

Programmability

- action export-to-telemetry, on page 4
- app-default-gateway, on page 5
- app-hosting, on page 6
- app-hosting appid, on page 8
- app-hosting data appid, on page 9
- app-hosting settings appid, on page 10
- app-resource docker, on page 11
- app-resource profile, on page 12
- app-vnic gateway, on page 13
- app-vnic AppGigabitEthernet, on page 14
- app-vnic management , on page 15
- boot ipxe, on page 16
- boot manual, on page 17
- boot system, on page 18
- ca-trustpoint, on page 19
- clear configuration lock, on page 20
- clear netconf-yang session, on page 21
- clear telemetry ietf subscription, on page 22
- cpu (App Hosting), on page 24
- dampening-period, on page 26
- debug netconf-yang, on page 28
- debug netconf-yang diagnostics, on page 30
- debug restconf, on page 31
- default boot, on page 33
- dig, on page 34
- enable (App Hosting), on page 36
- encoding, on page 37
- filter , on page 38
- gnxi, on page 39
- gnxi access-list, on page 42
- gnxi vrf, on page 43
- guest-interface (App Hosting) , on page 44
- guest-ipaddress (App Hosting), on page 45

- guest-ipv6address, on page 47
- guestshell, on page 49
- guestshell portforwarding, on page 50
- host, on page 52
- id-trustpoint, on page 53
- install, on page 54
- iox, on page 59
- keepalive (grpc-profile), on page 60
- mac-forwarding (App Hosting), on page 61
- memory (App Hosting) , on page 62
- mirroring, on page 63
- mlog, on page 64
- monitor log profile netconf-yang, on page 65
- monitor log profile restconf, on page 68
- multicast (App Hosting), on page 71
- name-server (App Hosting), on page 73
- net-debug, on page 74
- net-dhcp, on page 76
- net-show , on page 77
- net-tcp-bufs, on page 78
- net-tcp-mss, on page 79
- net6-dhcp, on page 80
- net6-show, on page 81
- netconf detailed-error, on page 82
- netconf legacy, on page 84
- netconf-yang feature candidate-datasource, on page 85
- netconf-yang feature side-effect-sync, on page 87
- netconf-yang ssh, on page 88
- netconf-yang ssh local-vrf guestshell, on page 90
- netconf-yang ssh port disable, on page 91
- netconf-yang ssh server algorithm encryption , on page 92
- netconf-yang ssh server algorithm hostkey , on page 94
- netconf-yang ssh server algorithm kex , on page 95
- netconf-yang ssh server algorithm mac , on page 96
- netconf-yang ssh trustpoint, on page 97
- persist-disk (App Hosting), on page 98
- ping, on page 99
- ping4, on page 100
- ping6, on page 101
- prepend-pkg-opts, on page 102
- protocol, on page 103
- receiver, on page 104
- receiver name, on page 106
- receiver-type protocol, on page 107
- resource profile, on page 108
- restconf access-list, on page 110

- request platform software yang-management nacm, on page 112
- run-opts, on page 114
- show app-hosting, on page 115
- show controller ethernet-controller AppGigabitEthernet, on page 117
- show gnxi server vrf, on page 119
- show gnxi state, on page 120
- show install, on page 123
- show iox-service, on page 126
- show log profile netconf-yang, on page 129
- show log profile restconf, on page 132
- show netconf, on page 135
- show netconf-yang , on page 138
- show netconf-yang diagnostics, on page 141
- show netconf-yang ssh server, on page 143
- show netconf-yang ssh trustpoint, on page 145
- show netconf-yang status, on page 146
- show platform software yang-management process, on page 147
- show platform software yang-management process state, on page 150
- show tech-support gnxi, on page 152
- show telemetry connection , on page 158
- show telemetry ietf subscription, on page 161
- show telemetry internal connection, on page 164
- show telemetry internal diagnostics, on page 166
- show telemetry internal sensor, on page 170
- show telemetry internal subscription , on page 172
- show telemetry receiver, on page 173
- source-address (telemetry), on page 175
- source-vrf (telemetry), on page 176
- start (App Hosting), on page 177
- stream, on page 178
- telemetry ietf subscription, on page 179
- telemetry protocol grpc profile, on page 180
- telemetry receiver protocol , on page 181
- update-policy, on page 182
- vcpu (App Hosting), on page 183
- vlan (App Hosting), on page 184
- vnic gateway, on page 185
- vnic management, on page 186
- yang-interfaces aaa, on page 187
- yang-interfaces feature deprecated disable, on page 189

action export-to-telemetry

To export Embedded Event Manager (EEM) variables to telemetry, use the **action export-to-telemetry** command in applet configuration mode. To disable the action of exporting EEM variables to telemetry, use the **no** form of this command.

action *label* **export-to-telemetry** [*EEM-variable*]
no action *label*

Syntax Description

label	Unique identifier that can be any string value. Actions are sorted and run in ascending alphanumeric key sequence using the label as the sort key. If the string contains embedded blanks, enclose it in double quotation marks.
EEM-variable	(Optional) User-defined EEM variable.

Command Default

Command Modes

Applet configuration (config-applet)

Command History

Release	Modification
Cisco IOS XE Amsterdam 17.1.1	This command was introduced.

Usage Guidelines

The EEM Event Publish capability is part of the Cisco-IOS-XE-ios-events-oper.YANG module for on-change telemetry notifications.

This command exports the event-specific data of the EEM policy using YANG notification to an external telemetry collector. The variables are exported in the *key:value* pair format for the external telemetry collector to use. For example, if the EEM applet script detects a certain percentage of packet loss on an interface, a custom message can be added to notify about the loss.

Example

This example shows how to export EEM variables to telemetry.

```
Device> enable
Device# configure terminal
Device(config)# event manager applet one
Device(config-applet)# action 1.0 export-to-telemetry
Device(config-applet)#
```

Related Commands

Command	Description
event manager applet	Registers an event applet with EEM and enters applet configuration mode.

app-default-gateway

To set the default gateway for an application, use the **app-default-gateway** command in application hosting configuration mode. To remove the default gatway, use the **no** form of this command.

app-default-gateway *ip-address* **guest-interface** *network-interface-number*
no app-default-gateway [*ip-address* **guest-interface** *network-interface-number*]

Syntax Description

ip-address	IP address of the default gateway.
guest-interface *network-interface-number*	Configures the guest interface. The *network-interface-number* maps to the container Ethernet number.

Command Default

The default gateway is not configured.

Command Modes

Application hosting configuration (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Example

The following example shows how to set the default gateway for the application:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-default-gateway 10.3.3.31 guest-interface 1
Device(config-app-hosting)#
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.

app-hosting

To initialize application hosting, use the **app-hosting** command in privileged EXEC mode.

app-hosting { {**install appid** *application-name* **package** *package-location* } | **activate** | **start** | **stop** | **deactivate** | **uninstall**} **appid** *application-name*

Syntax Description

install	Installs the application.
appid *application-name*	Installs the specified application.
package *package-location*	Installs the application package from the specified location.
activate	Activates the application package.
start	Starts the application by activating the start-up scripts.
stop	Stops the application.
deactivate	Deactivates the application.
uninstall	Uninstalls the application.

Command Default

Application hosting is not initialized.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

For application hosting to work, IOx services must be configured using the **iox** command.

Copy the IOx application to the local device storage medium using the Cisco IOS **copy** command, and enable the **app-hosting install** command to enable application hosting.

Applications can be installed from local storage locations such as, flash, bootflash, usbflash0, usbflash1, and harddisk.

The **activate** keyword validates all application resource requests, and if all requested resources are available, the application is activated; if not, the activation fails.

The **start** keyword executes the application's start-up script, and the **stop** keyword is equivalent to an application shutdown.

While uninstalling the application, all packages and images stored in the system are removed. All changes and updates to the application are also removed.

Example

The following example shows how to install a third-party application:

```
Device# app-hosting install appid iox_app package flash:my_iox_app.tar
```

Related Commands

Command	Description
iox	Configure IOx services.

app-hosting appid

To configure an application, and to enter application hosting configuration mode, use the **app-hosting appid** command in global configuration mode. To remove the application, use the **no** form of this command.

app-hosting appid *application-name*
no app-hosting appid *application-name*

Syntax Description

application-name	Application name.

Command Default

No application is configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

The *application name* argument can be up to 32 alphanumeric characters.

You can update the application hosting configuration, after configuring this command.

Example

The following example shows how to configure an application:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device (config-app-hosting)#
```

app-hosting data appid

To transfer application data contents into an application's persistent data mount, use the **app-hosting data appid** command in privileged EXEC mode.

app-hosting data appid *application-name* {**copy** *source-file-path destination-file-path* | **delete** *file-path* }

Syntax Description

application-name	Name of the application.
copy	Copies a file to destination file or directory under the application's shared data.
source-file-path	The folder where the source file resides.
destination-file-path	The folder where the file is to be copied.
delete *file-path*	Deletes a specified file or directory from the application's shared data.

Command Default

Application data is not transferred.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

Based on the specified file path, the **delete** keyword can delete either the file or the entire directory.

Example

The following example shows how to copy an application:

```
Device# app-hosting data appid app docker1 copy bootflash:IOXN.log cfg/IOXN.log

Successfully copied file /flash/IOXN.log to docker1 as cfg/IOXN.log
```

The following example shows how to delete an application:

```
Device# app-hosting data appid app1 delete bootflash:n2os_ids app-data-dir cfg/n2os_ids
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.

app-hosting settings appid

To enable the settings of an application, use the **app-hosting settings appid** command in privileged EXEC mode.

app-hosting settings appid *application-name***file** *file-path*

Syntax Description

application-name	Name of the application.
file *file-path*	Specifies the file that contains the application settings.

Command Default

Application settings are not enabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Amsterdam 16.12.1	This command was introduced.

Example

The following example shows how to enable the settings of an application:

```
Device# app-hosting settings appid app1 file bootflash:n2os_ids app-data-dir cfg/n2os_ids
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.

app-resource docker

To enable the configuration of runtime Docker options, use the **app-resource docker**command in application hosting configuration mode. To disable the configuration of runtime Docker options, use the **no** form of this command.

app-resource docker
no app-resource docker

This command has no arguments or keywords.

Command Default

Runtime options are disabled.

Command Modes

Application hosting configuration mode (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

When you configure the **app-resource docker** command, the command mode changes to application-hosting docker configuration mode.

Example

The following example shows how to configure the **app-resource docker** command:

```
Device> enable
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-resource docker
Device(config-app-hosting-docker)#
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.

app-resource profile

To override the application-provided resource profile, use the **app-resoure profile** command in application hosting configuration mode. To revert to the application-specified resource profile, use the **no** form of this command.

app-resoure profile *profile-name*
no app-resoure profile {[*profile-name*]}

Syntax Description

profile-name	Name of the resource profile.

Command Default

Resource profile is configured.

Command Modes

Application hosting configuration (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

Reserved resources specified in the application package can be changed by setting a custom resource profile. Only the CPU, memory, and virtual CPU (vCPU) resources can be changed. For the resource changes to take effect, stop and deactivate the application, then activate and start it again.

Note Only custom profile is supported.

The command configures the custom application resource profile, and enters custom application resource profile configuration mode.

Example

The following example shows how to change the allocation of resources of an application:

```
Device# configure terminal
Device(config)# application-hosting appid iox_app
Device(config-app-hosting)# app-resource profile custom
Device(config-app-resource-profile-custom)#
```

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.

app-vnic gateway

Note This command is supported only on routing platforms. It is not supported on switching platforms.

To configure a virtual network interface gateway for an application, use the **app-vnic gateway** command in application hosting configuration mode. To remove the configuration, use the **no** form of this command.

app-vnic gateway virtualportgroup *ip-address* **guest-interface** *network-interface-number*
no app-vnic gateway [**virtualportgroup** *ip-address* **guest-interface** *network-interface-number*]

Syntax Description

virtualportgroup *number*	Configures a VirtualPortGroup interface for the gateway.
guest-interface *network-interface-number*	Configures a guest interface for the gateway.

Command Default

The virtual network gateway is not configured.

Command Modes

Application hosting configuration (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

After you configure the virtual network interface gateway for an application, the command mode changes to application-hosting gateway configuration mode. In this mode, you can configure the IP address of the guest interface.

Example

The following example shows how to configure the management gateway of an application:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-vnic gateway1 virtualportgroup 0 guest-interface 1
Device(config-app-hosting-gateway)# guest-ipaddress 10.0.0.3 netmask 255.255.255.0
Device(config-app-hosting-gateway)#
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
guest-ipaddress	Configures an IP address for the guest interface.

app-vnic AppGigabitEthernet

To configure the front-panel port for application hosting, use the **app-vnic AppGigabitEthernet** command in application hosting configuration mode. To remove a front-panel port, use the **no** form of this command.

app-vnic AppGigabitEthernet {**access** | **trunk**}
no app-vnic AppGigabitEthernet {**access** | **trunk**}

Syntax Description

access	Configures.
trunk	Configures the front-panel trunk port for application hosting.

Command Default

Front-panel ports are not configured for application hosting.

Command Modes

Application hosting configuration (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

Cisco Catalyst 9300 Series Switches support front-panel trunk ports for application hosting.

You can configure the front-panel port as either a trunk interface or a VLAN-specific interface. When using as a trunk interface, the front-panel port is extended to work as a Layer 2 trunk port, and all traffic received by the port is available to the application. When using the port as a VLAN interface, the application is connected to a specific VLAN network. A VLAN interface is created on the host and it is associated with the front-panel port *eth0* interface.

Example

The following example shows how to configure the front-panel trunk port for application hosting:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-vnic AppGigbitEthernet trunk
Device(config-config-app-hosting-trunk)#
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.

app-vnic management

To configure the management gateway of the virtual network interface, use the **app-vnic management** command in application hosting configuration mode. To remove the configuration, use the **no** form of this command.

app-vnic management guest-interface *network-interface-number*
no app-vnic management [**guest-interface** *network-interface-number*]

Syntax Description

guest-interface *network-interface-number*	Configures a guest interface for the gateway.

Command Default

Management gateway is not configured.

Command Modes

Application hosting configuration (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

After you configure the management gateway of an application, the command mode changes to application-hosting management-gateway configuration mode. In this mode, you can configure the IP address of the guest interface.

Example

The following example shows how to configure the management gateway of an application:

```
Device# configure terminal
Device(config)# app-hosting appid lxc_app
Device(config-app-hosting)# app-vnic management guest-interface 0
Device(config-app-hosting-mgmt-gateway)# guest-ipaddress 172.19.0.24 netmask 255.255.255.0
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
guest-ipaddress	Configures an IP address for the guest interface.

boot ipxe

To configure iPXE boot, use the **boot ipxe** command in global configuration mode. To disable the configuration, use the **no** form of this command.

boot ipxe {**forever** | **timeout** *seconds*} **switch** *switch-number*
no boot ipxe {**forever** | **timeout** *seconds*} **switch** *switch-number*

Syntax Description

forever	Attempts iPXE boot forever.
timeout *seconds*	Configures a timeout in seconds for iPXE network boot. Valid values are from 1 to 2147483647.
switch *switch-number*	Enables iPXE boot for switches in the stack. Valid values are from 0 to 9.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Denali 16.3.2	This command was introduced on Cisco Catalyst 3650 and 3850 Series Switches.
Cisco IOS XE Everest 16.6.1	This command was implemented on Cisco Catalyst 9300 and 9500 Series Switches

Usage Guidelines

iPXE is an open source implementation of the Preboot eXecution Environment (PXE). Bootloaders boot an image located on a File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), or Trivial File Transfer Protocol (TFTP) server.

If the **forever** keyword is configured, the switch sends Dynamic Host Configuration Protcol (DHCP) requests forever. If the **timeout** keyword is configured, DHCP requests are sent for the specified amount of time, and when the timeout expires, the switch reverts to device boot.

Example

The following example shows how to configure an iPXE boot timeout for switch 2:

```
Device(config)# boot ipxe timeout 240 switch 2
```

boot manual

To configure manual boot, use the **boot manual** command in global configuration mode. To remove the configuration, use the **no** form of this command.

boot manual switch *switch-number*
no boot manual switch *switch-number*

Syntax Description

switch *switch-number*	Configures manual boot for the switches in the stack.

Command Default

Manual boot is enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Denali 16.3.2	This command was introduced on Cisco Catalyst 3650 and 3850 Series Switches.
Cisco IOS XE Everest 16.6.1	This command was implemented on Cisco Catalyst 9300 and 9500 Series Switches

Usage Guidelines

When manual boot is disabled, and the switch reloads, the boot process starts automatically. When manual boot is disabled, the bootloader determines whether to execute a device boot or a network boot based on the configured value of the iPXE ROMMON variable.

Example

The following example shows how to configure manual boot for switch 2:

```
Device(config)# boot manual switch 2
```

boot system

To enable a system image boot, use the **boot system** command in global configuration mode. To disable the configuration, use the **no** form of this command.

boot system switch {**all** *number*} {**flash:** | **ftp:** | **http:** | **tftp:**}

no boot system [**switch** | {**all** *number*}] [**flash:** | **ftp:** | **http:** | **tftp:**]

Syntax Description

flash:	Specifies the flash filesytem to boot an image.
ftp:	Specifies a File Transfer Protocol (FTP) location to boot an image.
http:	Specifies a Hypertext Transfer Protocol (HTTP) location to boot an image.
tftp:	Specifies a Trivial File Transfer Protocol (TFTP) location to boot an image.
switch *number*	Enables booting for switches in a stack. Valid values are from 0 to 9.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Denali 16.3.2	This command was introduced on Cisco Catalyst 3650 and 3850 Series Switches.
Cisco IOS XE Everest 16.6.1	This command was implemented on Cisco Catalyst 9300 and 9500 Series Switches

Usage Guidelines

You can either use an IPv4 or an IPv6 address for the remote FTP/HTTP/TFTP servers. When using an IPv6 address, you must enter the IPv6 address inside square brackets (as per RFC 2732); otherwise, the device will not boot.

Note IPv6 is not supported on Catalyst 9000 Series Switches.

Example

The following example shows how to boot an image from an IPv4 HTTP server:

```
Device(config)# boot system switch 1 http://192.0.2.42/image-filename
```

The following example shows how to boot an image from an IPv6 HTTP server:

```
Device(config)# boot system switch 1 http://[2001:db8::1]/image-filename
```

ca-trustpoint

To configure the server Certificate Authority (CA) trustpoint for a gRPC telemetry connection, use the **ca-trustpoint** command in telemetry gRPC-protocol profile configuration mode. To remove the server CA trustpoint, use the **no** form of this command

ca-trustpoint *trustpoint*
no ca-trustpoint *trustpoint*

Syntax Description

trustpoint	Name of the server CA trustpoint.

Command Default

Server CA trustpoint is not configured.

Command Modes

Telemetry gRPC-protocol profile configuration (config-mdt-protocol-grpc-profile)

Command History

Release	Modification
Cisco IOS XE Cupertino 17.9.1	This command was introduced.

Usage Guidelines

Crypto trustpoints and certifications are defined in the telemetry configuration by using

- gNOI cert.proto,
- Cisco-IOS-XE-crypto-rpc.YANG, or
- the **crypto pki import** command.

Example

The following example shows how to configure a server CA trustpoint for a gRPC telemetry connection:

```
Device> enable
Device# configure terminal
Device(config)# telemetry protocol grpc profile myprofile
Device(config-mdt-protocol-grpc-profile)# ca-trustpoint server-ca
Device(config-mdt-protocol-grpc-profile)#
```

Related Commands

Command	Description
id-trustpoint	Configures a client ID trustpoint for a gRPC telemetry connection.
telemetry protocol grpc profile	Configures a profile for the gRPC telemetry connection.

clear configuration lock

To clear the configuration session lock, use the **clear configuration lock** in privileged EXEC mode.

clear configuration lock

This command has no arguments or keywords.

Command Default

Session lock times out after 10 minutes.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release Fuji 16.8.1	This command was introduced.

Usage Guidelines

Use this command to remove the configuration lock on a session. A full synchronization of the database is triggered when a lock is cleared.

Read operation is allowed by any NETCONF/RESTCONF sessions during the global lock. However, write operation is only allowed by the NETCONF session that owns the lock.

Example

The following example shows how to clear a configuration lock:

```
Device# clear configuration lock
```

clear netconf-yang session

To clear NETCONF-YANG sessions, use the **clear netconf-yang session** command in privileged EXEC mode.

clear netconf-yang session *session-id*
[**R0** | **R1** | **RP** {**active** | **standby**}]

Syntax Description

session-id	Clears the specified session. Valid values are from 1 to 4294967295.
R0	(Optional) Clears the Route Processor (RP) slot 0.
R1	(Optional) Clears the RP slot 1.
RP	(Optional) Clears the RP.
active	(Optional) Clears the active instance of the RP.
standby	(Optional) Clears the standby instance of the RP.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.8.1	This command was introduced.

Usage Guidelines

You can use this command to unlock a datastore by killing the locked session that has the ownership of the datastore lock. When a global lock is cleared by using the **clear netconf-yang session** command, a full synchronization of the datastore is triggered. However; clearing a session while the global lock is in place, only schedules a full synchronization.

Examples

The following example shows how to clear a NETCONF-YANG session:

```
Device# clear netconf-yang session 2 RP active
```

clear telemetry ietf subscription

To clear dynamic subscriptions, use the **clear telemetry ietf subscription** command in privileged EXEC mode.

clear telemetry ietf subscription *subscription-ID*

Syntax Description

subscription-ID	Dynamic subscription ID.

Command Default

Subscriptions are not cleared.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.11.1	This command was introduced.

Usage Guidelines

You can delete dynamic subscriptions by using the **clear telemetry ietf subscription** command, the <kill-subscription> RPC, and the in-band <delete subscription> RPC.

A subscription is also deleted when the parent NETCONF session is torn down or disconnected. If the network connection is interrupted, it may take some time for the SSH/NETCONF session to timeout, and subsequent subscriptions to be removed.

Example

The following sample output displays all subscriptions:

```
Device# show telemetry ietf subscription all

  Telemetry subscription brief

  ID               Type        State       Filter type
  ---------------------------------------------------------
  2147483648       Dynamic     Valid       xpath
  2147483649       Dynamic     Valid       xpath
```

The following example shows how to clear dynamic subscriptions:

```
Device# clear telemetry ietf subscription 2147483648
```

The following sample output displays all available subscriptions:

