Command output text | Output of command automatically executed. | /aml/attachments/attachment/atdata |

Fields for Reactive and Proactive Event Messages

The following table describes the reactive and proactive event message format for full text or XML messages.

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|--------------------------------|----------------------------------|-----------------------------|
| Chassis hardware version | Hardware version of chassis. | /aml/body/chassis/hwVersion |

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|------------------------------------|--|-----------------------------|
| Supervisor module software version | Top-level software version. | /aml/body/chassis/swVersion |
| Affected FRU name | Name of the affected FRU that is generating the event message. | /aml/body/fru/name |
| Affected FRU serial number | Serial number of the affected FRU. | /aml/body/fru/serialNo |
| Affected FRU part number | Part number of the affected FRU. | /aml/body/fru/partNo |
| FRU slot | Slot number of the FRU that is generating the event message. | /aml/body/fru/slot |
| FRU hardware version | Hardware version of the affected FRU. | /aml/body/fru/hwVersion |
| FRU software version | Software version(s) that is running on the affected FRU. | /aml/body/fru/swVersion |

Fields for Inventory Event Messages

The following table describes the inventory event message format for full text or XML messages.

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|------------------------------------|--|-----------------------------|
| Chassis hardware version | Hardware version of the chassis. | /aml/body/chassis/hwVersion |
| Supervisor module software version | Top-level software version. | /aml/body/chassis/swVersion |
| FRU name | Name of the affected FRU that is generating the event message. | /aml/body/fru/name |
| FRU s/n | Serial number of the FRU. | /aml/body/fru/serialNo |
| FRU part number | Part number of the FRU. | /aml/body/fru/partNo |
| FRU slot | Slot number of the FRU. | /aml/body/fru/slot |
| FRU hardware version | Hardware version of the FRU. | /aml/body/fru/hwVersion |
| FRU software version | Software version(s) that is running on the FRU. | /aml/body/fru/swVersion |

Fields for User-Generated Test Messages

The following table describes the user-generated test message format for full text or XML.

| Data Item (Plain Text and XML) | Description (Plain Text and XML) | XML Tag (XML Only) |
|--------------------------------|--|--------------------------------|
| Process ID | Unique process ID. | /aml/body/process/id |
| Process state | State of process (for example, running or halted). | /aml/body/process/processState |
| Process exception | Exception or reason code. | /aml/body/process/exception |