```
Device# show telemetry ietf subscription all

  Telemetry subscription brief

  ID               Type        State       Filter type
  ---------------------------------------------------------
  2147483649       Dynamic     Valid       xpath
```

Related Commands

Command	Description
show telemetryietf subscription	Display information about telemetry subscriptions on a device.
telemetry ietf subscription	Creates a telemetry subscription and enters telemetry-subscription mode.

cpu (App Hosting)

To change the CPU quota/unit allocated for an application, use the **cpu** command in custom application resource profile configuration mode. To revert to the application-provided CPU quota, use the **no** form of this command.

cpu *unit*
no cpu [*unit*]

Syntax Description

unit	CPU quota to be allocated for an application. Valid values are from 0 to 20000.

Command Default

Default CPU depends on the platform.

Command Modes

Custom application resource profile configuration (config-app-resource-profile-custom)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.1	This command was introduced.

Usage Guidelines

A CPU unit is the minimal CPU allocation by the application. Total CPU units is based on normalized CPU units measured for the target device.

Within each application package, an application-specific resource profile is provided that defines the recommended CPU load, memory size, and number of virtual CPUs (vCPUs) required for the application. Use this command to change the allocation of resources for specific processes in the custom resource profile.

Reserved resources specified in the application package can be changed by setting a custom resource profile. Only the CPU, memory, and vCPU resources can be changed. For the resource changes to take effect, stop and deactivate the application, then activate it and start it again.



Note Resource values are application-specific, and any adjustment to these values must ensure that the application can run reliably with the changes.

Examples

The following example shows how to override the application-provided CPU quota using a custom resource profile:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-resource profile custom
Device(config-app-resource-profile-custom)# cpu 7400
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.

Command	Description
app-resource profile	Overrides the application-provided resource profile.

dampening-period

To configure a dampening interval for on-change subscriptions, use the **dampening-period** command in update on-change configuration mode. To remove the dampening interval, use the **no** form of this command.

dampening-period *interval*
no dampening-period [*interval*]

Syntax Description

interval	The dampening-period interval in centiseconds.

Command Default

Dampening period is not configured.

Command Modes

Update on-change configuration mode (config-update-onchange)

Command History

Release	Modification
Cisco IOS XE Dublin 17.11.1	This command was introduced.

Usage Guidelines

You can configure a dampening period for on-change subscriptions. When a dampening period is configured, the publisher streams the latest version of all changed records at the end of the period. The dampening period is supported only for native TDL telemetry.

Without a dampening period, the receiver may be flooded with repeated updates that could exhaust the resources in both the publisher and receiver.

The dampening period is configured in the unit of 100th of a second. Based on the platform there is a maximum and minimum limit that can be configured for the dampening-period interval.

The output of the **show telemetry ietf subscription detail** commands displays the configured dampening period.

Subscription dampening is not supported for complex event processing (CEP) transforms.

Example

The following example shows how to configure a dampening period for on-change subscriptions:

```
Device> enable
Device# configure terminal
Device(config)# telemetry ietf subscription 1003
Device(config-mdt-subs)# update-policy on-change
Device(config-update-onchange)# dampening-period 0
```

The following output from the **show telemetry ietf subscriptiondetail** command displays the configured dampening period:

```
Device# show telemetry ietf subscription 1003 detail

Telemetry subscription detail:

  Subscription ID: 1003
  Type: Configured
  State: Valid
```

```
Stream: native
Filter:
  Filter type: tdl-uri
  TDL-URI: /services;serviceName=ewlc_oper/capwap_data
Update policy:
  Update Trigger: on-change
  Synch on start: Yes
  Dampening period: 9000
Encoding: encode-tdl
Source VRF:
Source Address:
Notes: Subscription validated
```

Related Commands

Command	Description
show telemetry ietf subscription	Displays information about telemetry subscriptions on a device.
telemetry ietf subscription	Configures telemetry subscription.
update-policy on-change	Configures on-change updates for a subscription.

debug netconf-yang

To log NETCONF-YANG debug messages, use the **debug netconf-yang** command in privileged EXEC mode.

debug netconf-yang [**level** {**debug** | **emergency** | **error** | **info** | **noise** | **notice** | **verbose** | **warning**}]

no debug netconf-yang [**level** {**debug** | **emergency** | **error** | **info** | **noise** | **notice** | **verbose** | **warning**}]

Syntax Description

level	(Optional) Specifies the log level of NETCONG-YANG processes.
debug	(Optional) Logs debug messages.
emergency	(Optional) Logs emergency messages.
error	(Optional) Logs error messages.
info	(Optional) Logs information messages.
noise	(Optional) Specifies the maximum log level setting. This setting includes all logs in the output such as, emergency, alert, critical, error, warning, notice, debug, verbose and so on.
notice	(Optional) Logs notice messages.
verbose	(Optional) Logs debug messages in detail.
warning	(Optional) Logs warning messages.

Command Default

Debug logs are not enabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.8.1	This command was introduced.

Usage Guidelines

The last enabled debug logging level is used for logging debug messages. For example, if **warning** level is enabled by NETCONF-YANG, and it is followed by **debug** level by RESTCONF; then debug messages are logged.

The last enabled debug logging level will remain persistent for data model interface (DMI) processes.

Examples

The following is sample output from the **debug netconf-yang level debug** command:

```
Device# debug netconf-yang level debug

Jan 24 13:33:20.441 EST: yang-infra: netconf-yang server log level set to debug
```

Related Commands

Command	Description
debug netconf-yang diagnostics	Enables the debugging of NETCONF-YANG diagnostics.

debug netconf-yang diagnostics

To enable the debugging of NETCONF-YANG diagnostics, use the **debug netconf-yang diagnostics** command in privileged EXEC mode.

debug netconf-yang diagnostics diag-level { **basic** | **maximum** }
no debug netconf-yang diagnostics diag-level { **basic** | **maximum** }

Syntax Description

diag-level	Specifies the level for the NETCONF-YANG diagnostics debugging.
basic	Enables the debugging of diagnostics information that contains data model interface (DMI) logs, ConfD logs, and rollback logs.
maximum	Enables the debugging of all diagnostic information, and the running configuration snapshots.

Command Default

Diagnostic debugs are not enabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.5.1	This command was introduced.

Example

The following example shows how to enable basic diagnostic debug messages:

```
Device> enable
Device# debug netconf-yang diagnostics diag-level basic

Diagnostic debugging is on
```

Related Commands

Command	Description
debug netconf-yang	Logs NETCONF-YANG debug messages.
show platform software yang-management process state	Displays the NETCONF-YANG process states.

debug restconf

To log RESTCONF debug messages, use the **debug restconf** command in privileged EXEC mode.

debug restconf [**level** {**debug** | **emergency** | **error** | **info** | **noise** | **notice** | **verbose** | **warning**}]

no debug restconf [**level** {**debug** | **emergency** | **error** | **info** | **noise** | **notice** | **verbose** | **warning**}]

Syntax Description

level	(Optional) Specifies the log level of RESTCONF processes.
debug	(Optional) Logs debug messages.
emergency	(Optional) Logs emergency messages.
error	(Optional) Logs error messages.
info	(Optional) Logs information messages.
noise	(Optional) Specifies the maximum log level setting. This setting includes all logs in the output such as, emergency, alert, critical, error, warning, notice, debug, verbose and so on.
notice	(Optional) Logs notice messages.
verbose	(Optional) Logs debug messages in detail.
warning	(Optional) Logs warning messages.

Command Default

Debug logs are not enabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.8.1	This command was introduced.

Usage Guidelines

The last enabled debug logging level will be used for logging debug messages. For example, if **warning** level is enabled by NETCONF-YANG, and it is followed by **debug** level by RESTCONF; then debug level messages will be logged.

The last enabled debug logging level will remain persistent for data model interface (DMI) processes.

Examples

The following is sample output from the **debug restconf** command:

```
Device# debug restconf

Device# show debug

IOSXE Conditional Debug Configs:

Conditional Debug Global State: Stop

IOSXE Packet Tracing Configs:
```

```
license policy manager client:
  platform software policy_manager_error debugging is on

Packet Infra debugs:

Ip Address                                                  Port
------------------------------------------------------------|-----------
```

```
netconf-yang:
  netconf-yang debugging is on at level debug

restconf:
  restconf debugging is on at level debug
```

default boot

To modify the default boot system parameters, use the **defaut boot** command in global configuration mode.

default boot {**ipxe** {**forever** | **timeout** | **seconds**} | **manual** | **system** {**flash:** | **ftp:** | **http:** | **tftp:**}}**switch** *number*

Syntax Description

ipxe	Enables iPXE boot.
forever	Attempts iPXE boot forever.
timeout *seconds*	Configures a boot timeout in seconds. Valid values are from 1 to 2147483647.
manual	Enables manual boot.
system	Enables a system image boot.
flash:	Specifies the flash filesytem to boot an image.
ftp:	Specifies an File Transfer Protocol (FTP) location to boot an image.
http:	Specifies an Hypertext Transfer Protocol (HTTP) location to boot an image.
tftp:	Specifies a Trivial File Transfer Protocol (TFTP) location to boot an image.
switch *number*	Enables booting for switches in a stack. Valid values are from 0 to 9.

Command Default

Device boot is enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Denali 16.3.2	This command was introduced on Cisco Catalyst 3650 and 3850 Series Switches.
Cisco IOS XE Everest 16.6.1	This command was implemented on Cisco Catalyst 9300 and 9500 Series Switches

Usage Guidelines

You can either use the **no boot ipxe** or the **default boot ipxe** command to configure device boot.

If the **forever** keyword is configured, the switch sends Dynamic Host Configuration Protocol (DHCP) requests forever. If the **timeout** keyword is configured, DHCP requests are sent for the specified amount of time, and when the timeout expires, the switch reverts to device boot.

Examples

The following example shows how to enable the default boot mode:

```
Device(config)# default boot ipxe
```

dig

To do a lookup of the Domain Name System (DNS) server, use the **dig** command in rommon mode.

dig *hostname* {*v4 v6*} [*dns-server-address*]

Syntax Description

hostname	DNS host name
v4	IPv4 address.
v6	IPv6 address.
dns-server-address	(Optional) DNS Server IP address.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

This command does a look up of the DNS name and displays the IP/IPv6 address of the DNS server.

Example

The following is sample output from the **dig** *hostname* command:

```
Device: dig example.org

DNS lookup using 2001:DB8::1
addr = 2001:DB8:0000:0000:0000:0000:0000:0001
```

The following is sample output from the **dig** *hostname v4* command:

```
Device: dig example.org v4

DNS lookup using 10.29.27.5
addr = 172.16.0.1
```

The following is sample output from the **dig** *hostname v4 dns-server-address* command:

```
Device: dig example.org v4 10.29.27.5

DNS lookup using 10.29.27.5
addr = 172.16.0.1
```

The following is sample output from the **dig** *hostname v6* command:

```
Device: dig example.org v6

DNS lookup using 2001:DB::1
addr = 2001:DB8:0000:0000:0000:0000:0000:0001
```

Related Commands

Command	Description
net-debug	Displays or changes the network debug values.

enable (App Hosting)

To enable the AppGigabitEthernet port, use the **enable** command in interface configuration mode. To disable the port, use the **no** form of this command.

enable

no enable

This command has no arguments or keywords.

Command Default The AppGigabitEthernet port is not enabled.

Command Modes Interface configuration (config-if)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.5.1	This command was introduced on Cisco Catalyst 9410 Series Switches.

Usage Guidelines

Note This command is supported only on Cisco Catalyst 9410 Series Switches

In a high availability setup, we recommend that you configure the **enable** command on both the AppGigabitEthernet interface ports.

Example

The following example shows how to enable the AppGigabitEthernet interface:

```
Device> enable
Device# configure terminal
Device(config)# interface AppGigabitEthernet 1/0/1
Device(config-if)# enable
```

encoding

To configure telemetry encoding for a subscription, use the **encoding** command in telemetry-subscription configuration mode.

encoding { **encode-kvgpb** | **encode-tdl** }

Syntax Description

encode-kvgpb	Configures Key-value Google Protocol Buffers (kvGPB) encoding.
encode-tdl	Configures TDL encoding.

Command Modes

Telemetry-subscription configuration (config-mdt-subs)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.10.1	This command was introduced.
Cisco IOS XE Bengaluru 17.6.1	This command was modified. The **encode-tdl** keyword was added.

Example

The following example shows how to configure telemetry encoding for a subscription:

```
Device> enable
Device# configure terminal
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# encoding encode-kvGPB
```

Related Commands

Command	Description
telemetry ietf subscription	Configures telemetry subscription.

filter

To configure a filter, use the **filter** command in telemetry-subscription configuration mode.

filter { **nested-uri** | **tdl-transform** | **tdl-uri** | **xpath** } *filter*

Syntax Description

nested-uri	Configures a nested uniform resource identifier (URI) filter.
tdl-transform	Configures a top-level domain (TDL) transform filter.
tdl-uri	Configures a TDL URI filter.
xpath	Configures an XPath filter.
path	Specifies XPath filter.

Command Modes

Telemetry-subscription configuration (config-mdt-subs)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.10.1	This command was introduced.
	This command was modified. The **nested-uri**, **tdl-transform**, and **tdl-uri** keywords were added.

Usage Guidelines

The set of events from a stream are filtered. Different filter types are used for different stream types. Cisco IOS XE supports the yang-push stream.

The dataset within the yang-push stream to be subscribed to is specified by the use of an XPath filter.

Example

The following example shows how to configure XPath filter for subscription:

```
Device> enable
Device# configure terminal
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# filter xpath /memory-ios-xe-oper:memory-statistics/memory-statistic
```

Related Commands

Command	Description
telemetry ietf subscription	Configures telemetry subscription.

gnxi

To enable the Google Remote Procedure Call (gRPC) Network Operations Interface (gNOI) or gNxI tools, use the **gnxi** command in global configuration mode. To disable gNOI, use the **no** form of this command.

gnxi [**port** *port-number* | **secure-allow-self-signed-trustpoint** | **secure-client-auth** | **secure-init** | **secure-password-auth** | **secure-peer-verify-trustpoint** *trustpoint-name* | **secure-port** *port-number* | **secure-server** | **secure-trustpoint** *trustpoint-name* | **server**]

no gnxi [**port** { [*port-number*] } | **secure-allow-self-signed-trustpoint** | **secure-client-auth** | **secure-init** | **secure-password-auth** | **secure-peer-verify-trustpoint** [*trustpoint-name*] | **secure-port** { [*port-number*] } | **secure-server** | **secure-trustpoint** [*trustpoint-name*] | **server**] [**grpctunnel target** { [*GNMI_GNOI*] | *GNMI_GNOI_INSECURE* }] [**grpctunnel destination** { [*address*] | [*port*] | [*enable*] | [*identity-trustpoint*] | [*insecure*] | [*source-address*] | [*source-vrf*] }]

Syntax Description

port *port-number*	(Optional) Specifies the gNMI port number. Valid values for the *port-number* argument are from 1024 to 65535.
secure-allow-self-signed-trustpoint	(Optional) Allows the gNMI secure server to use a self-signed certificate.
secure-client-auth	(Optional) Sets the gNMI client authentication.
secure-init	(Optional) Enables the gNMI secure server by using the primary self-signed certificate.
secure-password-auth	(Optional) Sets the gNMI password authentication.
secure-peer-verify-trustpoint *trustpoint-name*	(Optional) Sets the gNMI server peer validation for the specified trustpoint.
secure-port *port-number*	(Optional) Sets the gNMI secure server port. Valid values for the *port-number* argument are from 1024 to 65535.
secure-server	(Optional) Enables the gNMI secure server.
secure-trustpoint *trustpoint-name*	(Optional) Sets the gNMI server certificate trustpoint.

server	(Optional) Enables the gNMI insecure server. **Note** In Cisco IOS XE 26.1.1, this keyword was removed. Instead of this keyword, use the **secure-server** keyword.
grpctunnel target*GNMI_GNOI* | *GNMI_GNOI_INSECURE*	• GNMI_GNOI—gNxI Service. For more information, see Github. • GNMI_GNOI_INSECURE—gNxI Service without TLS. For more information, see Github.
grpctunnel destination *address* | *port* | *enable* | *identity-trustpoint* | *insecure* | *source-address* | *source-vrf*	• address—Specify the tunnel server/destination address. Both IPv4 and IPv6 are supported. No FQDN. • port—Specify the destination port. • enable—Enables the destination. • identity-trustpoint—The certificate to use in the TLS handshake when connecting to the tunnel server or destination. • insecure—Disables TLS on the tunnel. Ignores the identity-trustpoint configuration. • source-address—Sets the outgoing source address to use when connecting to the tunnel server or destination. • source-vrf—Sets the outgoing VRF when connecting to the tunnel server or destination.

Command Default gNXI is not configured.

Command Modes Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Amsterdam 17.3.1	This command was introduced. This command replaces the **gnmi-yang** command.
Cisco IOS XE Dublin 17.11.1	The **grpctunnel target** keyword was introduced.
Cisco IOS XE 26.1.1	The **server** keyword was deprecated. Use the **secure-server** keyword instead of the **server** keyword.

Usage Guidelines

If no authentication is configured for the secure-server keyword, the following syslog message is displayed on the console.

```
"secure-client-auth" nor "secure-password-auth" are configured.
Jun 17 2026 14:30:10.723: %GNMIB-4-CONFIG_SECURE_SERVER_NO_AUTH:
Switch 1 R0/0: gnmib: gnxi secure-server configured without
authentication. Configuring authentication is recommended.
Password authentication and mutual TLS are supported
```

The following example shows how to start the gNxI process.

```
Device> enable
Device# configure terminal
Device(config)# gnxi
Device
```

Related Commands

Command	Description
show gnxi state detail	Displays the status of gNMI interfaces.

gnxi access-list

To configure IPv4 or IPv6 service-level access control lists (ACLs) for gRPC Network Management/Operations Interface (gNXI) services, use the **gnxi access-list** command in global configuration mode. To remove the configuration, use the **no** form of this command.

gnxi access-list { **ipv4** | **ipv6** } **name** *named-ACL*
no gnxi access-list { **ipv4** | **ipv6** } **name** *named-ACL*

Syntax Description

ipv4	Configures IPv4 parameters.
ipv6	Configures IPv6 parameters.
name	Configures a named ACL.
named-ACL	Name of the IPv4 or IPv6 named ACL.

Command Default

Service-level ACLs are not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE 26.1.1	This command was introduced.

Usage Guidelines

Only named ACLs are supported.

IPv4 named ACLs apply only to IPv4 traffic and IPv6 ACLs to IPv6 traffic. The service-level ACL configuration applies to all gNXI servers. The ACL check is done after the username and password authentication is successful.

You can also use the Cisco-IOS-XE-gnmi-cfg.yang model to configure IPv4 or IPv6 service-level ACLs.

Example

This example shows how to configure a named service-level ACL.

```
Device> enable
Device# configure terminal
Device(config)# gnxi access-list ipv4 name ipv4-acl
Device(config)# gnxi access-list ipv6 name ipv6-acl
Device(config)# exit
```

gnxi vrf

To configure a gRPC Network Management/Operations Interface (gNXI) server VRF, use the **gnxi vrf** command in global configuration mode. To remove a gNXI server VRF, use the **no** form of this command.

gnxi vrf *vrf-name*
no gnxi vrf *vrf-name*

Syntax Description

vrf-name	Name of the user-configured VRF.

Command Default

gNXI server is not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE 17.18.2	This command was introduced.

Usage Guidelines

When a new gNXI server VRF is configured, new servers are provisioned.

If the configured VRF name does not exist, a syslog message will be displayed on the console.

Example

The following example shows how to configure a gNXI server VRF.

```
Device> enable
Device# configure terminal
Device(config)# gnxi secure-password-auth
Device(config)# gnxi secure-trustpoint gnxi-test-cert
Device(config)# gnxi secure-server
Device(config)# gnxi vrf Gnxi-vrf
```

Table 1: Related commands

Command	Description
show gnxi server vrf	Displays the state of the gNXI servers and configured VRFs.

guest-interface (App Hosting)

To configure a guest interface for the front-panel trunk port, use the **guest-interface** command in application-hosting trunk configuration mode. To remove a guest interface, use the **no** form of this command.

guest-interface *interface-number*
no guest-interface *interface-number*

Syntax Description

interface-number	Guest interface number. Valid values are from 0 to 63.

Command Default

A guest interface is not configured.

Command Modes

Application-hosting trunk configuration (config-config-app-hosting-trunk)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

When you configure the front-panel trunk port for application hosting, the command mode changes to application-hosting trunk configuration mode. Configure the **guest-interface** command in this mode.

Example

The following example shows how to configure a guest-interface for a front-panel trunk port:

```
Device# configure terminal
Device(config)# app-hosting appid lxc_app
Device(config-app-hosting)# app-vnic AppGigEthernet trunk
Device(config-config-app-hosting-trunk)# guest-interface 9
Device(config-config-app-hosting-trunk)# end
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-vnic AppGigEthernet trunk	Configures the front-panel trunk port for application hosting, and enters application-hosting trunk configuration mode.

guest-ipaddress (App Hosting)

To configure an IP address for a guest interface, use the **guest-ipaddress** command in application-hosting gateway, application-hosting management-gateway, or application-hosting VLAN-access IP configuration modes. To remove the guest interface IP address, use the **no** form of this command.

guest-ipaddress *ip-address* **netmask** *netmask*
no guest-ipaddress [*ip-address* **netmask** *netmask*]

Syntax Description

ip-address	IP address of the guest interface.
netmask *netmask*	Specifies the subnet mask for the guest IP address.

Command Default

The guest interface IP address is not configured.

Command Modes

Application-hosting gateway configuration (config-app-hosting-gateway)

Application-hosting management-gateway configuration (config-app-hosting-mgmt-gateway)

Application-hosting VLAN-access IP configuration (config-config-app-hosting-vlan-access-ip)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

Configure this command, after configuring the **app-vnic gateway**, the **app-vnic management**, or **app-vnic AppGigabitEthernet vlan-access** commands.

Use this command to configure the guest interface address for the front-panel VLAN port for application-hosting.

Examples

The following example shows how to configure the guest interface address for a virtual network interface gateway:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-vnic gateway1 VirtualPortGroup 0 guest-interface 1
Device(config-app-hosting-gateway)# guest-ipaddress 10.0.0.3 netmask 255.255.255.0
```

The following example shows how to configure the guest interface address for a management gateway:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-vnic management guest-interface 0
Device(config-app-hosting-mgmt-gateway)# guest-ipaddress 172.19.0.24 netmask 255.255.255.0
```

The following example shows how to configure the guest interface address for the front-panel VLAN port:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
```

```
Device(config-app-hosting)# app-vnic AppGigabitEthernet trunk
Device(config-config-app-hosting-trunk)# vlan 1 guest-interface 9
Device(config-config-app-hosting-vlan-access-ip)# guest-ipaddress 192.168.0.2
netmask 255.255.255.0
Device(config-config-app-hosting-vlan-access-ip)#
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-vnic gateway	Configures a virtual network interface gateway.
app-vnic AppGigabitEthernet trunk	Configures a front-panel trunk port and enters application-hosting trunk configuration mode.
app-vnic management	Configures the management gateway of a virtual network interface.
vlan (App Hosting)	Configures a VLAN guest interface and enters application-hosting VLAN-access IP configuration mode.

guest-ipv6address

To configure an IPv6 address for an application or the guest interface, use the **guest-interface** command in application-hosting VLAN-access IP configuration mode. To remove the IPv6 address, use the **no** form of this command.

guest-ipv6address *ipv6-address* **prefix** *ipv6-prefix*
no guest-ipv6address *ipv6-address* **prefix** [*ipv6-prefix*]

Syntax Description

ipv6-address	IPv6 address of the application or guest interface.
prefix *ipv6-prefix*	Specifies the IPv6 prefix.

Command Default

IPv6 address of the application or interface is not configured.

Command Modes

Application-hosting VLAN-access IP configuration (config-config-app-hosting-vlan-access-ip)

Command History

Release	Modification
Cisco IOS XE Dublin 17.11.1	This command was introduced.