Sample Syslog Alert Notification in Full-Text Format

This sample shows the full-text format for a syslog port alert-group notification:

```
Severity Level:5
Series:Nexus7000
Switch Priority:0
Device Id:N7K-C7010@C@TXX12345678
Server Id:N7K-C7010@C@TXX12345678
Time of Event:2008-01-17 16:31:33 GMT+0000 Message Name:
Message Type:syslog
System Name:dc3-test
Contact Name:Jay Tester
Contact Email:contact@example.com
Contact Phone:+91-80-1234-5678
Street Address:#1 Any Street
Event Description:SYSLOG_ALERT 2008 Jan 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error
(0x20) while
communicating with component MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT:
Ethernet3/1)

syslog_facility:ETHPORT
start chassis information:
Affected Chassis:N7K-C7010
Affected Chassis Serial Number:TXX12345678 Affected Chassis Hardware Version:0.405 Affected
Chassis Software
Version:4.1(1) Affected Chassis Part No:73-10900-04 end chassis information:
start attachment
name:show logging logfile | tail -n 200
type:text
data:
2008 Jan 17 10:57:51 dc3-test %SYSLOG-1-SYSTEM_MSG : Logging logfile (messages) cleared by
user
2008 Jan 17 10:57:53 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 17 10:58:35 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 17 10:59:00 dc3-test %DAEMON-3-SYSTEM_MSG: error: setsockopt IP_TOS 16: Invalid
argument: - sshd[14484]
2008 Jan 17 10:59:05 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 17 12:11:18 dc3-test %SYSMGR-STANDBY-5-SUBPROC_TERMINATED: "System Manager (gsync
controller)" (PID 12000)
has finished with error code SYSMGR_EXITCODE_GSYNCFAILED_NONFATAL (12).
2008 Jan 17 16:28:03 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 17 16:28:44 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
not generated by system
for eltm(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:28:44 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 3504) hasn't
caught signal 9 (no core).
2008 Jan 17 16:29:08 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
not generated by system
for eltm(0). WCOREDUMP(9) returned zero.
2008 Jan 17 16:29:08 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 23210) hasn't
caught signal 9 (no core).
2008 Jan 17 16:29:17 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
not generated by system
for eltm(0). WCOREDUMP(9) returned zero.
2008 Jan 17 16:29:17 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 23294) hasn't
caught signal 9 (no core).
2008 Jan 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_PRE_START: This supervisor is becoming
active (pre-start phase).
2008 Jan 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_START: This supervisor is becoming
```

```

active.
2008 Jan 17 16:29:26 dc3-test %USER-3-SYSTEM_MSG: crdcfg_get_srvinfo: mts_send failed -
device_test
2008 Jan 17 16:29:27 dc3-test %NETSTACK-3-IP_UNK_MSG_MAJOR: netstack [4336] Unrecognized
message from MRIB. Major
type 1807
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 1
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 2
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 3
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 4
2008 Jan 17 16:29:28 dc3-test %SYSMGR-2-SWITCHOVER_OVER: Switchover completed.
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 10 - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:ipv6 only defined - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:bindv6 only defined - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2008 Jan 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2008 Jan 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19072]
2008 Jan 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19072]
2008 Jan 17 16:29:31 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19073]
2008 Jan 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19079]
2008 Jan 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19079]
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 1
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 2
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 3
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 4
2008 Jan 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19105]
2008 Jan 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19105]
2008 Jan 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 2 present but all
AC inputs are not
connected, ac-redundancy might be affected
2008 Jan 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 3 present but all
AC inputs are not
connected, ac-redundancy might be affected
2008 Jan 17 16:29:38 dc3-test %CALLHOME-2-EVENT: SUP_FAILURE
2008 Jan 17 16:29:46 dc3-test vsh[19166]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2008 Jan 17 16:30:24 dc3-test vsh[23810]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2008 Jan 17 16:30:24 dc3-test vsh[23803]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2008 Jan 17 16:30:24 dc3-test vsh[23818]: CLIC-3-FAILED_EXEC: Can not exec command <more>
return code <14>
2008 Jan 17 16:30:47 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
not generated by
system for eltm(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:30:47 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 4820) hasn't
caught signal 9
(no core).
2008 Jan 17 16:31:02 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
not generated by
system for eltm(0). WCOREDUMP(9) returned zero .

```



```

2008 Jan 17 16:31:02 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltn" (PID 24239) hasn't
  caught signal 9
  (no core).
2008 Jan 17 16:31:14 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
  not generated by
  system for eltn(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:31:14 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltn" (PID 24401) hasn't
  caught signal 9
  (no core).
2008 Jan 17 16:31:23 dc3-test %CALLHOME-2-EVENT: SW_CRASH alert for service: eltn
2008 Jan 17 16:31:23 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
  not generated by
  system for eltn(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:31:23 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltn" (PID 24407) hasn't
  caught signal 9
  (no core).
2008 Jan 17 16:31:24 dc3-test vsh[24532]: CLIC-3-FAILED_EXEC: Can not exec command <more>
  return code <14>
2008 Jan 17 16:31:24 dc3-test vsh[24548]: CLIC-3-FAILED_EXEC: Can not exec command <more>
  return code <14>
2008 Jan 17 16:31:24 dc3-test vsh[24535]: CLIC-3-FAILED_EXEC: Can not exec command <more>
  return code <14>
2008 Jan 17 16:31:33 dc3-test %NETSTACK-3-INTERNAL_ERROR: netstack [4336] (null)
2008 Jan 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating
  with component MTS_SAP_ELTM
  opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT: Ethernet3/1) end attachment start
  attachment
  name:show vdc membership
  type:text
  data:
  vdc_id: 1 vdc_name: dc3-test interfaces:
  Ethernet3/1 Ethernet3/2 Ethernet3/3
  Ethernet3/4 Ethernet3/5 Ethernet3/6
  Ethernet3/7 Ethernet3/8 Ethernet3/9
  Ethernet3/10 Ethernet3/11 Ethernet3/12
  Ethernet3/13 Ethernet3/14 Ethernet3/15
  Ethernet3/16 Ethernet3/17 Ethernet3/18
  Ethernet3/19 Ethernet3/20 Ethernet3/21
  Ethernet3/22 Ethernet3/23 Ethernet3/24
  Ethernet3/25 Ethernet3/26 Ethernet3/27
  Ethernet3/28 Ethernet3/29 Ethernet3/30
  Ethernet3/31 Ethernet3/32 Ethernet3/33
  Ethernet3/34 Ethernet3/35 Ethernet3/36
  Ethernet3/37 Ethernet3/38 Ethernet3/39
  Ethernet3/40 Ethernet3/41 Ethernet3/42
  Ethernet3/43 Ethernet3/44 Ethernet3/45
  Ethernet3/46 Ethernet3/47 Ethernet3/48
  vdc_id: 2 vdc_name: dc3-aaa interfaces:
  vdc_id: 3 vdc_name: dc3-rbac interfaces:
  vdc_id: 4 vdc_name: dc3-call interfaces:
  end attachment
  start attachment
  name:show vdc current-vdc
  type:text
  data:
  Current vdc is 1 - dc3-test
  end attachment
  start attachment
  name:show license usage
  type:text
  data:
  Feature Ins Lic Status Expiry Date Comments
  Count
  -----