Example

The following example shows how to configure the IPv6 address of an application or the guest interface:

```
Device> enable
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-vnic AppGigabitEthernet trunk
Device(config-config-app-hosting-trunk)# vlan 1 guest-interface 9
Device(config-config-app-hosting-vlan-access-ip)# guest-ipv6address 2001:db8::2 prefix 128
Device(config-config-app-hosting-vlan-access-ip)# end
Device#
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-vnic gateway	Configures a virtual network interface gateway.
app-vnic AppGigabitEthernet trunk	Configures a front-panel trunk port and enters application-hosting trunk configuration mode.
app-vnic management	Configures the management gateway of a virtual network interface.

Command	Description
vlan (App Hosting)	Configures a VLAN guest interface and enters application-hosting VLAN-access IP configuration mode.

guestshell

To configure the Guest Shell infastructure functionality, use the **guestshell** command in privileged EXEC mode.

guestshell {**destroy** | **disable** | **enable** | **run** [*linux-executable*]}

Syntax Description

destroy	Deactivates and uninstalls the Guest Shell service.
disable	Disables the Guest Shell service.
enable	Disables the Guest Shell service.
run [*linux-executable*]	Executes or runs a Linux program in the Guest Shell .

Command Default

Guest Shell is not enabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

Guest Shell is an embedded Linux environment that allows customers to develop and run custom Python applications for automated control and management of Cisco switches. Guest Shell is packaged as a Cisco application hosting framework (CAF)-formatted tar file (guest_shell.tar) into the Cisco IOS XE Everest 16.5.x release image read-only file system.

Configure the **iox** command in global configuration mode, before configuring this command. IOx is the Cisco-developed framework for hosting customer-deployed Linux applications on Cisco networking systems.

Examples

The following example shows how to enable and run the Guest Shell:

```
Device# configure terminal
Device(config)# iox
Device(config)# exit
Device# guestshell enable
Device# guestshell run
```

Related Commands

Command	Description
iox	Configure IOx services.

guestshell portforwarding

To enable Guest Shell port forwarding, use the **guestshell portforwarding** command in privileged EXEC mode.

guestshell portforwarding {**add table-entry** *entry-name* **service** {**tcp** | **udp** }**source-port** *port-number* **destination-port** *port-number* | **delete table-entry** *entry-name* }

Syntax Description

add	Adds an IP table entry.
table-entry *entry-name*	Specifies the IP table name. The *table-name* argument must be unique, and it can be alphanumeric characters.
service	Specifies the service protocol.
tcp	Specifies TCP as the service protocol.
udp	Specifies UDP as the service protocol.
source-port *port-number*	Specifies the source port. Valid values for the *port-number* argument are from 1 to 65535.
destination-port *port-number*	Specifies the destination port. Valid values for the *port-number* argument are from 1 to 65535.
delete	Deletes an IP table entry.

Command Default

Port forwarding is not enabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Everest 16.6.1	This command was introduced.

Usage Guidelines

Use this command to enable port forwarding for Guest Shell, when it connected through the GigabitEthernet 0/0 management interface

Examples

The following example shows how to enable port forwarding for Guest Shell:

```
Device# configure terminal
Device(config)# iox
```

```
Device(config)# exit
Device# guestshell portforwarding add table-entry table1 service tcp
 source-port 32 destination-port 9
Device#
```

The following example shows how to disable port forwarding for Guest Shell:

```
Device# guestshell portforwarding delete table-entry table1
Device#
```

Related Commands

Command	Description
guestshell	Configures the Guest Shell infrastructure functionality.

host

To specify the details of the named receiver host, use the **host** command in telemetry protocol-receiver configuration mode. To remove the host details, use the **no** form of this command.

host { **ip-address** *ip-ipv6-address* | **name** *hostname* } *receiver-port*
no host { **ip-address** *ip-ipv6-address* | **name** *hostname* } *receiver-port*

Syntax Description

ip-address *ip-ipv6-address*	Specifies the host IPv4 or IPv6 address.
name *hostname*	Specifies the hostname.
receiver-port	Destination port number. Valid values are from 0 to 65535.

Command Default

Host details are not specified.

Command Modes

Telemetry protocol-receiver configuration (config-mdt-protocol-receiver)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Usage Guidelines

The host specification for a named receiver takes a hostname or an IP address, and a destination port number.

Example

The following example shows how to configure a host name for a named receiver:

```
Device> enable
Device# configure terminal
Device(config)# telemetry receiver protocol receiver1
Device(config-mdt-protocol-receiver)# host name rcvr.test.com 45000
```

The following example shows how to configure the host IP address:

```
Device> enable
Device# configure terminal
Device(config)# telemetry receiver protocol receiver1
Device(config-mdt-protocol-receiver)# host ip-address 2001:db8::1 45000
```

Related Commands

Command	Description
protocol	Specifies a protocol for the named receiver.
telemetry receiver protocol	Configures a named protocol receiver.

id-trustpoint

To configure the client ID trustpoint for a gRPC telemetry connection, use the **id-trustpoint** command in telemetry gRPC-protocol profile configuration mode. To remove the client ID trustpoint, use the **no** form of this command.

id-trustpoint *trustpoint*
no id-trustpoint *trustpoint*

Syntax Description

trustpoint	Name of the client ID trustpoint.

Command Default

Client ID trustpoint is not configured.

Command Modes

Telemetry gRPC-protocol profile configuration (config-mdt-protocol-grpc-profile)

Command History

Release	Modification
Cisco IOS XE Cupertino 17.9.1	This command was introduced.

Usage Guidelines

Crypto trustpoints and certifications are defined in the telemetry configuration by using

- gNOI cert.proto,
- Cisco-IOS-XE-crypto-rpc.YANG, or
- the **crypto pki import** command.

Example

The following example shows how to configure a client ID trustpoint for a gRPC telemetry connection:

```
Device> enable
Device# configure terminal
Device(config)# telemetry protocol grpc profile myprofile
Device(config-mdt-protocol-grpc-profile)# id-trustpoint client-ca
Device(config-mdt-protocol-grpc-profile)#
```

Related Commands

Command	Description
ca-trustpoint	Configures the server CA trustpoint for a gRPC telemetry connection.
telemetry protocol grpc profile	Configures a profile for the gRPC telemetry connection.

install

To install data model update packages, use the **install** command in privileged EXEC mode.

install {**activate** | **file** {**bootflash:** | **flash:** | **webui:**} [**prompt-level** {**all** | **none**}] | **add file** {**bootflash:** | **flash:** | **ftp:** | **http:** | **https:** | **rcp:** | **scp:** | **tftp:** | **webui:**} [**activate** [**prompt-level** {**all** | **none**}]] | **commit** | **deactivate file** {**bootflash:** | **flash:** | **webui:**} [**prompt-level** {**all** | **none**}] | **remove** {**file** {**bootflash:** | **flash:** | **ftp:** | **http:** | **https:** | **rcp:** | **scp:** | **tftp:** | **webui:**} | **inactive** } | **rollback to** {**base** | **committed** | **id** *install-ID* }}

Syntax Description

activate	Validates whether the model update package is added through the **install add** command, and restarts NETCONF processes (confd and opdatamgrd). This keyword runs a compatibility check, updates package status, and if the package can be restarted, it triggers post-install scripts to restart the necessary processes, or triggers a reload for non-restartable packages.
file	Specifies the package to be activated.
{**bootflash:** | **flash:** | **http:** | **https:** | **rcp:** | **scp:** | **tftp:webui:**}	Specifies the location of the installed package.
prompt-level {**all** | **none**}	(Optional) Prompts the user about installation activities. For example, the **activate** keyword, automatically triggers a reload for packages that require a reload. Before activating the package, a message will prompt users as to whether they want to continue. The **all** keyword allows you to enable prompts. The **none** keyword disables prompts.
add	Copies files from a remote location (via FTP, TFTP) to a device, and performs a compatibility check for the platform and image versions. This keyword runs base compatibility checks to ensure that a specified package is supported on a platform. It also adds an entry in the package file, so that the status can be monitored and maintained.
{**http:** | **https:** | **rcp:** | **scp:** | **tftp:**}	Specifies the package to be added.

commit	Makes changes persistent over reloads. You can do a commit after activating a package, while the system is up, or after the first reload. If a package is activated, but not committed, it remains active after the first reload, but not after the second reload.
deactivate	Deactivates an installed package. Deactivating a package also updates the package status and triggers a process restart or a reload.
remove	Remove installed packages. The package file is removed from the file system. The **remove** keyword can only be used on packages that are currently inactive.
inactive	Removes all inactive packages from the device.
rollback	Rolls back the data model update package to the base version, the last committed version, or a known commit ID, and restarts NECONF processes.
to base	Returns to the base image.
committed	Returns to the installation state when the last commit operation was performed.
id *install-ID*	Returns to the specific install point ID. Valid values are from 1 to 4294967295.

Command Default

Model update packages are not installed.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced on the following platforms: • Cisco 4000 Series Integrated Services Routers • Cisco Catalyst 9300 Series Switches • Cisco Catalyst 9500 Series Switches • Cisco Cloud Services Router 1000v • Cisco Integrated Services Virtual Routers (ISRv)

Release	Modification
Cisco IOS XE Everest 16.6.1	This command was implemented on the following platforms: • Cisco Catalyst 3650 Series Switches • Cisco Catalyst 3850 Series Switches

Usage Guidelines

In Service Model Update adds new data models or extend functionality to existing data models. The update package provides YANG model enhancements outside of a release cycle. The update package is a superset of all existing models; it includes all existing models as well as updated YANG models.

A model update package must be added prior to activating the update package. A package must be deactivated, before it is removed from the bootflash.

Cisco 4000 Series Integrated Services Routers

The following example shows how to add an install package on a device:

```
Device# install add file tftp://172.16.0.1//tftpboot/folder1/isr4300-
universalk9.2017-01-10_13.15.1.CSCxxxxxxx.dmp.bin

install_add: START Sun Feb 26 05:57:04 UTC 2017
Downloading file tftp://172.16.0.1//tftpboot/folder1/isr4300-universalk9.2017-01-10_13.15.1.
CSCxxxxxxx.dmp.bin
Finished downloading file
tftp://172.16.0.1//tftpboot/folder1/isr4300-universalk9.2017-01-10_13.15.1.
CSCxxxxxxx.dmp.bin to bootflash:isr4300-universalk9.2017-01-10_13.15.1.CSCxxxxxxx.dmp.bin
SUCCESS: install_add /bootflash/isr4300-universalk9.2017-01-10_13.15.1.CSCxxxxxxx.dmp.bin
Sun Feb 26 05:57:22 UTC 2017
```

The following example shows how to activate an install package:

```
Device# install activate file bootflash:
isr4300-universalk9.2017-01-10_13.15.1.CSCxxxxxxx.dmp.bin

install_activate: START Sun Feb 26 05:58:41 UTC 2017
DMP package.
Netconf processes stopped
SUCCESS: install_activate /bootflash/isr4300-universalk9.2017-01-10_13.15.1.CSCxxxxxxx.dmp.bin

Sun Feb 26 05:58:58 UTC 2017
*Feb 26 05:58:47.655: %DMI-4-CONTROL_SOCKET_CLOSED:  SIP0: nesd:
Confd control socket closed Lost connection to ConfD (45): EOF on socket to ConfD.
*Feb 26 05:58:47.661: %DMI-4-SUB_READ_FAIL:  SIP0: vtyserverutild:
Confd subscription socket read failed Lost connection to ConfD (45):
EOF on socket to ConfD.
*Feb 26 05:58:47.667: %DMI-4-CONTROL_SOCKET_CLOSED:  SIP0: syncfd:
Confd control socket closed Lost connection to ConfD (45): EOF on socket to ConfD.
*Feb 26 05:59:43.269: %DMI-5-SYNC_START:  SIP0: syncfd:
External change to running configuration detected.
The running configuration will be synchronized to the NETCONF running data store.
*Feb 26 05:59:44.624: %DMI-5-SYNC_COMPLETE:  SIP0: syncfd:
The running configuration has been synchronized to the NETCONF running data store.
```

The following example shows how to commit an installed package:

```
Device# install commit

install_commit: START Sun Feb 26 06:46:48 UTC 2017
SUCCESS: install_commit  Sun Feb 26 06:46:52 UTC 2017
```

The following example shows how to rollback to the base package:

```
Device# install rollback to base

install_rollback: START Sun Feb 26 06:50:29 UTC 2017
7 install_rollback: Restarting impacted processes to take effect
7 install_rollback: restarting confd

*Feb 26 06:50:34.957: %DMI-4-CONTROL_SOCKET_CLOSED:  SIP0: syncfd:
Confd control socket closed Lost connection to ConfD (45): EOF on socket to ConfD.
*Feb 26 06:50:34.962: %DMI-4-CONTROL_SOCKET_CLOSED:  SIP0: nesd:
Confd control socket closed Lost connection to ConfD (45): EOF on socket to ConfD.
*Feb 26 06:50:34.963: %DMI-4-SUB_READ_FAIL:  SIP0: vtyserverutild:
Confd subscription socket read failed Lost connection to ConfD (45):
EOF on socket to ConfD.Netconf processes stopped
7 install_rollback: DMP activate complete
SUCCESS: install_rollback  Sun Feb 26 06:50:41 UTC 2017
*Feb 26 06:51:28.901: %DMI-5-SYNC_START:  SIP0: syncfd:
External change to running configuration detected.
The running configuration will be synchronized to the NETCONF running data store.
*Feb 26 06:51:30.339: %DMI-5-SYNC_COMPLETE:  SIP0: syncfd:
The running configuration has been synchronized to the NETCONF running data store.
```

Cisco Catalyst 3000 Series Switches

The following example shows how to add an install package on a device:

```
Device# install add file tftp://172.16.0.1//tftpboot/folder1/i
cat3k_caa-universalk9.16.06.01.CSCxxxxxxx.dmp.bin

install_add: START Sat Jul 29 05:57:04 UTC 2017
Downloading file tftp://172.16.0.1//tftpboot/folder1/
cat3k_caa-universalk9.16.06.01.CSCxxxxxxx.dmp.bin
Finished downloading file tftp://172.16.0.1//tftpboot/folder1/
cat3k_caa-universalk9.16.06.01.CSCxxxxxxx.Sdmp.bin to
bootflash:cat3k_caa-universalk9.16.06.01.CSCxxxxxxx.dmp.bin
SUCCESS: install_add /bootflash/
cat3k_caa-universalk9.16.06.01.CSCxxxxxxx.dmp.bin
Sat Jul 29 05:57:22 UTC 2017
```

The following sample output from the **show install summary** command displays that the update package is now committed, and that it will be persistent across reloads:

```
Device# show install summary

Active Packages:
bootflash:cat3k_caa-universalk9.16.06.01.CSCxxxxxxx.dmp.bin
Inactive Packages:
No packages
Committed Packages:
bootflash:cat3k_caa-universalk9.16.06.01.CSCxxxxxxx.dmp.bin
Uncommitted Packages:
No packages
Device#
```

Related Commands

Command	Description
show install	Displays information about model update packages.

iox

To configure IOx services, use the **iox** command in global configuration mode. To remove the configuration, use the **no** form of this command.

iox
no iox

This command has no arguments or keywords.

Command Default

IOx services are not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

IOx is the Cisco-developed framework for hosting customer-deployed Linux applications on Cisco networking systems. IOx facilitates the life-cycle management of app and data exchange by providing a set of services that helps developers to package pre-built apps, and host them on a target device. IOx life-cycle management includes distribution, deployment, hosting, starting, stopping (management), and monitoring of apps and data. IOx services also include app distribution and management tools that help users discover and deploy apps to the IOx framework.

Examples

The following example shows how to configure IOx services:

```
Device# configure terminal
Device(config)# iox
Device(config)# exit
```

Related Commands

Command	Description
guestshell	Configures Guest Shell infrastructure functionality.

keepalive (grpc-profile)

To configure a gRPC keepalive interval, use the **keepalive** command in MDT-protocol gRPC profile configuration mode. To remove the keepalive interval, use the **no** form of this command.

keepalive *interval*
no keepalive *interval*

Syntax Description

interval	The keepalive interval in seconds. Valid values are from 0 to 4294967295. An interval of 0 disables the keepalive interval.

Command Default

No keepalive interval is configured.

Command Modes

MDT-protocol gRPC profile configuration mode (config-mdt-protocol-grpc-profile)

Command History

Release	Modification
Cisco IOS XE 26.1.1	This command was introduced.

Usage Guidelines

Ensure that the keepalive interval is set to a value lower than the timeout threshold of any intermediate network devices. For example, if your proxy drops idle connections after 300 seconds, a 15-second keepalive is appropriate.

This command within the telemetry gRPC profile ensures that connections remain active by sending periodic pings, which is useful when traffic passes through network components, such as, proxies or firewalls that may terminate idle connections.

This command is recommended for environments where telemetry subscriptions may experience long periods of inactivity. By enabling the keepalive interval, the gRPC client proactively sends pings to keep the connection state active, preventing premature termination of the telemetry stream.

Example

The following example shows how to configure a keepalive interval for gRPC connections.

```
Device> enable
Device# configure terminal
Device(config)# telemetry protocol grpc profile profile1
Device(config-mdt-protocol-grpc-profile)# ca-trustpoint server_ca
Device(config-mdt-protocol-grpc-profile)# keepalive interval 15
Device(config-mdt-protocol-grpc-profile)# end
Device#
```

Related Commands

Command	Description
ca-trustpoint	Configures a server CA trustpoint for gRPC connections.
telemetry protocol grpc profile	Configures a profile for gRPC connections.

mac-forwarding (App Hosting)

To enable MAC-address forwarding on an interface, use the **mac-forwarding** command in application-hosting VLAN-access IP configuration mode. To disable MAC-address forwarding, use the **no** form of this command.

mac-forwarding
no mac-forwarding

This command has no arguments or keywords.

Command Default

MAC forwarding is not enabled.

Command Modes

Application-hosting VLAN-access IP configuration (config-config-app-hosting-vlan-access-ip)

Command History

Release	Modification
Cisco IOS XE Dublin 17.11.1	This command was introduced.

Example

The following example shows how to enable MAC-address forwarding on an interface:

```
Device> enable
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-vnic AppGigabitEthernet trunk
Device(config-config-app-hosting-trunk)# vlan 1 guest-interface 9
Device(config-config-app-hosting-vlan-access-ip)# guest-ipaddress 192.168.0.2
netmask 255.255.255.0
Device(config-config-app-hosting-vlan-access-ip)# mac-forwarding
Device(config-config-app-hosting-vlan-access-ip)# end
Device#
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-vnic gateway	Configures a virtual network interface gateway.
app-vnic AppGigabitEthernet trunk	Configures a front-panel trunk port and enters application-hosting trunk configuration mode.
app-vnic management	Configures the management gateway of a virtual network interface.
guest-ipaddress (App Hosting)	Configure an IP address for a guest interface.
vlan (App Hosting)	Configures a VLAN guest interface and enters application-hosting VLAN-access IP configuration mode.

memory (App Hosting)

To change the memory allocated by the application, use the **memory** command in custom application resource profile configuration mode. To revert to the application-provided memory size, use the **no** form of this command.

memory *memory*
no memory {[*memory*]}

Syntax Description

memory	Memory allocation in MB. Valid values are from 0 to 4096.

Command Default

The default memory size depends on the platform.

Command Modes

Custom application resource profile configuration (config-app-resource-profile-custom)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.1	This command was introduced.

Usage Guidelines

Within each application package, an application-specific resource profile is provided that defines the recommended CPU load, memory size, and number of virtual CPUs (vCPUs) required for the application. Use this command to change the allocation of resources for specific processes in the custom resource profile.

Reserved resources specified in the application package can be changed by setting a custom resource profile. Only the CPU, memory, and vCPU resources can be changed. For the resource changes to take effect, stop and deactivate the application, then activate it and start it again.

Note Resource values are application-specific, and any adjustment to these values must ensure that the application can run reliably with the changes.

Examples

The following example shows how to override the application-provided memory using a custom resource profile:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-resource profile custom
Device(config-app-resource-profile-custom)# memory 2048
Device(config-app-resource-profile-custom)#
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-resource profile	Overrides the application-provided resource profile.

mirroring

To enable the mirroring of the guest-interface, use the **mirroring** command in application-hosting VLAN-access IP configuration mode. To disable the guest-interface mirroring, use the **no** form of this command.

mirroring
no mirroring

This command has no arguments or keywords.

Command Default

Mirroring is not enabled.

Command Modes

Application-hosting VLAN-access IP configuration (config-config-app-hosting-vlan-access-ip)

Command History

Release	Modification
Cisco IOS XE Dublin 17.11.1	This command was introduced.

Example

The following example shows how to enable mirroring on an AppGigabitEthernet interface:

```
Device> enable
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-vnic AppGigabitEthernet trunk
Device(config-config-app-hosting-trunk)# vlan 1 guest-interface 9
Device(config-config-app-hosting-vlan-access-ip)# guest-ipaddress 192.168.0.2
netmask 255.255.255.0
Device(config-config-app-hosting-vlan-access-ip)# mirroring
Device(config-config-app-hosting-vlan-access-ip)# end
Device#
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-vnic gateway	Configures a virtual network interface gateway.
app-vnic AppGigabitEthernet trunk	Configures a front-panel trunk port and enters application-hosting trunk configuration mode.
app-vnic management	Configures the management gateway of a virtual network interface.
guest-ipaddress (App Hosting)	Configure an IP address for a guest interface.
vlan (App Hosting)	Configures a VLAN guest interface and enters application-hosting VLAN-access IP configuration mode.

mlog

To direct log messages to a memory buffer instead of the serial port, use the **mlog** command in rommon mode.

mlog [**show** | **reset** | **ctrl** [**on** | **off** | **toggle**]]

Syntax Description

show	(Optional) Displays memory log messages.
reset	(Optional) Resets the logging of messages to the memory log.
ctrl	(Optional)
on	(Optional)
off	(Optional)
toggle	(Optional)

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

This command directs protocol log (that is all logs controlled by the **net-debug** command) messages to a memory buffer instead of the serial port.

With memory logging, log messages are displayed after a test is run. For example, HTTP debugs can be enabled through memory logging. Log messages are displayed in the memory buffer after running a copy from http://server/name to null: command.

Example

The following example shows how to direct log messages to the memory buffer:

```
Device: mlog show
```

Related Commands

Command	Description
net-debug	Displays or changes the network debug values.

monitor log profile netconf-yang

To display debug logs for NETCONF-YANG processes, use the **monitor log profile netconf-yang** command in privileged EXEC mode.

monitor log profile netconf-yang internal

Syntax Description

internal	Displays all debug logs. **Note** This keyword is mainly used by customer support.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.8.1	This command was introduced.

Usage Guidelines

Logs generated by this command are rendered on the device console.

Example

The following example shows how to enable the **monitor log profile netconf-yang internal** command:

```
Device# monitor log profile netconf-yang internal

2018/01/24 15:58:50.356 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  gdb port
 9919 allocated
2018/01/24 15:58:50.365 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  swift_repl
 port 8019 allocated
2018/01/24 15:58:50.430 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  process
 scoreboard /tmp/rp/
 process/pttcd%rp_0_0%0 pttcd%rp_0_0%0.pid is 12040
2018/01/24 15:58:50.430 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
pttcd%rp_0_0%0.gdbport is 9919
2018/01/24 15:58:50.430 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
pttcd%rp_0_0%0.swift_replport is 8019
2018/01/24 15:58:50.439 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  Launching
 pttcd  on fru rp slot 0
 bay 0 instance 0 log /tmp/rp/trace/pttcd_pmanlog
2018/01/24 15:58:50.439 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  Hold
failures 2, hold interval 1800
2018/01/24 15:58:50.439 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  PATH is
 /tmp/sw/rp/0/0/rp_daemons/

mount/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/conf:/tmp/sw/rp/0/0/


rp_daemons/mount/usr/binos/sbin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin:/tmp/sw/rp/0/0/rp_daemons/mount/


usr/cpp/bin:/usr/bin:/bin:/sbin:/usr/binos/conf:/usr/binos/bin:/sbin:/bin:/usr/bin:/usr/sbin:/usr/binos/conf:
```