```

```

LAN_ADVANCED_SERVICES_PKG Yes - In use Never -
LAN_ENTERPRISE_SERVICES_PKG Yes - Unused Never -
-----
end attachment

```

Sample Syslog Alert Notification in XML Format

This sample shows the XML format for a syslog port alert-group notification:

```

<?xml version="1.0" encoding="UTF-8" ?>
<soap-env:Envelope xmlns:soap-env="http://www.w3.org/2003/05/soap-envelope">
  <soap-env:Header>
    <aml-session:Session xmlns:aml-session="http://www.cisco.com/2004/01/aml-session"
      soap-env:mustUnderstand="true"
      soap-env:role="http://www.w3.org/2003/05/soap-envelope/role/next">
      <aml-session:To>http://tools.cisco.com/neddce/services/DDCEService</aml-session:To>
      <aml-session:Path>
      <aml-session:Via>http://www.cisco.com/appliance/uri</aml-session:Via>
      </aml-session:Path>
      <aml-session:From>http://www.cisco.com/appliance/uri</aml-session:From>
      <aml-session:MessageId>1004:TXX12345678:478F82E6</aml-session:MessageId>
    </aml-session:Session>
  </soap-env:Header>
  <soap-env:Body>
    <aml-block:Block xmlns:aml-block="http://www.cisco.com/2004/01/aml-block">
      <aml-block:Header>
        <aml-block:Type>http://www.cisco.com/2005/05/callhome/syslog</aml-block:Type>
        <aml-block:CreationDate>2008-01-17 16:31:33 GMT+0000</aml-block:CreationDate>
        <aml-block:Builder>
          <aml-block:Name>DC3</aml-block:Name>
          <aml-block:Version>4.1</aml-block:Version>
        </aml-block:Builder>
        <aml-block:BlockGroup>
          <aml-block:Block:GroupId>1005:TXX12345678:478F82E6</aml-block:Block:GroupId>
          <aml-block:Block:Number>0</aml-block:Block:Number>
          <aml-block:Block:IsLast>true</aml-block:Block:IsLast>
          <aml-block:Block:IsPrimary>true</aml-block:Block:IsPrimary>
          <aml-block:Block:WaitForPrimary>>false</aml-block:Block:WaitForPrimary>
        </aml-block:BlockGroup>
        <aml-block:Block:Severity>5</aml-block:Block:Severity>
      </aml-block:Header>
      <aml-block:Content>
        <ch:CallHome xmlns:ch="http://www.cisco.com/2005/05/callhome" version="1.0">
          <ch:EventTime>2008-01-17 16:31:33 GMT+0000</ch:EventTime> <ch:MessageDescription>SYSLOG_ALERT
            2008 Jan 17 16:31:33
            dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating with component MTS_SAP_ELTM
            opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP
            (for:RID_PORT: Ethernet3/1) </ch:MessageDescription> <ch:Event> <ch:Type>syslog</ch:Type>
          <ch:SubType></ch:SubType>
          <ch:Brand>Cisco</ch:Brand> <ch:Series>Nexus7000</ch:Series> </ch:Event> <ch:CustomerData>
          <ch:UserData>
            <ch:Email>contact@example.com</ch:Email>
          </ch:UserData>
          <ch:ContractData>
            <ch:DeviceId>N7K-C7010@C@TXX12345678</ch:DeviceId>
          </ch:ContractData>
          <ch:SystemInfo>
            <ch:Name>dc3-test</ch:Name>
            <ch:Contact>Jay Tester</ch:Contact> <ch:ContactEmail>contact@example.com</ch:ContactEmail>
            <ch:ContactPhoneNumber>+91-80-1234-5678</ch:ContactPhoneNumber>
            <ch:StreetAddress>#1, Any Street</ch:StreetAddress> </ch:SystemInfo> </ch:CustomerData>
          <ch:Device>