```
 /sbin:/bin:/usr/bin:/usr/sbin:/usr/binos/conf
2018/01/24 15:58:50.439 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
LD_LIBRARY_PATH is
2018/01/24 15:58:50.441 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
PREPROC_OPTIONS ==
2018/01/24 15:58:50.441 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  command
 line used   pttcd  >>
 /tmp/rp/trace/pttcd_pmanlog_cmd 2&>1 &
2018/01/24 15:58:50.444 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  full_path
 is /tmp/sw/rp/0/0
 /rp_daemons/mount/usr/binos/bin/pttcd
2018/01/24 15:58:50.446 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  Resolved
 readlink process

/tmp/sw/mount/asr1000rpx86-rpcontrol.BLD_V168_THROTTLE_LATEST_20180122_164958_V16_8_0_177.SSA.pkg/usr/binos/bin/pttcd
2018/01/24 15:58:50.446 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  Full
path used to spawn the process:
 /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pttcd
2018/01/24 15:58:50.452 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  Binary_arch
 set to: [x86_64_cge7]
2018/01/24 15:58:50.461 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  actual
pttcd pid is 12542
2018/01/24 15:58:50.461 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  Checking
 for cgroup for PID 12542
2018/01/24 15:58:50.461 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
 /tmp/rp/pvp/process_state/pttcd%rp_0_0%0#12040_state marked up
2018/01/24 15:58:50.474 {pttcd_R0-0}{1}: [pttcd] [12542]: (ERR): init_callhome() failed
2018/01/24 15:58:50.475 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  oom score
 adj value is 399
2018/01/24 15:58:50.475 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  Wait for
 signal or process exit: 12542
2018/01/24 15:58:52.077 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  gdb port
9920 allocated
2018/01/24 15:58:52.085 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  swift_repl
 port 8020 allocated
2018/01/24 15:58:52.157 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  process
scoreboard /tmp/rp/process
 /pubd%rp_0_0%0 pubd%rp_0_0%0.pid is 14416
2018/01/24 15:58:52.157 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
pubd%rp_0_0%0.gdbport is 9920
2018/01/24 15:58:52.157 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
pubd%rp_0_0%0.swift_replport is 8020
2018/01/24 15:58:52.166 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  Launching
 pubd  on fru rp slot 0 bay 0
 instance 0 log /tmp/rp/trace/pubd_pmanlog
2018/01/24 15:58:52.166 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  Hold
failures 2, hold interval 1800
2018/01/24 15:58:52.166 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  PATH is
/tmp/sw/rp/0/0/rp_daemons

/mount/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/conf:/tmp/sw/rp/0/0


/rp_daemons/mount/usr/binos/sbin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr


/cpp/bin:/usr/bin:/bin:/sbin:/usr/binos/conf:/usr/binos/bin:/sbin:/bin:/usr/bin:/usr/sbin:/usr/binos/conf:/sbin:/bin:

 /usr/bin:/usr/sbin:/usr/binos/conf
2018/01/24 15:58:52.166 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
LD_LIBRARY_PATH is
2018/01/24 15:58:52.167 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
PREPROC_OPTIONS ==
```

```
2018/01/24 15:58:52.167 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  command
line used   pubd  >>
 /tmp/rp/trace/pubd_pmanlog_cmd 2&>1 &
2018/01/24 15:58:52.170 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  full_path
 is /tmp/sw/rp/0/0
 /rp_daemons/mount/usr/binos/bin/pubd
2018/01/24 15:58:52.172 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  Resolved
readlink process

/tmp/sw/mount/asr1000rpx86-rpcontrol.BLD_V168_THROTTLE_LATEST_20180122_164958_V16_8_0_177.SSA.pkg/usr/binos/bin/pubd
2018/01/24 15:58:52.172 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  Full path
 used to spawn the process:
 /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pubd
2018/01/24 15:58:52.177 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  Binary_arch
 set to: [x86_64_cge7]
2018/01/24 15:58:52.184 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  actual
pubd pid is 14920
2018/01/24 15:58:52.184 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  Checking
for cgroup for PID 14920
2018/01/24 15:58:52.184 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  Setting
cgroup iosxe_control_processes
 /iosxe_mgmt_processes for PID 14920 and PID 14416
2018/01/24 15:58:52.188 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
 /tmp/rp/pvp/process_state/pubd%rp_0_0%0#14416_state marked up
2018/01/24 15:58:52.193 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  oom score
 adj value is 399
2018/01/24 15:58:52.194 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  Wait for
signal or process exit: 14920
2018/01/24 15:58:52.540 {pttcd_R0-0}{1}: [pttcd] [12542]: (ERR): PPTCD_1_abcdefghi
transaction id = 1
2018/01/24 15:58:57.133 {syncfd_pmanlog_R0-0}{1}: [syncfd_pmanlog] [19542]: (note):  gdb
port 9922 allocated
2018/01/24 15:58:57.147 {syncfd_pmanlog_R0-0}{1}: [syncfd_pmanlog] [19542]: (note):
swift_repl port 8022 allocated
2018/01/24 15:58:57.296 {syncfd_pmanlog_R0-0}{1}: [syncfd_pmanlog] [19542]: (note):
 process scoreboard /tmp/rp/process/syncfd%rp_0_0%0 syncfd%rp_0_0%0.pid is 19470
```

monitor log profile restconf

To display debug logs for RESTCONF processes, use the **monitor log profile restconf** command in privileged EXEC mode.

monitor log profile netconf-yang internal

Syntax Description

internal	Displays all debug logs. **Note** This keyword is used by customer support.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.8.1	This command was introduced.

Usage Guidelines

Logs generated by this command are rendered on the device console.

Example

The following example shows how to enable the **monitor log profile restconf internal** command:

```
Device# monitor log profile restconf internal

Displaying traces starting from 2018/03/23 09:10:02.000.  If no traces are present, the
command will wait until one is.

2018/03/23 13:05:13.945 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  gdb port
 9908 allocated
2018/03/23 13:05:13.962 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  swift_repl
 port 8008 allocated
2018/03/23 13:05:14.050 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
  process scoreboard /tmp/rp/process/pttcd%rp_0_0%0 pttcd%rp_0_0%0.pid is 2550
2018/03/23 13:05:14.050 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
pttcd%rp_0_0%0.gdbport is 9908
2018/03/23 13:05:14.050 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
pttcd%rp_0_0%0.swift_replport is 8008
2018/03/23 13:05:14.060 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
 Launching pttcd  on fru rp slot 0 bay 0 instance 0 log /tmp/rp/trace/pttcd_pmanlog
2018/03/23 13:05:14.060 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Hold
failures 2, hold interval 1800
2018/03/23 13:05:14.060 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
 PATH is /tmp/sw/rp/0/0/rp_daemons/mount/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/bin:

/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/conf:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/sbin:

 /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/cpp/bin:


/usr/bin:/bin:/sbin:/usr/binos/conf:/usr/binos/bin:/sbin:/bin:/usr/bin:/usr/sbin:/usr/binos/conf:/sbin:/bin:

 /usr/bin:/usr/sbin:/usr/binos/conf
```

```
2018/03/23 13:05:14.060 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
LD_LIBRARY_PATH is
2018/03/23 13:05:14.063 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
PREPROC_OPTIONS ==
2018/03/23 13:05:14.063 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  command
line used   pttcd  >>
 /tmp/rp/trace/pttcd_pmanlog_cmd 2&>1 &
2018/03/23 13:05:14.068 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
 full_path is /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pttcd
2018/03/23 13:05:14.069 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
 Resolved readlink process /tmp/sw/mount/asr1000rpx86-rpcontrol.2018-03-07_18.30_rifu.SSA.pkg

 /usr/binos/bin/pttcd
2018/03/23 13:05:14.069 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Full path
 used to spawn the process:
 /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pttcd
2018/03/23 13:05:14.076 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Binary_arch
 set to: [x86_64_cge7]
2018/03/23 13:05:14.088 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  actual
pttcd pid is 2936
2018/03/23 13:05:14.088 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Checking
 for cgroup for PID 2936
2018/03/23 13:05:14.088 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
 /tmp/rp/pvp/process_state/pttcd%rp_0_0%0#2550_state marked up
2018/03/23 13:05:14.097 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  oom score
 adj value is 399
2018/03/23 13:05:14.102 {pttcd_R0-0}{1}: [pttcd] [2936]: (ERR): init_callhome() failed
2018/03/23 13:05:14.102 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Wait for
 signal or process exit: 2936
2018/03/23 13:05:16.895 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  gdb port
9920 allocated
2018/03/23 13:05:16.904 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  swift_repl
 port 8020 allocated
2018/03/23 13:05:16.987 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  process
scoreboard
 /tmp/rp/process/pubd%rp_0_0%0 pubd%rp_0_0%0.pid is 4922
2018/03/23 13:05:16.987 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
pubd%rp_0_0%0.gdbport is 9920
2018/03/23 13:05:16.987 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
pubd%rp_0_0%0.swift_replport is 8020
2018/03/23 13:05:16.997 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
 Launching pubd  on fru rp slot 0 bay 0 instance 0 log /tmp/rp/trace/pubd_pmanlog
2018/03/23 13:05:16.997 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  Hold failures
 2, hold interval 1800
2018/03/23 13:05:16.997 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  PATH is
 /tmp/sw/rp/0/0/rp_daemons/mount/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/bin:/tmp/sw/rp/0/0/


rp_daemons/mount/usr/binos/conf:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/sbin:/tmp/sw/rp/0/0/


rp_daemons/mount/usr/binos/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/cpp/bin:/usr/bin:/bin:/sbin:


/usr/binos/conf:/usr/binos/bin:/sbin:/bin:/usr/bin:/usr/sbin:/usr/binos/conf:/sbin:/bin:/usr/bin:

 /usr/sbin:/usr/binos/conf
2018/03/23 13:05:16.997 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
LD_LIBRARY_PATH is
2018/03/23 13:05:17.001 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
PREPROC_OPTIONS ==
2018/03/23 13:05:17.001 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  command
line used   pubd  >>
 /tmp/rp/trace/pubd_pmanlog_cmd 2&>1 &
```

```
2018/03/23 13:05:17.007 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
 full_path is /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pubd
2018/03/23 13:05:17.009 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  Resolved
readlink process
 /tmp/sw/mount/asr1000rpx86-rpcontrol.2018-03-07_18.30_rifu.SSA.pkg/usr/binos/bin/pubd
2018/03/23 13:05:17.009 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  Full path
used to spawn the process:
 /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pubd
2018/03/23 13:05:17.017 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  Binary_arch
 set to: [x86_64_cge7]
2018/03/23 13:05:17.031 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  actual pubd
 pid is 5303
2018/03/23 13:05:17.031 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  Checking
for cgroup for PID 5303
2018/03/23 13:05:17.031 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
 Setting cgroup iosxe_control_processes/iosxe_mgmt_processes for PID 5303 and PID 4922
2018/03/23 13:05:17.045 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
 /tmp/rp/pvp/process_state/pubd%rp_0_0%0#4922_state marked up
2018/03/23 13:05:17.047 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note): oom score
adj value is 399
```

multicast (App Hosting)

To enable multicast routing on an AppGigabitEthernet interface, use the **multicast** command in application-hosting VLAN-access IP configuration mode. To disable multicast routing, use the **no** form of this command.

multicast
no multicast

This command has no arguments or keywords.

Command Default

Multicast is not enabled.

Command Modes

Application-hosting VLAN-access IP configuration (config-config-app-hosting-vlan-access-ip)

Command History

Release	Modification
Cisco IOS XE Dublin 17.11.1	This command was introduced.

Usage Guidelines

Multicast traffic forwarding cannot be enabled on the management interface. However, when the management interface is used as an external AppGigabitEthernet port, multicast traffic forwarding can be enabled.

On some platforms, IGMP Snooping must be disabled for multicast forwarding to work.

Example

The following example shows how to enable multicast routing on an AppGigabitEthernet interface:

```
Device> enable
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-vnic AppGigabitEthernet trunk
Device(config-config-app-hosting-trunk)# vlan 1 guest-interface 9
Device(config-config-app-hosting-vlan-access-ip)# guest-ipaddress 192.168.0.2
netmask 255.255.255.0
Device(config-config-app-hosting-vlan-access-ip)# multicast
Device(config-config-app-hosting-vlan-access-ip)# end
Device#
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-vnic gateway	Configures a virtual network interface gateway.
app-vnic AppGigabitEthernet trunk	Configures a front-panel trunk port and enters application-hosting trunk configuration mode.
app-vnic management	Configures the management gateway of a virtual network interface.

Command	Description
guest-ipaddress (App Hosting)	Configure an IP address for a guest interface.
vlan (App Hosting)	Configures a VLAN guest interface and enters application-hosting VLAN-access IP configuration mode.

name-server (App Hosting)

To configure a Domain Name System (DNS) server, use the **name-server** command in application hosting configuration mode. To remove the DNS server configuration, use the **no** form of this command.

name-server*number ip-address*
no name-server*number* [*ip-address*]

Syntax Description

ip-address	IP address the of the DNS server.

Command Default

DNS server is not configured.

Command Modes

Application hosting configuration (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

While configuring a static IP address in a Linux container for application hosting, only the last configured name server configuration is used.

Example

The following example shows how to configure a DNS server for a virtual network interface gateway:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-vnic gateway1 VirtualPortGroup 0 guest-interface 1
Device(config-app-hosting-gateway1)# guest-ipaddress 10.0.0.3 netmask 255.255.255.0
Device(config-app-hosting-gateway1)# exit
Device(config-app-hosting)# name-server0 10.2.2.2
Device(config-app-hosting)# end
```

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-hosting gateway	Configures a virtual network interface gateway.
guest-ipaddress	Configures an IP address for the guest interface.

net-debug

To display or change the network debug values use the **net-debug** command in rommon mode.

net-debug [*new-value*]

Syntax Description

new-value	(Optional) New debug value to use.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

This command enables or disables log levels for each of the following functional areas:

- Domain Name System (DNS)
- Dynamic Host Control Protocol (DHCP)
- File Transfer Protocol (FTP)
- Hypertext Transfer Protocol (HTTP)
- IP
- TCP
- UDP
- Uniform Resource Identifier (URI)

Example

This following is sample output from the **net-debug** command:

```
Device: net-debug

 ether: 0
    ip: 0
  dhcp: 0
  udp:  0
  tcp: 0
 http: 0
  dns: 0
  uri: 0
t/ftp: 2
  ip6: 0
dhcp6: 0:000 200 000 000
```

Related Commands

Command	Description
mlog	Directs log messages to a memory buffer instead of the serial port.

net-dhcp

To initiate an IPv4 Dynamic Host Control Protocol (DHCP) request for remote configuration, use the **net-dhcp** command in rommon mode.

net-dhcp [**timeout**]

Syntax Description

timeout	(Optional) Timeout in seconds.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

This command initiates an IPv4 DHCP request and processes the reply.

Example

The following example shows how to enable the **net-dhcp** command:

```
Device: net-dhcp
```

Related Commands

Command	Description
net-debug	Displays or changes the network debug values.
net-show	Displays network parameters.
net6-dhcp	Initiates an IPv6 DHCP request for remote configuration.

net-show

To display network parameters, use the **net-show** command in rommon mode.

net-show

This command has no arguments or keywords.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

This command displays network configuration such as IP address, gateway, MAC address and so on.

Example

The following is sample output from the **net-show** command:

```
Device:  net-show
Network params:
IPv4:
         ip addr 10.29.27.150
         netmask 255.255.0.0
         gateway 10.29.0.1
IPv6:
link-local addr fe80::366f:90ff:feb8:cb80
site-local addr fec0::366f:90ff:feb8:cb80
       DHCP addr 2001:dead:beef:cafe::9999
     router addr fe80::7ada:6eff:fe13:8580
      SLAAC addr 2001:dead:beef:cafe:366f:90ff:feb8:cb80 /64
      SLAAC addr f00d::366f:90ff:feb8:cb80 /64
      SLAAC addr feed::366f:90ff:feb8:cb80 /64
Common:
         macaddr 34:6f:90:b8:cb:80
             dns 2001:dead:beef:cafe::5
        bootfile http://www.example.org/ed10m
          domain ip6.example.org
```

Command	Description
net6-show	Displays IPv6 network parameters.

net-tcp-bufs

To display TCP buffers, use the **net-tcp-bufs** command in rommon mode.

net-tcp-bufs [*mss*]

Syntax Description

mss	(Optional) The Maximum Segment Size (MSS) of TCP buffers.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

You can set the MSS of TCP buffers using the *mss* argument.

Example

The following is sample output from the **net-tcp-bufs** command:

```
Device: net tcp-bufs

tcp_num_buffs 4
```

Related Commands

Command	Description
net-tcp-mss	View or set the TCP MSS.

net-tcp-mss

To view or set the TCP Maximum Segment Size (MSS), use the **net-tcp-mss** command in rommon mode.

net-tcp-mss [*mss*]

Syntax Description

mss	(Optional) The Maximum Segment Size (MSS) of TCP buffers.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

Use the *mss* argument to change the MSS size.

Example

The following is sample output from the **net-tcp-mss** command:

```
Device: net-tcp-mss

switch: net-tcp-mss
tcp_segment_size 1024
```

The following is sample output from the **net-tcp-mss** *mss* command:

```
Device: net-tcp-mss 700

switch: net-tcp-mss 700
tcp_segment_size 700
```

Related Commands

Command	Description
net-tcp-bufs	Displays TCP buffers.

net6-dhcp

To initiate an IPv6 Dynamic Host Control Protocol (DHCP) request for remote configuration, use the **net6-dhcp** command in rommon mode.

net6-dhcp [**timeout**]

Syntax Description

timeout	(Optional) Timeout in seconds.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

You can change the timeout by specifying a time in seconds

Example

The following example shows how to enable the **net6-dhcp** command:

```
Device: net6-dhcp
```

Related Commands

Command	Description
net-debug	Displays or changes the network debug values.
net-dhcp	Initiates an IPv4 DHCP request and processes the reply.
net-show	Displays network parameters.

net6-show

To display IPv6 network parameters, use the **net6-show** command in rommon mode.

net6-show

This command has no arguments or keywords.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

Example

The following is sample output from the **net6-show** command:

```
Device: net6-show

switch: net6-show
IP6 addresses
link-local addr fe80::366f:90ff:feb8:cb80
site-local addr fec0::366f:90ff:feb8:cb80
       DHCP addr 2001:dead:beef:cafe::9999
     router addr fe80::7ada:6eff:fe13:8580
      SLAAC addr 2001:dead:beef:cafe:366f:90ff:feb8:cb80 /64
      SLAAC addr f00d::366f:90ff:feb8:cb80 /64
      SLAAC addr feed::366f:90ff:feb8:cb80 /64
--
       null addr ::
  all-nodes addr ff02::1
all-routers addr ff02::2
   all-dhcp addr ff02::1:2
  Slct-node addr ff02::1:ffb8:cb80
    ll mmac addr 33:33:00:00:00:01
    sl mmac addr 33:33:00:00:00:02
    sn mmac addr 33:33:ff:b8:cb:80
  dhcp mmac addr 33:33:ff:00:99:99
router mac addr 78:da:6e:13:85:80

IP6 neighbour table
0: ip6 fec0::366f:90ff:feb8:cb80 MAC 34:6f:90:b8:cb:80
1: ip6 fe80::366f:90ff:feb8:cb80 MAC 34:6f:90:b8:cb:80
2: ip6 fe80::7ada:6eff:fe13:8580 MAC 78:da:6e:13:85:80
3: ip6 2001:dead:beef:cafe::5 MAC 30:f7:0d:08:7e:bd
4: ip6 fe80::32f7:dff:fe08:7ebd MAC 30:f7:0d:08:7e:bd
```

Related Commands

Command	Description
net-show	Displays network parameters.

netconf detailed-error

To display helpful return codes if an invalid command is executed in a NETCONF session, use the **netconf detailed-error** command in global configuration mode. To stop displaying the return codes, use the **no** form of this command.

netconf detailed-error
no netconf detailed-error

This command has no arguments or keywords.

Command Default

NETCONF does not send return codes for invalid command execution.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

The **netconf detailed-error** command configures NETCONF to send a "NOT OK" return code if you attempt to execute an invalid command.

For **show** commands, the return code appears in this form:

```
<return-code>NOT OK</return-code>
```

For configuration commands, the return code includes the line number of the invalid command. This example includes the request and the response, to illustrate:

```
Request:-
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<edit-config>
<target>
<running/>
</target>
<config>
<cli-config-data>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>interface nve 1</cmd>
<cmd>member vni 5005</cmd>
<cmd>ingress-replication 10.1.1.1</cmd>
```

```
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
<cmd>hostname sample-host1</cmd>
</cli-config-data>
</config>
</edit-config>
</rpc>]]>]]>

Response:-
<?xml version="1.0" encoding="UTF-8"?><rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"><rpc-error>
<error-type>protocol</error-type><error-tag>operation-failed</error-tag>
<error-severity>error</error-severity><error-message>
**CLI Line # 20: % VNI 5005 already exists on other nve
interface</error-message></rpc-error></rpc-reply>]]>]]>
```



Note For a series of commands provided in an input XML:

- If NETCONF attempts to execute a series of **show** commands and it encounters an invalid command, NETCONF does not stop execution. It continues to execute other commands in the input XML, and provides the error return code(s) for invalid commands in the output.
- If NETCONF attempts to execute a series of **configuration** commands and it encounters an invalid command, NETCONF stops execution. It provides the error return code for the invalid command, including line number, in the output.

Examples

Enabling detailed error reporting on a device:

```
Device(config)# netconf detailed-error
```

Related Commands

Command	Description
netconf beep initiator	Configures BEEP as the transport protocol for NETCONF and configures a peer as the BEEP initiator.
netconf beep listener	Configures BEEP as the transport protocol for NETCONF and configures a peer as the BEEP listener.
netconf format	Associates NETCONF with an ODM spec file for XML-formatted requests.
netconf lock-time	Specifies the maximum time a NETCONF configuration lock is in place without an intermediate operation.
netconf max-sessions	Specifies the maximum number of concurrent NETCONF sessions allowed.
netconf ssh	Enables NETCONF over SSHv2.

netconf legacy

To enable legacy NETCONF protocol, use the **netconf legacy** command in global configuration mode. To disable the legacy NETCONF protoocol, use the **no** form of this command.

netconf legacy
no netconf legacy

This command has no arguments or keywords.

Command Default

Legacy NETCONF protocol is not enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Denali 16.3.1	This command was introduced.

Usage Guidelines

If this command is enabled, the RFC-compliant NETCONF client (ncclient) does not work. This command enables the legacy NETCONF protocol that is non-RFC-compliant.

Example

The following example shows how to disable the legacy NETCONF protocol:

```
Device> enable
Devcie# configure terminal
Device(config)# no netconf legacy
```

netconf-yang feature candidate-datasource

To enable the candidate datasource functionality, use the **netconf-yang feature candidate-datasource** command in global configuration mode. To disable the feature, use the **no** form of this command.

netconf-yang feature candidate-datasource
no netconf-yang feature candidate-datasource

Syntax Description

This command has no arguments or keywords.

Command Default

Candidate datasource is not enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.1	This command was introduced.

Usage Guidelines

Use the **netconf-yang feature candidate-datastore** command to enable the candidate datastore functionality. When the datastore state changes from running to candidate or back, a warning message is displayed notifying the user that a restart of NETCONF-YANG or RESTCONF will occur in order for the change to take effect. When candidate is enabled, The running data store is not writable through NETCONF sessions, all configurations get committed only through candidate. In other words, the writable-running NETCONF capability is not enabled with candidate.



Note Candidate data store is a shared data store, that is, multiple NETCONF sessions can modify the contents simultaneously. Therefore, it is important for a user to lock the data store before modifying its contents, to prevent conflicting commits which can eventually lead to losing any configuration changes; wherein another user overwrites the configuration by modifying the configuration and issuing a commit.

The following example shows how to enable the feature. If the selection of candidate or running datastore, is specified in the configuration when a NETCONF-YANG or RESTCONF confd process starts, a warning appears:

```
Device(config)# netconf-yang feature candidate-datastore

netconf-yang initialization in progress - datastore transition not allowed, please try again
 after 30 seconds
```

If the selection of candidate or running is made after NETCONF-YANG or RESTCONF confd process starts, the following apply:

- If the **netconf-yang feature candidate-datastore** command is configured, the command enables the candidate datastore and prints the following warning:

  ```
  "netconf-yang and/or restconf is transitioning from running to candidate netconf-yang
  and/or
   restconf will now be restarted, and any sessions in progress will be terminated".
  ```

- If the **netconf-yang feature candidate-datastore** command is removed, the command disables the "candidate" datastore, enables the "running" datastore and prints the following warning:

  ```
  "netconf-yang and/or restconf is transitioning from candidate to running netconf-yang
  and/or
  restconf will now be restarted, and any sessions in progress will be terminated".
  ```

- When NETCONF-YANG or RESTCONF are restarted, sessions in progress will be lost.

netconf-yang feature side-effect-sync

To enable the partial synchronization NETCONF database, use the **netconf-yang feature side-effect-sync** command in global configuration mode. To disable the partial synchronization, use the **no** form of this command.

netconf-yang feature side-effect-sync
no netconf-yang feature side-effect-sync

This command has no arguments or keywords.

Command Default

Disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.4.1	This command was introduced.

Usage Guidelines

During configuration changes in the data model interface (DMI), a partial synchronization of the changes that are triggered when a command or RPC is configured happens. This is called the side-effect synchronization, and it reduces the synchronization time and NETCONF downtime.

Some commands, when they are configured, triggers changes in some already configured commands. For example, the following is the configuration on a device before the NETCONF edit-config RPC is configured:

```
hostname device123
```

The NETCONF edit-config RPC:

```
<native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
 <hostname xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:operation="delete"/>
</native>
```

The following is the configuration on the device after the NETCONF edit-config RPC is configured:

```
hostname Switch
```

Example

The following example shows how to enable the **netconf-yang feature side-effect-sync** command:

```
Device> enable
Device# configure terminal
Device(config)# netconf-yang feature side-effect-sync
```

netconf-yang ssh

To configure Secure Shell (SSH) options for a NETCONF-YANG session, use the **netconf-yang ssh** command in global configuration mode. To remove the SSH configuration, use the **no** form of this command.

netconf-yang ssh { {**ipv4** | **ipv6**}**access-list name** *access-list-name* | **port** *port-number*}
no netconf-yang ssh { {**ipv4** | **ipv6** }**access-list** [**name** *access-list-name*] | **port** *port-number*}

Syntax Description

ipv4	Specifies the IP access-list configuration parameters.
ipv6	Specifies the IPv6 access-list configuration parameters.
access-list *name*	Configures the NETCONF-YANG SSH service to use for a named IP or IPv6 ACL.
port *port-number*	Specifies the port number to listen on. Valid values for the *port-number* argument are from 1 to 65535.

Command Default

Client connections are allowed.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.11.1	This command was introduced.

Usage Guidelines

Clients that do not conform to the configured ACL are not allowed to connect to the network. You can use an access-list name that is not defined.

Example

The following example shows how to configure an IPv4 ACL for a NETCONF-YANG session.:

```
Device# configure terminal
Device(config)# netconf-yang ssh ipv4 access-list ipv4-acl
Device (config)#
```

The following example shows how to configure an IPv6 ACL for a NETCONF-YANG session:

```
Device# configure terminal
Device(config)# netconf-yang ssh ipv6 access-list ipv6-acl
Device (config)#
```

The following example shows how to configure the port number to listen on for a NETCONF-YANG session:

```
Device# configure terminal
Device(config)# netconf-yang ssh port 5
Device (config)#
```

The following example shows how to define an IP access list and associate it with a NETCONF-YANG session:

```
Device# configure terminal
Device(config)# ip access-list standard acl1_permit
Device(config-std-nacl)# permit 192.168.255.0 0.0.0.255
Device(config-std-nacl)# deny any
Device(config-std-nacl)# exit
Device(config)# netconf-yang ssh ipv4 access-list name acl1_permit
Device(config)# end
```

Related Commands

Command	Description
deny	Sets conditions in an IP/IPv6 access list that will deny packets.
ip access-list	Defines a standard IP access list and enters standard access-list configuration mode.
ipv6 access-list	Defines an IPv6 access list and enters IPv6 access list configuration mode.
permit	Sets conditions in an IP/IPv6 access list that will permit packets.

netconf-yang ssh local-vrf guestshell

To enable NETCONF-YANG access through an SSH connection from within the Guest Shell, use the **netconf-yang ssh local-vrf guestshell** command in global configuration mode. To disable the NETCONF-YANG access, use the **no** form of this command.

netconf-yang ssh local-vrf guestshell *port-number*
no netconf-yang ssh local-vrf guestshell *port-number*

Syntax Description

port-number	The port number for NETCONF access.

Command Default

NETCONF access from Guest Shell is disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Usage Guidelines

To enable NETCONF-YANG access from within the Guest Shell, you must run the following commands in the Guest Shell prompt:

- **iosp_client -f netconf_enable guestshell** *port-number*
- **iosp_client -f netconf_enable_passwordless guestshell** *username*

The **iosp_client -f netconf_enable guestshell** *port-number* command configures the **netconf-yang ssh local-vrf guestshell** command, and blocks connections until NETCONF-YANG is available. The **iosp_client -f netconf_enable_passwordless guestshell** *username* command generates the SSH keys for Guest Shell access.

Example

The following example shows how to enable NETCONF-YANG access through the Guest Shell:

```
Device> enable
Device# configure terminal
Device(config)# netconf-yang ssh local-vrf guestshell 803
```

netconf-yang ssh port disable

To disable all external connectivity for NETCONF-YANG, use the **netconf-yang ssh port disable** command in global configuration mode.

netconf-yang ssh port disable

This command has no arguments or keywords.

Command Default

External ports are enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Usage Guidelines

This command closes external ports, only internal connections, such as the ones used for Guest Shell, remain open.

Example

The following example shows how to disable external connections for NETCONF-YANG:

```
Device> enable
Device# configure terminal
Device(config)# netconf-yang ssh port-disable
```

netconf-yang ssh server algorithm encryption

To enable the encryption algorithms that are advertised to a third party, use the **netconf-yang ssh server algorithm encryption** command in global configuration mode. To disable the encryption algorithms, use the **no** form of this command.

netconf-yang ssh server algorithm encryption { **aes128-cbc** | **aes128-ctr** | **aes192-ctr** | **aes256-cbc** | **aes256-ctr** }
no netconf-yang ssh server algorithm encryption { **aes128-cbc** | **aes128-ctr** | **aes192-ctr** | **aes256-cbc** | **aes256-ctr** }

Syntax Description

aes128-cbc	Enables Advanced Encryption Standard (AES) with 128 bit key in Cipher Block Chaining (CBC) mode.
aes128-ctr	Enables AES with 128 bit key in Counter (CTR) mode.
aes192-ctr	Enables AES with 128 bit key in CTR mode.
aes256-cbc	Enables AES with 128 bit key in CBC mode.
aes256-ctr	Enables AES with 128 bit key in CTR mode.

Command Default

Encryption algorithms are enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Dublin 17.12.1	This command was introduced.

Usage Guidelines

AES supports three key sizes: 128 bits, 192 bits, and 256 bits. The default key size is 128 bits, and all implementations must support this key size.

Example

The following example shows how to enable the aes-192-ctr encryption algorithm:

```
Device> enable
Device# configure terminal
Device(config)# netconf-yang ssh server algorithm encryption aes192-ctr
```

Related Commands

Command	Description
netconf-yang ssh server algorithm hostkey	Enables the hostkey algorithms that are advertised to a third party.
netconf-yang ssh server algorithm kex	Enables the KEX algorithms that are advertised to a third party.

Command	Description
netconf-yang ssh server algorithm mac	Enables the MAC algorithms that are advertised to a third party.

netconf-yang ssh server algorithm hostkey

To enable the hostkey algorithms that are advertised to a third party, use the **netconf-yang ssh server algorithm hostkey** command in global configuration mode. To disable the hostkey algorithms, use the **no** form of this command.

netconf-yang ssh server algorithm hostkey { **rsa-sha2-256** | **rsa-sha2-512** | **ssh-rsa** }
netconf-yang ssh server algorithm hostkey { **rsa-sha2-256** | **rsa-sha2-512** | **ssh-rsa** }

Syntax Description

rsa-sha2-256	Enables Rivet, Shamir-Adelman (RSA) sha2-256 as the public key-based authentication algorithm.
rsa-sha2-512	Enables RSA sha2-512 as the public key-based authentication algorithm.
ssh-rsa	Enables SSH-RSA as the public key-based authentication algorithm.

Command Default

Hostkey algorithms are enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Dublin 17.12.1	This command was introduced.

Usage Guidelines

The **ssh-rsa** keyword is not supported in Federal Information Processing Standard (FIPS) mode.

Example

The following example shows how to configure the SSH-RSA hostkey:

```
Device> enable
Device# configure terminal
Device(config)# netconf-yang ssh server algorithm hostkey ssh-rsa
```

Related Commands

Command	Description
netconf-yang ssh server algorithm encryption	Enables the encryption algorithms that are advertised to a third party.
netconf-yang ssh server algorithm kex	Enables the KEX algorithms that are advertised to a third party.
netconf-yang ssh server algorithm mac	Enables the MAC algorithms that are advertised to a third party.

netconf-yang ssh server algorithm kex

To enable the key exchange (KEX) algorithms that are advertised to a third party, use the **netconf-yang ssh server algorithm kex** command in global configuration mode. To disable the KEX algorithms, use the **no** form of this command.

netconf-yang ssh server algorithm kex { **diffie-hellman-group14-sha1** | **diffie-hellman-group14-sha256** | **diffie-hellman-group16-sha512** | **ecdh-sha2-nistp256** | **ecdh-sha2-nistp384** | **ecdh-sha2-nistp521** }
no netconf-yang ssh server algorithm kex { **diffie-hellman-group14-sha1** | **diffie-hellman-group14-sha256** | **diffie-hellman-group16-sha512** | **ecdh-sha2-nistp256** | **ecdh-sha2-nistp384** | **ecdh-sha2-nistp521** }

Syntax Description

diffie-hellman-group14-sha1	Enables Diffie-Hellman (DH) group14-sha1 as the KEX algorithm.
diffie-hellman-group14-sha256	Enables DH group14-sha256 as the KEX algorithm.
diffie-hellman-group16-sha512	Enables DH group16-sha512 as the KEX algorithm.
ecdh-sha2-nistp256	Enables ecdh-sha2-nistp256 as the KEX algorithm.
ecdh-sha2-nistp384	Enables ecdh-sha2-nistp384 as the KEX algorithm.
ecdh-sha2-nistp521	Enables ecdh-sha2-nistp521 as the KEX algorithm.

Command Default

KEX algorithms are enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Dublin 17.12.1	This command was introduced.

Example

The following example shows how to enable the ecdh-sha2-nistp521 KEX algorithm:

```
Device> enable
Device# configure terminal
Device(config)# netconf-yang ssh server algorithm kex ecdh-sha2-nistp521
```

Related Commands

Command	Description
netconf-yang ssh server algorithm encryption	Enables the encryption algorithms that are advertised to a third party.
netconf-yang ssh server algorithm hostkey	Enables the hostkey algorithms that are advertised to a third party.
netconf-yang ssh server algorithm mac	Enables the MAC algorithms that are advertised to a third party.

netconf-yang ssh server algorithm mac

To enable the message authentication code (MAC) algorithms that are advertised to a third party, use the **netconf-yang ssh server algorithm mac** command in global configuration mode. To disable the MAC algorithms, use the **no** form of this command.

netconf-yang ssh server algorithm mac { **hmac-sha1** | **hmac-sha2-256** | **hmac-sha2512** }
no netconf-yang ssh server algorithm mac { **hmac-sha1** | **hmac-sha2-256** | **hmac-sha2512** }

Syntax Description

hmac-sha1	Enables hash-based message authentication code (HMAC) sha1 as the MAC algorithm. Both the digest length and key length should be 160 bits.
hmac-sha2-256	Enables HMAC sha2-256 as the MAC algorithm. Both digest length and key length should be 256 bits.
hmac-sha2512	Enables HMAC sha2512 as the MAC algorithm. Both digest length and key length should be 512 bits.

Command Default

All MAC algorithms are enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Dublin 17.12.1	This command was introduced.

Example

The following example shows how to enable hmac-sha2512 algorithm:

```
Device> enable
Device# configure terminal
Device(config)# netconf-yang ssh server algorithm mac hmac-sha2512
```

Related Commands

Command	Description
netconf-yang ssh server algorithm encryption	Enables the encryption algorithms that are advertised to a third party.
netconf-yang ssh server algorithm hostkey	Enables the hostkey algorithms that are advertised to a third party.
netconf-yang ssh server algorithm kex	Enables the KEX algorithms that are advertised to a third party.

netconf-yang ssh trustpoint

To configure Public Key Infrastructure (PKI) trustpoints for NETCONF-YANG, use the **netconf-yang ssh trustpoint** command in global configuration mode.

netconf-yang ssh trustpoint *trustpoint-name*
no netconf-yang ssh trustpoint [*trustpoint-name*]

Syntax Description

trustpoint-name	Trustpoint name.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE 17.16.1	This command was introduced.

Usage Guidelines

To view the configured PKI trustpoints, use the **show netconf-yang ssh trustpoint** command.

Example

The following example shows how to configure PKI trustpoints:

```
Device> enable
Device# configure terminal
Device(config)# netconf-yang ssh trustpoint t1
```

persist-disk (App Hosting)

To reserve persistent disk space for an application, use the **persist-disk** command in configuration mode. To remove the reserved space, use the **no** form of this command.

persist-disk *unit*

no persist-disk [*unit*]

Syntax Description

unit	Persistent disk reservation in MB. Valid values are from 0 to 65535.

Command Default

If the command is not configured, the storage size is determined based on the application requirement.

Command Modes

Custom application resource profile configuration (config-app-resource-profile-custom)

Command History

Release	Modification
Cisco IOS XE Cupertino 17.9.1	This command was introduced in a release prior to Cisco IOS XE Cupertino 17.9.1.

Example

The following example shows how to reserve :

```
Device# configure terminal
Device(config)# app-hosting appid lxc_app
Device(config-app-hosting)# app-resource profile custom
Device(config-app-resource-profile-custom)# persist-disk 1
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-resource profile	Overrides the application-provided resource profile.

ping

To diagnose basic network connectivity, use the **ping** command in rommon mode.

ping [*host_ip_address*] [*retries*]

Syntax Description

host_ip_address	(Optional) IP address of the host.
retries	(Optional) Number of retries.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

The **ping** and **ping4** commands are the same.

The **ping** command is a very common method for troubleshooting the accessibility of devices

A timeout is implemented at the bootloader device prompt, that allows the bootloader to poll the TCP stack every 200 ms. As a result, the bootloader may take up to 200 ms to respond to pings. However, when the bootloader is downloading a file, and thus actively polling for new packets, it responds to ping quickly.

Example

The following is sample output from the **ping** command:

```
Device: ping 10.29.27.5

Ping 10.29.27.5 with 32 bytes of data ...
Host 10.29.27.5 is alive.
```

The following is sample output from the **ping** *host_ip_address retries* command:

```
Device: ping 10 6.29.27.5 6

Ping 10.29.27.5 with 32 bytes of data ... reply received in 0 ms
Ping 10.29.27.5 with 32 bytes of data ... reply received in 0 ms
Ping 10.29.27.5 with 32 bytes of data ... reply received in 0 ms
Ping 10.29.27.5 with 32 bytes of data ... reply received in 1 ms
Ping 10.29.27.5 with 32 bytes of data ... reply received in 0 ms
Ping 10.29.27.5 with 32 bytes of data ... reply received in 0 ms
```

Related Commands

Command	Description
ping4	Diagnoses basic network connectivity.
ping6	Determines the network connectivity to another device using IPv6 addressing.

ping4

To diagnose basic network connectivity, use the **ping4** command in rommon mode.

ping4 [*host_ip_address*][*retries*]

Syntax Description

host_ip_address	(Optional) IP address of the host to be pinged.
retries	(Optional) Number of retries.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

The **ping** and **ping4** commands are the same

A timeout is implemented at the bootloader device prompt, that allows the bootloader to poll the TCP stack every 200 ms. As a result, the bootloader may take up to 200 ms to respond to pings. However, when the bootloader is downloading a file, and thus actively polling for new packets, it responds to ping quickly.

Example

The following is sample output from the **ping4** *host_ip_address* command:

```
Device: ping4 10.29.27.5

Ping 10.29.27.5 with 32 bytes of data ...
Host 10.29.27.5 is alive.
```

Related Commands

Command	Description
ping	Diagnoses basic network connectivity.
ping6	Determines the network connectivity to another device using IPv6 addressing.

ping6

To determine the network connectivity to another device using IPv6 addressing, use the **ping6** command in rommon mode.

ping6 [*host*] [*repeats*] [*len*]

Syntax Description

host	(Optional) IP address of the host to be pinged.
repeats	(Optional) Number of times to repeat the ping.

Command Modes

Rommon

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.

Usage Guidelines

A timeout is implemented at the bootloader device prompt, that allows the bootloader to poll the TCP stack every 200 ms. As a result, the bootloader may take up to 200 ms to respond to pings. However, when the bootloader is downloading a file, and thus actively polling for new packets, it responds to ping quickly.

Example

The following is sample output from the **ping6** *host retries len* command:

```
Device: ping6 2001:DB8::1 6 1000

Ping host 2001:DB8::1, 6 times, 1000 bytes
Pinging 2001:DB8::1 ... reply in 0 ms
Pinging 2001:DB8::1 ... reply in 1 ms
Pinging 2001:DB8::1 ... reply in 1 ms
Pinging 2001:DB8::1 ... reply in 0 ms
Pinging 2001:DB8::1 ... reply in 0 ms
Pinging 2001:DB8::1 ... reply in 0 ms
```

Related Commands

Command	Description
ping	Diagnoses basic network connectivity.
ping4	Diagnoses basic network connectivity.

prepend-pkg-opts

To merge the package options with the Docker runtime options, use the **prepend-pkg-opts** command in application-hosting docker configuration mode. To stop the merge, use the **no** form of this command.

prepend-pkg-opts
no prepend-pkg-opts

This command has no arguments or keywords.

Command Default

Package options are not merged with runtime options.

Command Modes

Application-hosting docker configuration mode (config-app-hosting-docker)

Command History

Release	Modification
Cisco IOS XE Amsterdam 17.3.3	This command was introduced.

Usage Guidelines

If the same variable is available in both package and runtime options, it is overwritten.

Example

The following example shows how to configure runtime options:

```
Device> enable
Device# configure terminal
Device(config)# app-hosting appid 1keyes
Device(config-app-hosting)# app-resource docker
Device(config-app-hosting-docker)# prepend-pkg-opts
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-resource docker	Enables the configuration of runtime Docker options.

protocol

To specify a protocol for the named receiver, use the **protocol** command in telemetry protocol-receiver configuration mode. To remove the specified protocol, use the **no** form of this command.

protocol { **cloud-native** | **cntp-tcp** | **cntp-tls profile** *profile-name* | **grpc-tcp** | **grpc-tls profile** *profile-name* | **native** | **tls-native profile** *profile-name* }
no protocol { **cloud-native** | **cntp-tcp** | **cntp-tls profile** *profile-name* | **grpc-tcp** | **grpc-tls profile** *profile-name* | **native** | **tls-native profile** *profile-name* }

Syntax Description

cloud-native	Specifies the Native Cloud protocol.
cntp-tcp	Specifies the Civil Network Time Protocol (CNTP) TCP protocol.
cntp-tls	Specifies the CNTP Transport Layer Security (TLS) protocol.
grpc-tcp	Specifies the Google Remote Procedure Call (gRPC) TCP protocol.
grpc-tls	Specifies the gRPC TLS protocol.
profile *profile-name*	Specifies the profile name for the connection.
native	Specifies the Native protocol.
tls-native	Specifies the Native-TLS protocol.

Command Default

A protocol is not configured.

Command Modes

Telemetry protocol-receiver configuration (config-mdt-protocol-receiver)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Example

The following example shows how to configure a protocol for the named receiver:

```
Device> enable
Device# configure terminal
Device(config)# telemetry receiver protocol receiver1
Device(config-mdt-protocol-receiver)# protocol grpc-tcp
```

Related Commands

Command	Description
host	Specifies named receiver host details.
telemetry receiver protocol	Configures a named protocol receiver.

receiver

To configure a receiver to receive update notifications, use the **receiver** command in telemetry-subscription configuration mode. To disable the configuration, use the **no** form of this command.

receiver ip address { *ipv4-address ipv6-address* } *port* **protocol** *protocol*
no receiver ip address { *ipv4-address ipv6-address* } *port* **protocol** *protocol*

Syntax Description

ip address	Configures the receiver IP address.
ipv4-address ipv6-address	IPv4 or IPv6 receiver address.
port	Configures a receiver port.
protocol *protocol*	Configures a protocol for notification. The following protocols are supported: • **cloud-native** • **cntp-tcp** • **cntp-tls profile** *profile-name* • **grpc-tcp** • **grpc-tls profile** *profile-name* • **native** • **tls-native profile** *profile-name*

Command Modes

Telemetry-subscription configuration (config-mdt-subs)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.10.1	This command was introduced.
Cisco IOS XE Bengaluru 17.6.1	This command was modified. The following keywords and arguments were added: **cloud-native**, **cntp-tcp**, **cntp-tls**, **grpc-tcp**, **grpc-tls**, **native tls-native**, **profile**, and *profile-name*.

Usage Guidelines

A receiver is a network element that receives telemetry data. Configured subscriptions can be configured with multiple receivers, however; only the first valid receiver is used. If the first valid receiver is deleted, another receiver is connected.

Example

The following example shows how to configure receiver information for receiving notifications:

```
Device> enable
Device# configure terminal
```

```
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# receiver ip address 10.28.35.45 57555 protocol grpc-tcp
```

Related Commands

Command	Description
telemetry ietf subscription	Configures telemetry subscription.
receiver name	Configures a named receiver for a subscription.

receiver name

To configure a named receiver for a subscription, use the **receiver name** command in telemetry-subscription configuration mode. To remove the named receiver, use the **no** form of this command.

receiver name *receiver-name*

no receiver name *receiver-name*

Syntax Description

receiver-name	Host name of the receiver.

Command Default

A named receiver is not configured.

Command Modes

Telemetry subscription configuration (config-mdt-subs)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Usage Guidelines

To use a named receiver in a subscription, both the receiver type and the receiver name must be specified. You can also configure a named receiver through the YANG model.

Example

The following example shows how to configure a named receiver for a subscription:

```
Device> enable
Device# configure terminal
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# receiver type protocol
Device(config-mdt-subs)# receiver name my-receiver
```

Related Commands

Command	Description
receiver	Configures a receiver to receive update notifications.
show telemetry receiver	Displays the state of all telemetry receivers.
telemetry ietf subscription	Configures telemetry subscription.

receiver-type protocol

To configure a protocol-type named receiver, use the **receiver-type protocol** command in telemetry-subscription configuration mode. To remove the protocol-type named receiver, use the **no** form of this command.

receiver-type protocol
no receiver-type protocol

This command has no arguments or keywords.

Command Default

Protocol-type named receiver is not configured.

Command Modes

Telemetry-subscription configuration (config-mdt-subs)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Usage Guidelines

Protocols are the only type of named receivers supported. For legacy receivers, the value is the default rcvr-type-unspecified.

Example

The following example shows how to configure a protocol-type named receiver:

```
Device> enable
Device> configure terminal
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# receiver-type protocol
```

Related Commands

Command	Description
telemetry ietf subscription	Configures telemetry subscription.

resource profile

To override the application-provided resource profile, use the **resource profile** command in application hosting configuration mode. To revert to the application-specified resource profile, use the **no** form of this command.

resource profile *profile-name* [**cpu** *number* **memory** *memory* **vcpu** *number*]
no resource [**profile** *profile-name*]

Syntax Description

profile-name	Application profile name.
cpu *number*	Specifies the application CPU quota. Valid values are from 0 to 20000.
memory *memory*	Specifies the memory allocation in MB. Valid values are from 0 to 4096.
vcpu *number*	Specifies the application virtual CPU (vCPU) count. Valid values are from 0 to 65535.

Command Modes

Application hosting configuration (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 1612.1	This command was introduced.

Usage Guidelines

Within each application package, an application-specific resource profile is provided that defines the recommended CPU load, memory size, and number of vCPUs required for the application. Use this command to change the allocation of resources for specific processes in the custom resource profile.

Reserved resources specified in the application package can be changed by setting a custom resource profile. Only the CPU, memory, and vCPU resources can be changed. For the resource changes to take effect, stop and deactivate the application, then activate it and start it again.

Note Resource values are application-specific, and any adjustment to these values must ensure that the application can run reliably with the changes.

Example

The following example shows how to change the allocation of resources of an application:

```
Device# configure terminal
Device(config)# application-hosting appid iox_app
Device(config-app-hosting)# resource profile custom cpu 7400 memory 2048 vcpu 2
```

Related Commands

Command	Description
app-hosting	Initializes application hosting.
app-hosting appid	Enables application hosting and enters application hosting configuration mode.

restconf access-list

To configure an access control list (ACL) for a RESTCONF session, use the **restconf access-list** command in global configuration mode. To remove the ACL, use the **no** form of this command.

restconf [**ipv4** | **ipv6**]**access-list name** *access-list-name*
no restconf [**ipv4** | **ipv6**]**access-list** [**name** *access-list-name*]

Syntax Description

ipv4	(Optional) Specifies RESTCONF IPv4 configuration parameters.
ipv6	(Optional) Specifies RESTCONF IPv6 configuration parameters.
name	(Optional) Access-list name.

Command Default

Clients connections are allowed.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.11.1	This command was introduced.

Usage Guidelines

Clients that do not conform to the configured ACL are not allowed to connect to the network. You can use an access-list name that is not defined.

Example

The following example shows how to configure an IPv4 ACL for a RESTCONF session.:

```
Device# configure terminal
Device(config)# ip access-list standard ipv4_acl1_permit
Device(config-std-nacl)# permit 192.168.255.0 0.0.0.255
Device(config-std-nacl)# deny any
Device(config-std-nacl)# exit
Device(config)# restconf ipv4 access-list name ipv4_acl1_permit
Device(config)# end
```

The following example shows how to configure an IPv6 ACL for a RESTCONF session:

```
Device# configure terminal
Device(config)# ip access-list standard ipv6_acl1_permit
Device(config-std-nacl)# permit ipv6 2001:db8::1/32 any
Device(config-std-nacl)# deny any any
Device(config-std-nacl)# exit
Device(config)# restconf ipv6 access-list name ipv6_acl1_permit
Device(config)# end
```

Related Commands

Command	Description
deny	Sets conditions in an IP/IPv6 access list that will deny packets.
ip access-list	Defines a standard IP access list and enters standard access-list configuration mode.
ipv6 access-list	Defines an IPv6 access list and enters IPv6 access list configuration mode.
permit	Sets conditions in an IP/IPv6 access list that will permit packets.

request platform software yang-management nacm

To request platform software actions for the YANG management Network Configuration Access Control Module (NACM), use the **request platform software yang-management nacm** command in privileged EXEC mode.

request platform software yang-management nacm { **populate-read-rules privilege** *privilege-level* | **reset-config** [**switch** { *switch-number* { **active** } | **active** | **standby** } { **R0** | **RP** { **active** } }] }

Syntax Description

populate-read-rules	Populates the read-only rules.
privilege *privilege-level*	Specifies the user privilege levels. Valid values for the *privilege-level* argument is from 0 to 14.
reset-config	Resets the local NACM configuration.
switch *switch-number*	Specifies the switch number. Valid values for the *switch-number* argument are 1 and 2.
active	Specifies the active instance of the switch.
standby	Specifies the standby instance of the switch.
R0	Specifies slot 0 of the Route Processor (RP).
RP	Specifies the RP.

Command Default

The **reset-config** keyword is enabled by default.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Cupertino 17.9.1	This command was introduced.

Usage Guidelines

The NACM configuration is persistent across reloads. The NACM rules can be read by using the GET or READ operation against the NACM XPath. And the NACM rules can be reset by using the **reset-config** keyword.

The **populate-read-rules** keyword can be used to enable read-only operations for privilege level 1.

Example

The following example shows how to populate NACM with read-only rules:

```
Device> enable
Device# request platform software yang-management nacm populate-read-rules privilege 1
```

The following example shows how to reset the local configuration access control module:

```
Device> enable
Device# request platform software yang-management nacm reset-config
```

run-opts

To specify or change the runtime Docker options, use the **run-opts** command in application-hosting docker configuration mode. To remove the runtime Docker options, use the **no** form of this command.

run-opts *options*
no run-opts *options*

Syntax Description

options	Runtime Docker options.

Command Default

Runtime options are not configured.

Command Modes

Application-hosting docker configuration mode (config-app-hosting-docker)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

You can add a maximum of 30 lines of runtime options. The system generates a concatenated string from line 1 though line 30. Each line can have a maximum of 235 characters. A string can have more than one Docker runtime option.

When a runtime option is changed, you need to stop, deactivate, activate, and start the application again for the new runtime options to take effect.

Example

The following example shows how to configure runtime options:

```
Device> enable
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# app-resource docker
Device(config-app-hosting-docker)# run-opts 1 "-v $(APP_DATA):/data"
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-resource docker	Enables the configuration of runtime Docker options.

show app-hosting

To display application hosting-related information, use the **show app-hosting** command in privileged EXEC mode.

show app-hosting {**detail** [**appid** *name*] | **infra** | **list** | **resource** | **utilization appid** *name*}

Syntax Description

detail	Displays detailed information about the application.
appid *name*	Displays detailed information about the specified application.
infra	Displays infrastructure details about the application hosting framework.
list	Displays information about the application or appliance.
resource	Displays the available resources.
utilization	Displays resource utilization information about the application/appliance.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.12.1	This command was introduced.

Example

The following is sample output from the **show app-hosting detail** command:

```
Device# show app-hosting detail

App id                 : perfsonar
Owner                  : iox
State                  : RUNNING
Application
  Type                 : lxc
  Name                 : perfsonar-lxc
  Version              : 1.0.0
  Description          : PerfSONAR 4.1 Cisco IOx LXC
Activated profile name : custom

Resource reservation
  Memory               : 2048 MB
  Disk                 : 10 MB
  CPU                  : 4000 units

Attached devices
  Type              Name               Alias
  ---------------------------------------------
  serial/shell     iox_console_shell   serial0
  serial/aux       iox_console_aux     serial1
  serial/syslog    iox_syslog          serial2
  serial/trace     iox_trace           serial3

Network interfaces
```

```
   ----------------------------------------
eth0:
   MAC address         : 52:54:dd:38:a3:da
```

The following is sample output from the **show app-hosting infra** command:

```
Device# show app-hosting infra

App signature verification: disabled
```

The following is sample output from the **show app-hosting list** command:

```
Device# show app-hosting list

App id                           State
---------------------------------------------------------
perfsonar                        RUNNING
```

The following is sample output from the **show app-hosting resource** command:

```
Device# show app-hosting resource

Disk space:
  Total: 115300 MB
  Available: 111282 MB
Memory:
  Total: 2048 MB
  Available: 0 MB
CPU:
  Total: 7400 units
  Available: 3400 units
```

The following is sample output from the **show app-hosting utilization appid** command:

```
Device# show app-hosting utilization appid perfsonar

Application: perfsonar
CPU Utilization:
  CPU Allocation: 4000 units
  CPU Used:       0.01 %
Memory Utilization:
  Memory Allocation: 2048 MB
  Memory Used:       399112 KB
Disk Utilization:
  Disk Allocation: 10 MB
  Disk Used:       0.00 MB
```

All output fields are self-explanatory.

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
resource profile	Changes the application resource profile.

show controller ethernet-controller AppGigabitEthernet

To display details about the application hosting AppGigabitEthernet controller interface, use the **show controller ethernet-controller AppGigabitEthernet** command in privileged EXEC mode.

show controller ethernet-controller AppGigabitEthernet *interface-number*

Syntax Description

interface-number	Interface number.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

Example

The following sample output from the **show controller ethernet-controller AppGigabitEthernet** *interface-number* command:

```
Device# show controller ethernet-controller AppGigabitEthenet 1/0/1

Transmit                  AppGigabitEthernet1/0/1   Receive
           0 Total bytes                            0 Total bytes
           0 Unicast frames                         0 Unicast frames
           0 Unicast bytes                          0 Unicast bytes
           0 Multicast frames                       0 Multicast frames
           0 Multicast bytes                        0 Multicast bytes
           0 Broadcast frames                       0 Broadcast frames
           0 Broadcast bytes                        0 Broadcast bytes
           0 System FCS error frames                0 IpgViolation frames
           0 MacUnderrun frames                     0 MacOverrun frames
           0 Pause frames                           0 Pause frames
           0 Cos 0 Pause frames                     0 Cos 0 Pause frames
           0 Cos 1 Pause frames                     0 Cos 1 Pause frames
           0 Cos 2 Pause frames                     0 Cos 2 Pause frames
           0 Cos 3 Pause frames                     0 Cos 3 Pause frames
           0 Cos 4 Pause frames                     0 Cos 4 Pause frames
           0 Cos 5 Pause frames                     0 Cos 5 Pause frames
           0 Cos 6 Pause frames                     0 Cos 6 Pause frames
           0 Cos 7 Pause frames                     0 Cos 7 Pause frames
           0 Oam frames                             0 OamProcessed frames
           0 Oam frames                             0 OamDropped frames
           0 Minimum size frames                    0 Minimum size frames
           0 65 to 127 byte frames                  0 65 to 127 byte frames
           0 128 to 255 byte frames                 0 128 to 255 byte frames
           0 256 to 511 byte frames                 0 256 to 511 byte frames
           0 512 to 1023 byte frames                0 512 to 1023 byte frames
           0 1024 to 1518 byte frames               0 1024 to 1518 byte frames
           0 1519 to 2047 byte frames               0 1519 to 2047 byte frames
           0 2048 to 4095 byte frames               0 2048 to 4095 byte frames
           0 4096 to 8191 byte frames               0 4096 to 8191 byte frames
           0 8192 to 16383 byte frames              0 8192 to 16383 byte frames
           0 16384 to 32767 byte frame              0 16384 to 32767 byte frame
           0 > 32768 byte frames                    0 > 32768 byte frames
```

```
0 Late collision frames                  0 SymbolErr frames
0 Excess Defer frames                    0 Collision fragments
0 Good (1 coll) frames                   0 ValidUnderSize frames
0 Good (>1 coll) frames                  0 InvalidOverSize frames
0 Deferred frames                        0 ValidOverSize frames
0 Gold frames dropped                    0 FcsErr frames
0 Gold frames truncated
0 Gold frames successful
0 1 collision frames
0 2 collision frames
0 3 collision frames
0 4 collision frames
0 5 collision frames
0 6 collision frames
0 7 collision frames
0 8 collision frames
0 9 collision frames
0 10 collision frames
0 11 collision frames
0 12 collision frames
0 13 collision frames
0 14 collision frames
0 15 collision frames
0 Excess collision frames
```

The output fields are self-explanatory.

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.

show gnxi server vrf

Displays the state of the gRPC servers and configured VRFs.

show gnxi server vrf

This command has no arguments or keywords.

Command Modes

Privileged EXEC (3)

Command History

Release	Modification
Cisco IOS XE 18.2.1	This command was introduced.

Example

The following is sample output from the **show gnxi server vrf** command.

```
Device# show gnxi server vrf

gRPC Servers

Port  State Visibility TLS    Address VRF ID Table ID    Configured
                              Family                     VRF
-------------------------------------------------------------------------
9339  Up    Public     TLS    IPv4     16     0x00000006  gn-vrf
9339  Up    Public     TLS    IPv4     16     0x1e000004  gn-vrf
19099 Up    Internal   No-TLS IPv4/     0     0x00000000
                              IPv6
```

Table 2: Related commands

Command	Description
gnxi vrf	Configures a gNXI server VRF.

show gnxi state

To display Google RPC (gRPC) Network Operations Interface (gNOI)/gRPC Network Management/Operations Interface (gNXI) state information, use the **show gnxi state** command in privileged EXEC mode.

show gnxi state [**detail** | **stats**]

Syntax Description

detail	(Optional) Displays detailed state information about the gNMI broker (GNMIB).
stats	(Optional) Display GNMIB operational statistics.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Amsterdam 17.3.1	This command was introduced. This command replaces the **show gnmi-yang state** command.

Example

The following is sample output from the **show gnxi state detail** command:

```
Device> enable
Device# show gnxi state detail

Settings
========
  Server: Enabled
  Server port: 1024
  Secure server: Disabled
  Secure server port: 9339
  Secure client authentication: Disabled
  Secure trustpoint:
  Secure client trustpoint:
  Secure password authentication: Disabled

GNMI
====
  Admin state: Enabled
  Oper status: Up
  State: Provisioned

  gRPC Server
  -----------
    Admin state: Enabled
    Oper status: Up

  Configuration service
  ---------------------
    Admin state: Enabled
    Oper status: Up

  Telemetry service
  -----------------
    Admin state: Enabled
    Oper status: Up
```

```
GNOI
====

  Cert Management service
  -----------------
    Admin state: Enabled
    Oper status: Up

  OS Image service
  ----------------
    Admin state: Disabled
    Oper status: Up
    Supported: Not supported on this platform
```

The output fields are self-explanatory.

The following is sample output from the **show gnxi state stats** command:

```
Device> enable
Device# show gnxi state stats

GNMI
====
  Get: 1
  Set: 1
  Capabilities: 1
  Subscribe: 0

GNOI CERT
=========
  Get: 0
  Install: 0
  Rotate: 0
  Revoke: 0
  Cert CSR: 0

GNOI OS
=======
  Install: 0
  Activate: 1
  Verify: 1
```

The table below lists the significant fields shown in the display.

Table 3: show gnxi state stats Field Descriptions

Field	Description
GNMI	gNMI protocol information.
Get	Number of Get RPCs received.
Set	Number of Set RPCs received.
GNOI Cert	gNOI certificate information.
Install	Number of Install RPCs received.

Field	Description
Rotate	Number of Rotate RPCs received.
Revoke	Number of Revoke RPCs received.
Cert CSR	Number of Certificate Signing Requests (CSRs) received.
GNOI OS	GNOI OS installation service information.
Install	Number of Install RPC requests received.
Activate	Number of Activate RPC requests received.
Verify	Number of Verify RPC requests received.

Related Commands

Command	Description
gnxi	Enables gNXI.

show install

To display information about data model update packages, use the **show install** command in privileged EXEC mode.

show install {**active** | **committed** | **inactive** | **log** | **package** {**bootflash:** | **flash:** | **webui:**} | **rollback** | **summary** | **uncommitted**}

Syntax Description

active	Displays information about active packages.
committed	Displays package activations that are persistent.
inactive	Displays inactive packages.
log	Displays entries stored in the logging installation buffer.
package	Displays metadata information about the package, including description, restart information, components in the package, and so on.
{**bootflash:** | **flash:** | **webui:**}	Specifies the location of the model update package.
rollback	Displays the software set associated with a saved installation.
summmary	Displays information about the list of active, inactive, committed, and superseded packages.
uncommitted	Displays package activations that are non persistent.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced on the following platforms: • Cisco 4000 Series Integrated Services Routers • Cisco Catalyst 9300 Series Switches • Cisco Catalyst 9500 Series Switches • Cisco Cloud Services Router 1000v • Cisco Integrated Services Virtual Routers (ISRv)
Cisco IOS XE Everest 16.6.1	This command was implemented on the following platforms: • Cisco Catalyst 3650 Series Switches • Cisco Catalyst 3850 Series Switches

Usage Guidelines

Use the show commands to view the status of an installed model update package.

Cisco 4000 Series Integrated Services Routers

The following is sample output from the **show install package** command:

```
Device# show install package bootflash:
isr4300-universalk9.16.05.01.CSCxxxxxxx.dmp.bin

Name: isr4300-universalk9.16.05.01.CSCxxxxxxx.dmp.bin
Version: 16.5.1.0.199.1484082952..Everest
Platform: ISR4300
Package Type: dmp
Defect ID: CSCxxxxxxx
Package State: Added
Supersedes List: {}
Smu ID: 1
Device#
```

The following is sample output from the **show install summary** command:

```
Device# show install summary

Active Packages:
bootflash:isr4300-universalk9.16.05.01.CSCxxxxxxx.dmp.bin
Inactive Packages:
No packages
Committed Packages:
No packages
Uncommitted Packages:
bootflash:isr4300-universalk9.16.05.01.CSCxxxxxxx.dmp.bin
Device#
```

The following is sample output from the **show install log** command:

```
Device# show install log

[0|install_op_boot]: START Fri Feb 24 19:20:19 Universal 2017
[0|install_op_boot]: END SUCCESS  Fri Feb 24 19:20:23 Universal 2017
[3|install_add]: START Sun Feb 26 05:55:31 UTC 2017
[3|install_add( FATAL)]: File path (scp) is not yet supported for this command
[4|install_add]: START Sun Feb 26 05:57:04 UTC 2017
[4|install_add]: END SUCCESS /bootflash/isr4300-universalk9.16.05.01.CSCxxxxxxx.dmp.bin
Sun Feb 26 05:57:22 UTC 2017
[5|install_activate]: START Sun Feb 26 05:58:41 UTC 2017
```

The table below lists the significant fields shown in the display.

Table 4: show install summary Field Descriptions

Field	Description
Active Packages	Name of the active model update package.
Inactive Packages	List of inactive packages.
Committed Packages	Installed model update packages that have saved or committed changes to the hard disk, so that the changes become persistent across reloads.

Field	Description
Uncommitted Packages	Model update package activations that are non persistent.

Cisco Catalyst 3000 Series Switches

The following sample output from the **show install summary** command displays that the update package is now committed, and that it will be persistent across reloads:

```
Device# show install summary

Active Packages:
bootflash:cat3k_caa-universalk9.16.06.01.CSCxxxxxxx.dmp.bin
Inactive Packages:
No packages
Committed Packages:
bootflash:cat3k_caa-universalk9.16.06.01.CSCxxxxxxx.dmp.bin
Uncommitted Packages:
No packages
Device#
```

Related Commands

Command	Description
install	Installs data model update packages.

show iox-service

To display the status of all IOx services, use the **show iox-service** command in privileged EXEC mode.

show iox-service [**detail**]

Syntax Description

detail	(Optional) Displays detailed information about the application/appliance.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Everest 16.5.1	This command was introduced.
Cisco IOS XE Amsterdam 17.2.1	The output of the command was modified to display the cold restart synchronization information.

Usage Guidelines

IOx is a Cisco-developed end-to-end application framework that provides application hosting capabilities fordifferent application types on Cisco network platforms. Cisco application hosting framework (CAF) is an IOx Python process that manages virtualized and container applications that run on devices. To enable IOx, configure the **iox** command. After configuring this command, you can update the application hosting configuration.

IOXMAN is a process that establishes a tracing infrastructure to provide logging or tracing services for guest applications, except Libvirt, that emulates serial devices.

Example

The following is sample output from the **show iox-service** command:

```
Device# show iox-service

IOx Infrastructure Summary:
---------------------------
IOx service (CAF) 1.10.0.0 : Running
IOx service (HA)           : Running
IOx service (IOxman)       : Running
IOx service (Sec storage)  : Not Running
Libvirtd   1.3.4           : Running
Dockerd    18.03.0         : Running
Application DB Sync Info   : Available
Sync status                : Successful
Last application sync time: 2020-03-25 15:23:37.132829
```

The table below lists the significant fields shown in the display.

Table 5: show iox-service Field Descriptions

Field	Description
IOx service (CAF)	Status of the Cisco Application Framework (CAF).

Field	Description
IOx service (HA)	Status of high availability. High availability must be running, if you have redundant hardware, like a redundant route processor (RP).
IOx service (IOxman)	Status of the IOx Manager.
Libvirtd	Status of the Linux Library Virtual daemon.
Sync status	Status of the IOx cold restart. Shows whether the synchronization was sucessful or not.
Last application sync time	Date and time when the last synchronization happened.

The following is sample output from the **show iox-service detail** command:

```
Device# show iox-service detail

IOx Infrastructure Summary:
---------------------------
IOx service (CAF) 1.10.0.0 : Running
IOx service (HA)           : Running
IOx service (IOxman)       : Running
IOx service (Sec storage) : Not Running
Libvirtd   1.3.4           : Running
Dockerd    18.03.0         : Running
Application DB Sync Info : Available
Sync Status : Disabled

------------------ show platform software process list switch active r0 name caf
------------------
Name: run_ioxn_caf.sh
  Process id       : 743
  Parent process id: 302
  Group id         : 743
  Status           : S
  Session id       : 9377
  User time        : 20
  Kernel time      : 10
  Priority         : 20
  Virtual bytes    : 6459392
  Resident pages   : 1420
  Resident limit   : 18446744073709551615
  Minor page faults: 17234
  Major page faults: 0

------------------ show platform software process list switch active r0 name libvirtd
------------------
Name: libvirtd.sh
  Process id       : 5839
  Parent process id: 1
  Group id         : 5839
  Status           : S
  Session id       : 5839
  User time        : 0
  Kernel time      : 0
  Priority         : 20
  Virtual bytes    : 4067328
  Resident pages   : 746
```

```
  Resident limit   : 18446744073709551615
  Minor page faults: 246
  Major page faults: 0

Name: libvirtd
  Process id       : 5862
  Parent process id: 5839
  Group id         : 5839
  Status           : S
  Session id       : 5839
  User time        : 122
  Kernel time      : 202
  Priority         : 20
  Virtual bytes    : 1246498816
  Resident pages   : 3976
  Resident limit   : 18446744073709551615
  Minor page faults: 2685
  Major page faults: 31

------------------ show platform software process list switch active r0 name dockerd
------------------
Name: dockerd
  Process id       : 8622
  Parent process id: 7979
  Group id         : 8622
  Status           : S
  Session id       : 9377
  User time        : 1957
  Kernel time      : 1132
  Priority         : 20
  Virtual bytes    : 1824083968
  Resident pages   : 15276
  Resident limit   : 18446744073709551615
  Minor page faults: 9515
  Major page faults: 338


Device#
```

Related Commands

Command	Description
iox	Configure IOx services.

show log profile netconf-yang

To write NETCONF-YANG process logs to a file, use the **show log profile netconf-yang** command in privileged EXEC mode.

show log profile netconf-yang internal

Syntax Description

internal	Selects all debug logs. **Note** This keyword for use by customer support.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.8.1	This command was introduced.

Usage Guidelines

Logs are displayed on the device console when the command is executed.