```

```

<rme:Chassis xmlns:rme="http://www.cisco.com/rme/4.1">
<rme:Model>N7K-C7010</rme:Model>
<rme:HardwareVersion>0.405</rme:HardwareVersion>
<rme:SerialNumber>TXXL12345678</rme:SerialNumber>
</rme:Chassis>
</ch:Device>
</ch:CallHome>
</aml-block:Content>
<aml-block:Attachments>
<aml-block:Attachment type="inline">
<aml-block:Name>show logging logfile | tail -n 200</aml-block:Name> <aml-block:Data
encoding="plain">
<![CDATA[2008 Jan 17 10:57:51 dc3-test %SYSLOG-1-SYSTEM_MSG : Logging logfile (messages)
cleared by user
2008 Jan 17 10:57:53 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 17 10:58:35 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 17 10:59:00 dc3-test %DAEMON-3-SYSTEM_MSG: error: setsockopt IP_TOS 16: Invalid
argument: - sshd[14484]
2008 Jan 17 10:59:05 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 17 12:11:18 dc3-test %SYSMGR-STANDBY-5-SUBPROC_TERMINATED: \"System Manager (gsync
controller)\"
(PID 12000) has finished with error code SYSMGR_EXITCODE_GSYNCFALIED_NONFATAL (12).
2008 Jan 17 16:28:03 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 17 16:28:44 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
not generated by system
for eltm(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:28:44 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 3504)
hasn&apos;t caught signal 9
(no core).
2008 Jan 17 16:29:08 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
not generated by system
for eltm(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:29:08 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 23210)
hasn&apos;t caught signal 9
(no core).
2008 Jan 17 16:29:17 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
not generated by system
for eltm(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:29:17 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 23294)
hasn&apos;t caught signal 9
(no core).
2008 Jan 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_PRE_START: This supervisor is becoming
active (pre-start phase).
2008 Jan 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_START: This supervisor is becoming
active.
2008 Jan 17 16:29:26 dc3-test %USER-3-SYSTEM_MSG: crdcfg_get_srvinfo: mts_send failed -
device_test
2008 Jan 17 16:29:27 dc3-test %NETSTACK-3-IP_UNK_MSG_MAJOR: netstack [4336] Unrecognized
message from MRIB.
Major type 1807
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 1
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 2
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 3
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 4
2008 Jan 17 16:29:28 dc3-test %SYSMGR-2-SWITCHOVER_OVER: Switchover completed.
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 10 - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:ipv6 only defined - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:bindv6 only defined - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
]]>

```

```

2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2008 Jan 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2008 Jan 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19072]
2008 Jan 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19072]
2008 Jan 17 16:29:31 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19073]
2008 Jan 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19079]
2008 Jan 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19079]
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 1
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 2
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 3
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 4
2008 Jan 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19105]
2008 Jan 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19105]
2008 Jan 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 2 present but all
AC inputs are not
connected, ac-redundancy might be affected
2008 Jan 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 3 present but all
AC inputs are not
connected, ac-redundancy might be affected
2008 Jan 17 16:29:38 dc3-test %CALLHOME-2-EVENT: SUP_FAILURE
2008 Jan 17 16:29:46 dc3-test vsh[19166]: CLIC-3-FAILED_EXEC: Can not exec command
<more>; return code <14>;
2008 Jan 17 16:30:24 dc3-test vsh[23810]: CLIC-3-FAILED_EXEC: Can not exec command
<more>; return code <14>;
2008 Jan 17 16:30:24 dc3-test vsh[23803]: CLIC-3-FAILED_EXEC: Can not exec command
<more>; return code <14>;
2008 Jan 17 16:30:24 dc3-test vsh[23818]: CLIC-3-FAILED_EXEC: Can not exec command
<more>; return code <14>;
2008 Jan 17 16:30:47 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
not generated by system
for eltm(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:30:47 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 4820)
hasn't caught signal 9
(no core).
2008 Jan 17 16:31:02 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
not generated by system
for eltm(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:31:02 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24239)
hasn't caught signal 9
(no core).
2008 Jan 17 16:31:14 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
not generated by system
for eltm(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:31:14 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24401)
hasn't caught signal 9
(no core).
2008 Jan 17 16:31:23 dc3-test %CALLHOME-2-EVENT: SW_CRASH alert for service: eltm
2008 Jan 17 16:31:23 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
not generated by system
for eltm(0). WCOREDUMP(9) returned zero .
2008 Jan 17 16:31:23 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24407)
hasn't caught signal 9
(no core).