Example

The following is sample output from the **show log profile netconf-yang internal** command:

```
Device# show log profile netconf-yang internal

excuting cmd on chassis local ...
Collecting files on current[local] chassis.


DECODER ERROR: NOTE: Tracelog may not be generated from clang binary, and is not encoded.
Please use native linux tools (vi/less/more/cat...) to read the file

2018/01/24 15:58:50.356 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  gdb port
 9919 allocated
2018/01/24 15:58:50.365 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  swift_repl
 port 8019 allocated
2018/01/24 15:58:50.422 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (info): (std):
cat: /tmp/sw/boot/boot_debug.conf: No such file or directory
2018/01/24 15:58:50.427 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (info): (std):
/usr/binos/conf/pman.sh: line 424: sigusr1_func: readonly function
2018/01/24 15:58:50.430 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
process scoreboard /tmp/rp/process/pttcd%rp_0_0%0 pttcd%rp_0_0%0.pid is 12040
2018/01/24 15:58:50.430 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
pttcd%rp_0_0%0.gdbport is 9919
2018/01/24 15:58:50.430 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
pttcd%rp_0_0%0.swift_replport is 8019
2018/01/24 15:58:50.439 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (info): (std):
12040 (process ID) old priority 0, new priority 0
2018/01/24 15:58:50.439 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
Launching pttcd  on fru rp slot 0 bay 0 instance 0 log /tmp/rp/trace/pttcd_pmanlog
2018/01/24 15:58:50.439 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
Hold failures 2, hold interval 1800
2018/01/24 15:58:50.439 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
```

```
PATH is
/tmp/sw/rp/0/0/rp_daemons/mount/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/bin:/tmp/sw/rp/0/0
/rp_daemons/mount/usr/binos/conf:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/sbin:/tmp/sw/rp/0/0
/rp_daemons/mount/usr/binos/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/cpp/bin:/usr/bin:/bin:/sbin:
/usr/binos/conf:/usr/binos/bin:/sbin:/bin:/usr/bin:/usr/sbin:/usr/binos
/conf:/sbin:/bin:/usr/bin:/usr/sbin:/usr/binos/conf
2018/01/24 15:58:50.439 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
LD_LIBRARY_PATH is
2018/01/24 15:58:50.441 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
PREPROC_OPTIONS ==
2018/01/24 15:58:50.441 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
command line used   pttcd  >> /tmp/rp/trace/pttcd_pmanlog_cmd 2&>1 &
2018/01/24 15:58:50.444 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
full_path is /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pttcd
2018/01/24 15:58:50.446 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
Resolved readlink process /tmp/sw/mount
/asr1000rpx86-rpcontrol.BLD_V168_THROTTLE_LATEST_20180122_164958_V16_8_0_177.SSA.pkg
/usr/binos/bin/pttcd
2018/01/24 15:58:50.446 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
Full path used to spawn the process: /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pttcd
2018/01/24 15:58:50.452 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
Binary_arch set to: [x86_64_cge7]
2018/01/24 15:58:50.460 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (info): (std):
chmod: cannot access '/tmp/tmppub/tracekey_cache//tmp/sw/mount
/asr1000rpx86-rpcontrol.BLD_V16_8_0_177.SSA.pkg/usr/binos/bin/pttcd':
No such file or directory
2018/01/24 15:58:50.461 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  actual
pttcd pid is 12542
2018/01/24 15:58:50.461 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
Checking for cgroup for PID 12542
2018/01/24 15:58:50.461 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
/tmp/rp/pvp/process_state/pttcd%rp_0_0%0#12040_state marked up
2018/01/24 15:58:50.474 {pttcd_R0-0}{1}: [pttcd] [12542]: (ERR): init_callhome() failed
2018/01/24 15:58:50.475 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):  oom score
 adj value is 399
2018/01/24 15:58:50.475 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (info): (std):
12040 (process ID) old priority 0, new priority -6
2018/01/24 15:58:50.475 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [12142]: (note):
Wait for signal or process exit: 12542
/harddisk/tracelogs/tmp_trace/pttcd_pmanlog_R0-0.12142_0.20180124155850.bin: DECODE(25:25:0:1)
2018/01/24 15:58:52.077 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  gdb port
9920 allocated
2018/01/24 15:58:52.085 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):  swift_repl
 port 8020 allocated
2018/01/24 15:58:52.150 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (info): (std):
cat: /tmp/sw/boot/boot_debug.conf: No such file or directory
2018/01/24 15:58:52.153 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (info): (std):
/usr/binos/conf/pman.sh: line 424: sigusr1_func: readonly function
2018/01/24 15:58:52.157 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
process scoreboard /tmp/rp/process/pubd%rp_0_0%0 pubd%rp_0_0%0.pid is 14416
2018/01/24 15:58:52.157 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
pubd%rp_0_0%0.gdbport is 9920
2018/01/24 15:58:52.157 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
pubd%rp_0_0%0.swift_replport is 8020
2018/01/24 15:58:52.165 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (info): (std):
14416 (process ID) old priority 0, new priority 0
2018/01/24 15:58:52.166 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
Launching pubd  on fru rp slot 0 bay 0 instance 0 log /tmp/rp/trace/pubd_pmanlog
2018/01/24 15:58:52.166 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
Hold failures 2, hold interval 1800
2018/01/24 15:58:52.166 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [14520]: (note):
PATH is
/tmp/sw/rp/0/0/rp_daemons/mount/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/bin:/tmp/sw/rp/0/0
/rp_daemons/mount/usr/binos/conf:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/sbin:/tmp/sw/rp/0/0/rp_daemons/mount
```

```
/usr/binos/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/cpp/bin:/usr/bin:/bin:/sbin:/usr/binos/conf:/usr/binos/bin:
/sbin:/bin:/usr/bin:/usr/sbin:/usr/binos/conf:/sbin:/bin:/usr/bin:/usr/sbin:/usr/binos/conf
```

show log profile restconf

To write RESTCONF process logs to a file, use the **show log profile restconf** command in privileged EXEC mode.

show log profile restconf internal

Syntax Description

internal	Selects all debug logs. **Note** This keyword for use by customer support.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.8.1	This command was introduced.

Usage Guidelines

Logs are displayed on the device console when he command is executed.

Example

The following is sample output from the **show log profile restconf** command:

```
Device# show log profile restconf internal


excuting cmd on chassis local ...
Collecting files on current[local] chassis.
Total # of files collected = 17
Decoding files:
DECODER ERROR: NOTE: Tracelog may not be generated from clang binary, and is not encoded.
Please use native linux tools (vi/less/more/cat...) to read the file

2018/03/23 13:05:13.945 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  gdb port
 9908 allocated
2018/03/23 13:05:13.962 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  swift_repl
 port 8008 allocated
2018/03/23 13:05:14.041 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (info): (std):
cat:
 /tmp/sw/boot/boot_debug.conf: No such file or directory
2018/03/23 13:05:14.046 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (info): (std):
 /usr/binos/conf/pman.sh: line 424: sigusr1_func: readonly function
2018/03/23 13:05:14.050 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  process
scoreboard
 /tmp/rp/process/pttcd%rp_0_0%0 pttcd%rp_0_0%0.pid is 2550
2018/03/23 13:05:14.050 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
pttcd%rp_0_0%0.gdbport is 9908
2018/03/23 13:05:14.050 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
 pttcd%rp_0_0%0.swift_replport is 8008
2018/03/23 13:05:14.059 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (info): (std):
2550
 (process ID) old priority 0, new priority 0
2018/03/23 13:05:14.060 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Launching
```

```
 pttcd
 on fru rp slot 0 bay 0 instance 0 log /tmp/rp/trace/pttcd_pmanlog
2018/03/23 13:05:14.060 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Hold
failures 2,
 hold interval 1800
2018/03/23 13:05:14.060 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  PATH is

/tmp/sw/rp/0/0/rp_daemons/mount/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/conf:

 /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/sbin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin:


/tmp/sw/rp/0/0/rp_daemons/mount/usr/cpp/bin:/usr/bin:/bin:/sbin:/usr/binos/conf:/usr/binos/bin:/sbin:/bin:/usr/bin:

 /usr/sbin:/usr/binos/conf:/sbin:/bin:/usr/bin:/usr/sbin:/usr/binos/conf
2018/03/23 13:05:14.060 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
LD_LIBRARY_PATH is
2018/03/23 13:05:14.063 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
PREPROC_OPTIONS ==
2018/03/23 13:05:14.063 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  command
line used   pttcd  >>
 /tmp/rp/trace/pttcd_pmanlog_cmd 2&>1 &
2018/03/23 13:05:14.068 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  full_path
 is
 /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pttcd
2018/03/23 13:05:14.069 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Resolved
 readlink process
 /tmp/sw/mount/asr1000rpx86-rpcontrol.2018-03-07_18.30_rifu.SSA.pkg/usr/binos/bin/pttcd
2018/03/23 13:05:14.069 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Full path
 used to spawn the process:
 /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pttcd
2018/03/23 13:05:14.076 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Binary_arch
 set to: [x86_64_cge7]
2018/03/23 13:05:14.087 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (info): (std):
chmod: cannot access

'/tmp/tmppub/tracekey_cache//tmp/sw/mount/asr1000rpx86-rpcontrol.2018-03-07_18.30_rifu.SSA.pkg

 /usr/binos/bin/pttcd': No such file or directory
2018/03/23 13:05:14.088 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  actual
pttcd pid is 2936
2018/03/23 13:05:14.088 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Checking
 for cgroup for PID 2936
2018/03/23 1 3:05:14.088 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):
/tmp/rp/pvp/process_state/pttcd%rp_0_0%0#2550_state marked up
2018/03/23 13:05:14.097 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  oom score
 adj value is 399
2018/03/23 13:05:14.102 {pttcd_R0-0}{1}: [pttcd] [2936]: (ERR): init_callhome() failed
2018/03/23 13:05:14.102 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (info): (std):
 2550 (process ID) old priority 0, new priority -6
2018/03/23 13:05:14.102 {pttcd_pmanlog_R0-0}{1}: [pttcd_pmanlog] [2628]: (note):  Wait for
 signal or process exit: 2936
/harddisk/tracelogs/tmp_trace/pttcd_pmanlog_R0-0.2628_0.20180323130513.bin: DECODE(25:25:0:1)
2018/03/23 13:05:16.895 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  gdb port
9920 allocated
2018/03/23 13:05:16.904 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  swift_repl
 port 8020 allocated
2018/03/23 13:05:16.978 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (info): (std):
 cat: /tmp/sw/boot/boot_debug.conf: No such file or directory
2018/03/23 13:05:16.983 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (info): (std):
 /usr/binos/conf/pman.sh: line 424: sigusr1_func: readonly function
2018/03/23 13:05:16.987 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  process
scoreboard
 /tmp/rp/process/pubd%rp_0_0%0 pubd%rp_0_0%0.pid is 4922
```

```
2018/03/23 13:05:16.987 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
pubd%rp_0_0%0.gdbport is 9920
2018/03/23 13:05:16.987 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
pubd%rp_0_0%0.swift_replport is 8020
2018/03/23 13:05:16.996 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (info): (std):
 4922 (process ID) old priority 0, new priority 0
2018/03/23 13:05:16.997 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
 Launching pubd  on fru rp slot 0 bay 0 instance 0 log /tmp/rp/trace/pubd_pmanlog
2018/03/23 13:05:16.997 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  Hold failures
 2, hold interval 1800
2018/03/23 13:05:16.997 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  PATH is
 /tmp/sw/rp/0/0/rp_daemons/mount/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/bin:/tmp/sw/rp/0/0/


rp_daemons/mount/usr/binos/conf:/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/sbin:/tmp/sw/rp/0/0/

 rp_daemons/mount/usr/binos/bin:/tmp/sw/rp/0/0/rp_daemons/mount/usr/cpp/bin:/usr/bin:/

bin:/sbin:/usr/binos/conf:/usr/binos/bin:/sbin:/bin:/usr/bin:/usr/sbin:/usr/binos/conf:/sbin:/bin:

 /usr/bin:/usr/sbin:/usr/binos/conf
2018/03/23 13:05:16.997 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
LD_LIBRARY_PATH is
2018/03/23 13:05:17.001 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
PREPROC_OPTIONS ==
2018/03/23 13:05:17.001 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  command
line used   pubd  >>
 /tmp/rp/trace/pubd_pmanlog_cmd 2&>1 &
2018/03/23 13:05:17.007 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):  full_path
is
/tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pubd
2018/03/23 13:05:17.009 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
 Resolved readlink process /tmp/sw/mount/asr1000rpx86-rpcontrol.2018-03-07_18.30_rifu.SSA.pkg/

 usr/binos/bin/pubd
2018/03/23 13:05:17.009 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
 Full path used to spawn the process: /tmp/sw/rp/0/0/rp_daemons/mount/usr/binos/bin/pubd
2018/03/23 13:05:17.017 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (note):
 Binary_arch set to: [x86_64_cge7]
2018/03/23 13:05:17.030 {pubd_pmanlog_R0-0}{1}: [pubd_pmanlog] [4998]: (info): (std): chmod:

 cannot access
!
!
!
```

show netconf

To display the configured NETCONF information, use the **show netconf** command in privileged EXEC mode.

show { **counters** | **schema** | **session** }

Syntax Description

counters	Displays information about the NETCONF statistics counters.
schema	Displays the NETCONF schema.
session	Displays NETCONF session information.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE 17.16.1	This command was introduced.

Usage Guidelines

Example

The following is sample output from the **show netconf counters** command:

```
Device# show netconf counters

NETCONF Counters
Connection Attempts:0: rejected:0 no-hello:0 success:0
Transactions
 total:0, success:0, errors:0
detailed errors:
 in-use 0             invalid-value 0        too-big 0
 missing-attribute 0  bad-attribute 0        unknown-attribute 0
 missing-element 0    bad-element 0          unknown-element 0
 unknown-namespace 0  access-denied 0        lock-denied 0
 resource-denied 0    rollback-failed 0            data-exists 0
 data-missing 0       operation-not-supported 0    operation-failed 0
 partial-operation 0
```

The following is sample output from the **show netconf schema** command:

```
Device# show netconf schema

New Name Space 'urn:ietf:params:xml:ns:netconf:base:1.0'
<VirtualRootTag> [0, 1] required
  <rpc-reply> [0, 1] required
    <ok> [0, 1] required
    <data> [0, 1] required
    <rpc-error> [0, 1] required
      <error-type> [0, 1] required
      <error-tag> [0, 1] required
      <error-severity> [0, 1] required
      <error-app-tag> [0, 1] required
      <error-path> [0, 1] required
      <error-message> [0, 1] required
      <error-info> [0, 1] required
```

```
      <bad-attribute> [0, 1] required
      <bad-element> [0, 1] required
      <ok-element> [0, 1] required
      <err-element> [0, 1] required
      <noop-element> [0, 1] required
      <bad-namespace> [0, 1] required
      <session-id> [0, 1] required
<hello> [0, 1] required
  <capabilities> 1 required
    <capability> 1+ required
<rpc> [0, 1] required
  <close-session> [0, 1] required
  <commit> [0, 1] required
    <confirmed> [0, 1] required
    <confirm-timeout> [0, 1] required
  <copy-config> [0, 1] required
    <source> 1 required
      <config> [0, 1] required
        <cli-config-data> [0, 1] required
          <cmd> 1+ required
        <cli-config-data-block> [0, 1] required
        <xml-config-data> [0, 1] required
          <Device-Configuration> [0, 1] required
            <> any subtree is allowed
      <candidate> [0, 1] required
      <running> [0, 1] required
      <startup> [0, 1] required
      <url> [0, 1] required
    <target> 1 required
      <candidate> [0, 1] required
      <running> [0, 1] required
      <startup> [0, 1] required
      <url> [0, 1] required
  <delete-config> [0, 1] required
    <target> 1 required
      <candidate> [0, 1] required
      <running> [0, 1] required
      <startup> [0, 1] required
      <url> [0, 1] required
  <discard-changes> [0, 1] required
  <edit-config> [0, 1] required
    <target> 1 required
      <candidate> [0, 1] required
      <running> [0, 1] required
      <startup> [0, 1] required
      <url> [0, 1] required
    <default-operation> [0, 1] required
    <test-option> [0, 1] required
    <error-option> [0, 1] required
    <config> 1 required
      <cli-config-data> [0, 1] required
        <cmd> 1+ required
      <cli-config-data-block> [0, 1] required
      <xml-config-data> [0, 1] required
        <Device-Configuration> [0, 1] required
          <> any subtree is allowed
  <get> [0, 1] required
    <filter> [0, 1] required
      <config-format-text-cmd> [0, 1] required
        <text-filter-spec> [0, 1] required
      <config-format-text-block> [0, 1] required
        <text-filter-spec> [0, 1] required
      <config-format-xml> [0, 1] required
      <oper-data-format-text-block> [0, 1] required
```

```
      <show> 0+ required
      <exec> 0+ required
    <oper-data-format-xml> [0, 1] required
      <show> 0+ required
      <exec> 0+ required
<get-config> [0, 1] required
  <source> 1 required
    <config> [0, 1] required
      <cli-config-data> [0, 1] required
        <cmd> 1+ required
      <cli-config-data-block> [0, 1] required
      <xml-config-data> [0, 1] required
        <Device-Configuration> [0, 1] required
          <> any subtree is allowed
    <candidate> [0, 1] required
    <running> [0, 1] required
    <startup> [0, 1] required
    <url> [0, 1] required
  <filter> [0, 1] required
    <config-format-text-cmd> [0, 1] required
      <text-filter-spec> [0, 1] required
    <config-format-text-block> [0, 1] required
      <text-filter-spec> [0, 1] required
    <config-format-xml> [0, 1] required
<kill-session> [0, 1] required
  <session-id> [0, 1] required
<lock> [0, 1] required
  <target> 1 required
    <candidate> [0, 1] required
    <running> [0, 1] required
    <startup> [0, 1] required
    <url> [0, 1] required
<unlock> [0, 1] required
  <target> 1 required
    <candidate> [0, 1] required
    <running> [0, 1] required
    <startup> [0, 1] required
    <url> [0, 1] required
<validate> [0, 1] required
  <source> 1 required
    <config> [0, 1] required
      <cli-config-data> [0, 1] required
        <cmd> 1+ required
      <cli-config-data-block> [0, 1] required
      <xml-config-data> [0, 1] required
        <Device-Configuration> [0, 1] required
          <> any subtree is allowed
    <candidate> [0, 1] required
    <running> [0, 1] required
    <startup> [0, 1] required
    <url> [0, 1] required
<notification-on> [0, 1] required
<notification-off> [0, 1] required
```

The following is sample output from the **show netconf schema** command:

```
Device# show netconf session

Netconf Sessions: 0 open, maximum is 4
```

show netconf-yang

To display information about NETCONF-YANG processes, use the **show netconf-yang** command in privileged EXEC mode.

show netconf-yang {**datastores** | **sessions** [**detail** | **session-id** *session-id*] | **statistics**} [**R0** | **R1** | **RP** {**active** | **standby**}]

Syntax Description

datastores	Displays information about NETCONF-YANG datastores.
sessions	Displays information about NETCONF-YANG sessions.
detail	(Optional) Displays detailed information about NETCONF-YANG sessions.
session-id *session-id*	(Optional) Displays information about the specified session. Valid values are from 1 to 4294967295.
statistics	Displays information about NETCONF-YANG statistics.
R0	(Optional) Displays information about the Route Processor (RP) slot 0.
R1	(Optional) Displays information about the RP slot 1.
RP	(Optional) Displays information about the RP.
active	(Optional) Displays information about the active instance of the RP.
standby	(Optional) Displays information about the standby instance of the RP.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.8.1	This command was introduced.

Usage Guidelines

This command displays information about global locks applied on the running datastore, candidate datastore, and startup datastore.

The **active** and **standby** keywords are only applicable to devices that supports both active and redundant route processors.

Example

This sample output from the **show netconf-yang datastores** commands displays the sessions that have global locks:

```
Device# show netconf-yang datastores

Datastore Name             : running
Globally Locked By Session : 42
```

```
Globally Locked Time       : 2018-01-15T14:25:14-05:00
```

The table below lists the significant fields shown in the display.

Table 6: show netconf-yang datastores Field Descriptions

Field	Description
Datastore Name	Name of the datastore supported by the device.
Globally Locked By Session	Number of NETCONF-YANG sessions that have the lock on the running datastore.
Globally Locked Time	Time when a NETCONF-YANG session acquires the lock.

The following is sample output from the **show netconf-yang sessions** command:

```
Device# show netconf-yang sessions

R: Global-lock on running datastore
C: Global-lock on candidate datastore
S: Global-lock on startup datastore

Number of sessions : 10

session-id  transport     username           source-host           global-lock
--------------------------------------------------------------------------------
40          netconf-ssh   admin              10.85.70.224          None
42          netconf-ssh   admin              10.85.70.224          None
44          netconf-ssh   admin              10.85.70.224          None
46          netconf-ssh   admin              10.85.70.224          None
48          netconf-ssh   admin              10.85.70.224          None
50          netconf-ssh   admin              10.85.70.224          None
52          netconf-ssh   admin              10.85.70.224          None
54          netconf-ssh   admin              10.85.70.224          None
56          netconf-ssh   admin              10.85.70.224          None
58          netconf-ssh   admin              10.85.70.224          None
```

The table below lists the significant fields shown in the display.

Table 7: show netconf-yang sessions Field Descriptions

Field	Description
session-id	Session identifier.
transport	Transport protocol used for session.
username	Client that is authenticated by the NETCONF-YANG system.
source-host	IP address of the client.
global-lock	True for sessions holding a global lock, and NONE, if there are no global locks.

This is sample output from the **show netconf-yang statistics** command:

```
Device# show netconf-yang statistics

netconf-start-time  : 2018-01-15T12:51:14-05:00
in-rpcs             : 0
in-bad-rpcs         : 0
out-rpc-errors      : 0
out-notifications   : 0
in-sessions         : 10
dropped-sessions    : 0
in-bad-hellos       : 0
```

The table below lists the significant fields shown in the display.

Table 8: show netconf-yang statistics Field Descriptions

Field	Description
netconf-start-time	Session establishment time.
in-rpcs	Total number of correct incoming RPCs.
in-bad-rpcs	Total number of incorrect incoming RPCs.
out-rpc-errors	Total number of RPC reply messages that indicate RPC errors.
out-notifications	Total number of outgoing notifications.
in-sessions	Total number of active NETCONF sessions.
dropped-sessions	Total number of dropped NETCONF sessions.

show netconf-yang diagnostics

To display NETCONF-YANG diagnostics information, use the **show netconf-yang diagnostics** command in privileged EXEC mode.

show netconf-yang diagnostics { **summary** | { **all** | **last** | **message** *number* } [**after** | **before** | **log** | **rollback**] }

Syntax Description

summary	Displays a summary of the NETCONF-YANG diagnostic information.
all	Displays all NETCONF-YANG diagnostic information.
last	Displays information about the last NETCONF RPC processed.
message *number*	Displays information about a specific NETCONF RPC message number.
after	(Optional) Displays the running configuration after a NETCONF RPC is processed.
before	(Optional) Displays the running configuration before a NETCONF RPC is processed.
log	(Optional) Displays the transaction logs for a NETCONF RPC.
rollback	(Optional) Displays information about the latest NETCONF rollback file.

Command Modes

Privileged EXEC (#)

Release	Modification
Cisco IOS XE Bengaluru 17.5.1	This command was introduced.