```

```

2008 Jan 17 16:31:24 dc3-test vsh[24532]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2008 Jan 17 16:31:24 dc3-test vsh[24548]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2008 Jan 17 16:31:24 dc3-test vsh[24535]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2008 Jan 17 16:31:33 dc3-test %NETSTACK-3-INTERNAL_ERROR: netstack [4336] (null)
2008 Jan 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating
with component
MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT: Ethernet3/1) ]]>
</aml-block:Data>
</aml-block:Attachment> <aml-block:Attachment type="inline"> <aml-block:Name>show vdc
membership</aml-block:Name>
<aml-block:Data encoding="plain"> <![CDATA[
vdc_id: 1 vdc_name: dc3-test interfaces:
Ethernet3/1 Ethernet3/2 Ethernet3/3
Ethernet3/4 Ethernet3/5 Ethernet3/6
Ethernet3/7 Ethernet3/8 Ethernet3/9
Ethernet3/10 Ethernet3/11 Ethernet3/12
Ethernet3/13 Ethernet3/14 Ethernet3/15
Ethernet3/16 Ethernet3/17 Ethernet3/18
Ethernet3/19 Ethernet3/20 Ethernet3/21
Ethernet3/22 Ethernet3/23 Ethernet3/24
Ethernet3/25 Ethernet3/26 Ethernet3/27
Ethernet3/28 Ethernet3/29 Ethernet3/30
Ethernet3/31 Ethernet3/32 Ethernet3/33
Ethernet3/34 Ethernet3/35 Ethernet3/36
Ethernet3/37 Ethernet3/38 Ethernet3/39
Ethernet3/40 Ethernet3/41 Ethernet3/42
Ethernet3/43 Ethernet3/44 Ethernet3/45
Ethernet3/46 Ethernet3/47 Ethernet3/48
vdc_id: 2 vdc_name: dc3-aaa interfaces:
vdc_id: 3 vdc_name: dc3-rbac interfaces:
vdc_id: 4 vdc_name: dc3-call interfaces:

]]>
</aml-block:Data>
</aml-block:Attachment>
<aml-block:Attachment type="inline">
<aml-block:Name>show vdc current-vdc</aml-block:Name> <aml-block:Data encoding="plain">
<![CDATA[Current vdc
is 1 - dc3-test ]]> </aml-block:Data> </aml-block:Attachment> <aml-block:Attachment
type="inline">
<aml-block:Name>show license usage</aml-block:Name> <aml-block:Data encoding="plain">
<![CDATA[Feature Ins Lic Status Expiry Date Comments
Count
-----
LAN_ADVANCED_SERVICES_PKG Yes - In use Never -
LAN_ENTERPRISE_SERVICES_PKG Yes - Unused Never -
-----
]]>
</aml-block:Data>
</aml-block:Attachment>
</aml-block:Attachments>
</aml-block:Block>
</soap-env:Body>
</soap-env:Envelope>

```

Additional References

Related Documents

| Related Topic | Document Title |
|-------------------------|---|
| Smart Call CLI commands | <i>Cisco Nexus 7000 Series NX-OS System Management Command Reference</i> |
| VDCs and VRFs | <i>Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide</i> |

MIBs

| MIBs | MIBs Link |
|---------------------------------|--|
| MIBs related to Smart Call Home | To locate and download supported MIBs, go to the following URL: ftp://ftp.cisco.com/pub/mibs/supportlists/nexus7000/Nexus7000MIBSupportList.html |

Feature History for Smart Call Home

Your software release might not support all the features in this document. For the latest caveats and feature information, see the Bug Search Tool at <https://tools.cisco.com/bugsearch/> and the release notes for your software release.

Table 3: Feature History for Smart Call Home

| Feature Name | Releases | Feature Information |
|--|----------|---|
| HTTP proxy server | 5.2(1) | Added the ability to send HTTP messages through an HTTP proxy server. |
| SMTP server configuration | 5.0(2) | Added the ability to configure multiple SMTP servers. |
| VRF support for HTTP transport of Smart Call Home messages | 5.0(2) | VRFs can be used to send e-mail and other Smart Call Home messages over HTTP. |
| Crash notifications | 4.0(1) | Messages are sent for process crashes on line cards. |
| Destination profile configuration | 4.1(3) | The commands destination-profile http and destination-profile transport-method cannot be distributed. |