Example

The following is sample output from the **show netconf-yang diagnostics summary** command:

```
Device# show netconf-yang diagnostics summary

Diagnostic Debugging is ON

Diagnostic Debugging Level: Maximum
Total Log Size (bytes): 20097
Total Transactions: 1

message     username     session-id     transaction-id     start-time         end-time
         log size
-------------------------------------------------------------------------------------------------
1            admin       35              53                03/12/21 14:31:03  03/12/21
14:31:04     20097
```

The output fields are self-explanatory.

The following is sample output from the **show netconf-yang diagnostics last before** command:

```
Device# show netconf-yang diagnostics last before

--------------- Message 1 ---------------

----- Running-Config Before the NETCONF RPC -----


Building configuration...

Current configuration : 7207 bytes
!
! Last configuration change at 13:38:50 EDT Tue Sep 15 2020 by lab
!
version 17.5
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service internal
service call-home
no platform punt-keepalive disable-kernel-core
platform shell
!
hostname host1
!
!
vrf definition Mgmt-vrf
.
.
.
```

Related Commands

Command	Description
debug netconf-yang diagnotics	Enables the debugging of NETCONF-YANG diagnostics.

show netconf-yang ssh server

To display the operational status of the configured NETCONF-YANG SSH algorithms, use the **show netconf-yang ssh server** command in privileged EXEC mode.

show netconf-yang ssh server

This command has no arguments and keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Dublin 17.12.1	This command was introduced.

Usage Guidelines

You can also use the Cisco-IOS-XE-yang-interfaces-oper YANG model to query the operational state of the algorithms.

Example

The following is sample output from the **show netconf-yang ssh server** command:

```
Device# show netconf-yang ssh server

Algorithm Type Status
-------------------------------------------------------
rsa-sha2-256 Hostkey Enabled
rsa-sha2-512 Hostkey Enabled
ssh-rsa Hostkey Enabled
aes128-ctr Cipher Enabled
aes192-ctr Cipher Enabled
aes256-ctr Cipher Enabled
aes128-cbc Cipher Enabled
aes256-cbc Cipher Enabled
hmac-sha2-256 MAC Enabled
hmac-sha2-512 MAC Enabled
hmac-sha1 MAC Enabled
diffie-hellman-group14-sha1 KEX Enabled
diffie-hellman-group14-sha256 KEX Enabled
diffie-hellman-group16-sha512 KEX Enabled
ecdh-sha2-nistp256 KEX Enabled
ecdh-sha2-nistp384 KEX Enabled
ecdh-sha2-nistp521 KEX Enabled
```

The output fields are self-explanatory.

Related Commands

Command	Description
netconf-yang ssh server algorithm encryption	Enables the encryption algorithms that are advertised to a third party.
netconf-yang ssh server algorithm hostkey	Enables the hostkey algorithms that are advertised to a third party.

Command	Description
netconf-yang ssh server algorithm kex	Enables the KEX algorithms that are advertised to a third party.
netconf-yang ssh server algorithm mac	Enables the MAC algorithms that are advertised to a third party.

show netconf-yang ssh trustpoint

To display the operational status of the configured Public Key Infrastructure (PKI) trustpoint, use the **show netconf-yang ssh trustpoint** command in privileged EXEC mode.

show netconf-yang ssh trustpoint

This command has no arguments and keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE 17.16.1	This command was introduced.

Usage Guidelines

Use the **netconf-yang ssh trustpoint** command to configure SSH trustpoints for NETCONF-YANG

Example

The following is sample output from the **show netconf-yang ssh trustpoint** command:

```
Device# show netconf-yang ssh trustpoint

netconf-yang ssh trustpoint: trust-netconf
```

The output fields are self-explanatory.

Command	Description
netconf-yang ssh trustpoint	Configures PKI trustpoints for NETCONF-YANG.

show netconf-yang status

To display the list of configured NETCONF-YANG SSH algorithms, use the **show netconf-yang status** command in privileged EXEC mode.

show netconf-yang status

This command has no arguments and keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Denali 16.3.1	This command was introduced.

Example

The following is sample output from the **show netconf-yang status** command:

```
Device# show netconf-yang status

netconf-yang: enabled
netconf-yang candidate-datastore: disabled
netconf-yang side-effect-sync: enabled
netconf-yang ssh port: 830
netconf-yang turbocli: disabled
Hostkey Algorithms: rsa-sha2-256,rsa-sha2-512,ssh-rsa
Encryption Algorithms: aes128-ctr,aes192-ctr,aes256-ctr
MAC Algorithms: hmac-sha2-256,hmac-sha2-512,hmac-sha1
KEX Algorithms: diffie-hellman-group14-sha1,diffie-hellman-group14-sha256,
ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group16-sha512
```

The output fields are self-explanatory.

Related Commands

Command	Description
netconf-ssh server algorithm encryption	Enables the encryption algorithms that are advertised to a third party.
netconf-ssh server algorithm hostkey	Enables the hostkey algorithms that are advertised to a third party.
netconf-ssh server algorithm kex	Enables the KEX algorithms that are advertised to a third party.
netconf-ssh server algorithm mac	Enables the MAC algorithms that are advertised to a third party.

show platform software yang-management process

To display the status of the software processes required to support NETCONF-YANG, use the **show platform software yang-management process** in privileged EXEC mode.

show platform software yang-management process [**monitor** [**switch** { *switch-number* | **active** | **standby** } **R0**] | **switch** | { *switch-number* | **active** | **standby** } | **R0**]

Syntax Description

monitor	(Optional) Displays detailed information about processes that are running.
switch *switch-number*	(Optional) Displays information about the specified switch.
active	(Optional) Displays information about the active instance of the switch.
standby	(Optional) Displays information about the standby instance of the switch.
R0	(Optional) Displays information about the Route Processor (RP) slot zero.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Everest 16.3.1	This command was introduced.

Examples

The following is sample output from the **show platform software yang-management process** command:

```
Device# show platform software yang-management process

confd            : Running
nesd             : Running
syncfd           : Running
ncsshd           : Running
dmiauthd         : Running
vtyserverutild   : Running
opdatamgrd       : Running
nginx            : Running
ndbmand          : Running
```

The table below lists the significant fields shown in the display.

Table 9: show platform software yang-management process Field Descriptions

Field	Description
confd	Configuration daemon
nesd	Network element synchronizer daemon
syncfd	Sync from daemon
ncsshd	NETCONF Secure Shell (SSH) daemon
dmiauthd	Device management interface (DMI) authentication daemon
vtyserverutild	VTY server util daemon
opdatamgrd	Operational Data Manager daemon
nginx	NGINX web server
ndbmand	NETCONF database manager

The following is sample output from the **show platform software yang-management process monitor** command:

```
Device# show platform software yang-management process monitor

COMMAND           PID S    VSZ   RSS %CPU %MEM     ELAPSED
nginx           24689 S 139328 11996  0.0  0.2 24-02:00:55
nginx           24695 S 146544  6824  0.0  0.1 24-02:00:55
```

The table below lists the significant fields shown in the display.

Table 10: show platform software yang-management process monitor Field Descriptions

Field	Description
COMMAND	Command name
PID	Process ID
S	Process state
VSZ	Virtual memory size (in KB)
RSS	Resident set size (in KB)
%CPU	CPU usage percentage
%MEM	Memory usage percentage
ELAPSED	Elapsed execution time

Related Commands

Command	Description
show platform software yang-management process state	Displays the NETCONF-YANG process states.

show platform software yang-management process state

To display the NETCONF-YANG process states, use the **show platform software yang-management process state** command in privileged EXEC mode.

show platform software yang-management process state [**switch** { *switch-number* | **active** | **standby** } **R0**]

Syntax Description

switch *switch-number*	(Optional) Displays information about the specified switch.
active	(Optional) Displays information about the active instance of the switch.
standby	(Optional) Displays information about the standby instance of the switch.
R0	(Optional) Displays information about the Route Processor (RP) slot zero.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.5.1	This command was introduced in a release prior to Cisco IOS XE Bengaluru 17.5.1.

Example

The following is sample output from the **show platform software yang-management process state** command:

```
Device# show platform software yang-management process state

Confd Status: Started


Process              Status              State
------------------------------------------------------
nesd                 Running             Active
syncfd               Running             Active
ncsshd               Running             Not Applicable
dmiauthd             Running             Active
nginx                Running             Not Applicable
ndbmand              Running             Active
pubd                 Running             Active
gnmib                Not Running         Not Applicable
```

The table below lists the significant fields shown in the display.

Table 11: show platform software yang-management process state Field Descriptions

Field	Description
Confd Status	Configuration daemon
nesd	Network element synchronizer daemon
syncfd	Sync from daemon
ncsshd	NETCONF Secure Shell (SSH) daemon
dmiauthd	Device management interface (DMI) authentication daemon
nginx	NGINX web server
ndbmand	NETCONF database manager

Related Commands

Command	Description
debug netconf-yang diagnostics	Enables the debugging of NETCONF-YANG diagnostics.
show platform software yang-management process	Displays the status of the software processes required to support NETCONF-YANG.

show tech-support gnxi

To automatically run **show** commands that display gRPC Network Management/Operations Interface (gNXI)-related information, use the **show tech-support gnxi** command in privileged EXEC mode.

This command has no arguments or keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE 17.18.1	This command was introduced.

Usage Guidelines

The output of the **show tech-support gnxi** command is very long. To better manage this output, you can redirect the output to an external file (for example, **show tech-support acl | redirect flash:show-tech-gnxi.txt**) in the local writable storage file system or remote file system.

The output of this command displays the following commands:

- **show clock**
- **show switch**
- **show module**
- **show version**
- **show running-config**
- **show tech-support yang-management**
- **show logging**
- **show romvar**
- **show fips status**
- **show netconf-yang status**
- **show platform software yang-management process state**
- **show platform software yang-management process monitor**
- **show platform software ndbman switch active R0 models**
- **show platform software ndbman switch active R0 threads**
- **show platform software ndbman switch active R0 history all**
- **show netconf-yang datastores**
- **show netconf-yang session**
- **show netconf-yang ssh trustpoint**
- **show netconf-yang ssh server**
- **show netconf-yang ssh hostkey**

- **show crypto pki trustpoints gnxi_test_selfsigned_trustpoint status**
- **show crypto pki certificates pem gnxi_test_selfsigned_trustpoint**
- **show crypto key mypubkey all**
- **show processes memory platform sorted**
- **show crypto pki certificates**
- **show crypto pki trustpoints**
- **show gnxi state**
- **show gnxi state detail**
- **show gnxi state stats**
- **show vrf**
- **show vrf tableid**
- **show gnxi server vrf**
- **show gnxi os activate**
- **show gnxi os detail**
- **show gnxi os install**
- **show gnxi os internal history**
- **show gnxi os summary**
- **show install summary**
- **show install active**
- **show install committed**
- **show install inactive**
- **show gnxi grpctunnel destinations**
- **show platform software process memory switch active R0 name gnmib maps**
- **show platform software process memory switch active R0 name gnmib smaps**
- **show platform software process database dbm switch active R0 summary GNMI_CONFIG_DB**
- **show platform software process memory switch active R0 all sorted**
- **show platform software process database dbm switch active R0 details GNMI_CONFIG_DB**

Example

This is sample output from the **show tech-support gnxi** command:

```
Device# show tech-support gnxi

.
```

```
.
.

------------------ show gnxi state ------------------

State            Status
--------------------------------
Enabled          Up


------------------ show gnxi state detail ------------------

Settings
========
  Server: Disabled
  Server port: 50052
  Secure server: Enabled
  Secure server port: 9339
  Secure client authentication: Disabled
  Secure trustpoint: gnxi-test-cert
  Secure client trustpoint:
  Secure password authentication: Enabled

GNMI
====
  Admin state: Enabled
  Oper status: Up
  State: Provisioned

  gRPC Server
  -----------
    Admin state: Enabled
    Oper status: Up

  Configuration service
  ---------------------
    Admin state: Enabled
    Oper status: Up

  Telemetry service
  -----------------
    Admin state: Enabled
    Oper status: Up

GNOI
====

  Cert Management service
  -----------------
    Admin state: Enabled
    Oper status: Up

  OS Image service
  ----------------
    Admin state: Enabled
    Oper status: Up
    Supported: Supported

  Factory Reset service
  ---------------------
    Admin state: Enabled
    Oper status: Up
    Supported: Supported
```

```
gRPC Tunnel
===========
  Admin state: Enabled
  Oper status: Up


------------------ show gnxi state stats ------------------

GNMI
====
  Get: 0
  Set: 0
  Capabilities: 0
  Subscribe: 0

GNOI CERT
=========
  Get: 0
  Install: 0
  Rotate: 0
  Revoke: 0
  Cert CSR: 0

GNOI OS
=======
  Install: 0
  Activate: 0
  Verify: 0


------------------ show vrf ------------------


  Name                              Default RD            Protocols   Interfaces
  Gnxi-test-vrf                     <not set>             ipv4,ipv6
  Mgmt-vrf                          <not set>             ipv4,ipv6   Gi0/0
  Test-__-VRF                       <not set>
  Test-vrf                          <not set>             ipv4,ipv6
  Test-□üòÄ-VRF                      <not set>             ipv4
  gnxi-test-vrf                     <not set>             ipv4,ipv6
  gnxi-test-vrf-v4-only             <not set>             ipv4
  gnxi-test-vrf-v6-only             <not set>             ipv6
  other-test-vrf                    <not set>             ipv4,ipv6
  test-name-with-euro-,Ç¨-done      <not set>
  vrf-no-af                         <not set>

  Platform iVRF Name                iVRF Id               Interfaces
  __Platform_iVRF:_ID00_            0                     LI18/2


------------------ show vrf tableid ------------------


VRF Name                          Tableid             Address Family
Gnxi-test-vrf                     0x00000002          ipv4 unicast
Gnxi-test-vrf                     0x1E000002          ipv6 unicast
Mgmt-vrf                          0x00000001          ipv4 unicast
Mgmt-vrf                          0x1E000001          ipv6 unicast
Test-vrf                          0x00000004          ipv4 unicast
Test-vrf                          0x1E000003          ipv6 unicast
Test-□üòÄ-VRF                      0x00000005          ipv4 unicast
gnxi-test-vrf                     0x00000006          ipv4 unicast
gnxi-test-vrf                     0x1E000004          ipv6 unicast
```

```
gnxi-test-vrf-v4-only            0x00000007           ipv4 unicast
gnxi-test-vrf-v6-only            0x1E000005           ipv6 unicast
other-test-vrf                   0x00000009           ipv4 unicast
other-test-vrf                   0x1E000006           ipv6 unicast


------------------ show gnxi server vrf ------------------

gRPC Servers

Port      State     Visibility  TLS       Address Family  VRF ID    Table ID    Configured
 VRF
-------------------------------------------------------------------------------------------------------
9339      Up        Internal    TLS       IPv4/IPv6       0         0x00000000
9339      Up        Public      TLS       IPv4/IPv6       2         0x00000000
9339      Up        Public      TLS       IPv4/IPv6       16        0x00000000
19099     Up        Internal    No-TLS    IPv4/IPv6       0         0x00000000


------------------ show gnxi os activate ------------------

Current Activated Version: 17.18.01.0.1281.1749433610


------------------ show gnxi os detail ------------------

Summary
-------
Mode: Bundle Mode
Current Operation: No operation in progress

Install Status
--------------
No image has been added as Inactive
Current Image State: N/A
Bytes Received: 0

Activate Status
---------------
Current Activated Version: 17.18.01.0.1281.1749433610


------------------ show gnxi os install ------------------

No image has been added as Inactive
Current Image State: N/A
Bytes Received: 0


------------------ show gnxi os internal history ------------------



------------------ show gnxi os summary ------------------

Mode: Bundle Mode
Current Operation: No operation in progress
No image has been added as Inactive
Current Activated Version: 17.18.01.0.1281.1749433610

.
.
.
```

Related Commands

Command	Description
gnxi	Starts the gNXI process.

show telemetry connection

To display telemetry connection information, use the **show telemetry connection** command in privileged EXEC mode.

show telemetry connection { *index* { **brief** | **detail** | **subscription** } | **all** }

Syntax Description

index	Connection index. Valid values are from 0 to 4294967294.
brief	Displays a brief summary of the connection information.
detail	Displays detailed connection information.
subscription	Displays all subscriptions that use this connection.
all	Displays all connection information.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Cupertino 17.7.1	This command was introduced.

Usage Guidelines

The output of the **show telemetry connection** *index* **subscription** command matches the output of the **show telemetry ietf subscription brief** command.

Example

The following is sample output from the **show telemetry connection** *index* **detail** command:

```
Device#  show telemetry connection 1 detail

Index             : 1
Peer Address      : 203.0.113.254
Port              : 34365
VRF               : 0
Source Address    : 0.0.0.0
Type              : PROTOCOL
State             : Active
Peer ID           : admin
Receiver Name     :
Transport         : netconf
Use Count         : 1
State change Time : 05/26/21 11:57:51
```

The table below lists the significant fields shown in the display.

Table 12: show telemetry connection detail Field Descriptions

Field	Description
Index	Unique identifier for the connection.

Field	Description
Peer Address	IP address of the remote receiver.
Port	Remote port number on the receiver to which this connection is connected.
VRF	Virtual Routing and Forwarding (VRF) instance used by the connection.
Source Address	Local source address used by the connection.
Type	Receiver type. Currently *protocol* is the only supported receiver type.
State	State of the connection. The state can be active, connecting, pending, or disconnecting.
Peer ID	ID used by the remote receiver to authenticate itself. The ID can be removed, depending on the protocol that is used.
Receiver Name	Receiver name as configured by the **telemetry receiver** configuration command. This parameter is not set for legacy receivers.
Transport	Transport protocol used.
Use Count	Number of subscriptions that are currently using the connection.
State Change Time	Date and time of the last change to the connection state.

The following is sample output from the **show telemetry connection** *index* **subscription** command:

```
Device#  show telemetry connection 1 subscription

ID         Type           State      State Description
1005       Configured     Valid
1006       Configured     Valid
```

The following is sample output from the **show telemetry connection all** command:

```
Device# show telemetry connection all

Telemetry connections

Index Peer Address               Port  VRF Source Address             State
----- -------------------------- ----- --- -------------------------- ----------
    1 192.0.2.2                  57589 3   172.16.0.1                 Connecting
    2 198.51.100.2               57588 3   172.16.0.1                 Connecting
```

Related Commands

Command	Description
show telemetry ietf subscription brief	Displays a brief summary of the subscription information.
telemetry receiver protocol	Configures a named protocol receiver.

show telemetry ietf subscription

To display information about telemetry subscriptions on a device, use the **show telemetry ietf subscription** command in privileged EXEC mode.

show telemetry ietf subscription { { { *subscription-ID* [**receiver**] | **all** | **configured** | **dynamic** | **permanent** } | [**brief** | **detail**] } | **summary** }

Syntax Description

subscription-ID	Subscription ID. Valid values are from 0 to 4294967295.
receiver	(Optional) Displays the receiver details for a subscription, including the IP address, port of the remote client, the transport protocol, and the connection state (connected, disconnected, or connecting).
all	Displays all subscription information.
configured	Displays a list of subscriptions configured through the command or NETCONF set config.
dynamic	Displays information about dynamic subscriptions created using the *establish-subscription* RPC.
permanent	Displays permanent subscription information.
brief	(Optional) Displays a brief summary of the subscription information.
detail	(Optional) Displays the subscription information in detail.
summary	Displays a summary of all subscription information.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Everest 16.6.1	This command was introduced.
Cisco IOS XE Gibraltar 16.12.1	This command was modified. The **receiver** keyword was added.
Cisco IOS XE Cupertino 17.7.1	This command was modified. The **permanent** and **summary** keywords were added.

Usage Guidelines

Use the **show telemetry ietf subscription** command or the *get* RPC to retrieve the list of current subscription details on a device.

The **summary** keyword highlights the number of subscriptions configured, and the maximum number of supported subscriptions. If the subscriptions exceed the maximum number, the additional subscriptions are ignored.

Example

The following is sample output from the **show telemetry ietf subscription** *subscription-ID* **detail** command:

```
Device#  show telemetry ietf subscription 2147483667 detail

Telemetry subscription detail:

  Subscription ID: 2147483667
  State: Valid
  Stream: yang-push
  Encoding: encode-xml
  Filter:
    Filter type: xpath
    XPath: /mdt-oper:mdt-oper-data/mdt-subscriptions
  Update policy:
    Update Trigger: periodic
    Period: 1000
  Notes:
```

The following is sample output from the **show telemetry ietf subscription** *subscription-ID* **receiver** command:

```
Device# show telemetry ietf subscription 2147483649 receiver

Telemetry subscription receivers detail:

  Subscription ID: 2147483649
  Address: 10.85.181.2
  Port: 45143
  Protocol: gNMI
  Profile:
  State: Connected
  Explanation:
```

The following is sample output from the **show telemetry ietf subscription dynamic brief** command:

```
Device# show telemetry ietf subscription dynamic brief

Telemetry subscription brief

  ID               Type        State       Filter type
  ------------------------------------------------------
  2147483667       Dynamic     Valid       xpath
  2147483668       Dynamic     Valid       xpath
  2147483669       Dynamic     Valid       xpath
```

The following is sample output from the **show telemetry ietf subscription summary** command:

```
Device# show telemetry ietf subscription summary

Subscription Summary
```

```
====================
Maximum supported: 128

Subscription     Total       Valid      Invalid
-----------------------------------------------
All              1           0          1
Dynamic          0           0          0
Configured       1           0          1
Permanent        0           0          0
```

The table below lists the significant fields shown in the display.

Table 13: show telemetry ietf subscription Field Descriptions

Field	Description
Subscription ID	Subscription identifier.
State	Validity of a configured subscription. State will always be valid for dynamic subscriptions. For example, a configured subscription can be in a half-configured state, and therefore invalid. However, if a dynamic establish subscription is invalid, an error RPC response is sent back, and the subscription will not appear in this table.
Stream	Type of streaming used for subscriptions. Only YANG-push is supported.
Encoding	Specifies encode-xml as the encoding type.
Filter Type	Type of filter used for subscriptions. Only XPath is supported.
XPath	XPath filter type or how the subscribed information was selected.
Update Trigger	Type of trigger used to update subscriptions.
Period	Periodic timer configured to trigger an update. Values are specified in centiseconds (1/100 of a second).
Notes	A brief explanation about why a subscription is invalid. But for dynamic subscriptions, this field will always be empty.
ID	Subscription ID.

show telemetry internal connection

To display internal telemetry connection information, use the **show telemetry internal connection** command in privileged EXEC mode.

show telemetry internal connection *index* **detail**

Syntax Description

index	Connection index. Valid values are from 0 to 429496729.
detail	Displays all the fields for the chosen connection.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.
Cisco IOS XE Cupertino 17.7.1	This command was modified. The **detail** keyword was added.

Usage Guidelines

This command is not supported by all transport protocols.

Example

The following is sample output from the **show telemetry internal connection detail** command:

```
Device# show telemetry internal connection 4 detail

Telemetry protocol manager stats:

Con str               : 223.255.254.247:60251:0:0.0.0.0
Sockfd                : 71
Protocol              : netconf
State                 : Credentials parsed
Version               : V1.1
Source ip             : 223.255.254.247
Bytes Sent            : 4712230
Msgs Sent             : 9010
Msgs Received         : 1
Bytes in queue        : 0
```

The table below lists the significant fields shown in the display.

Table 14: show telemetry internal connection detail Field Descriptions

Field	Description
Con str	A string that describes the connection parameters used. This can include the source IP, source port, remote IP, and VRF. The exact format may vary based on the transport protocol.

Field	Description
Sockfd	ID of the internal file descriptor that is used for the connection.
Protocol	Transport protocol that is used by the connection.
State	Internal state of the connection as reported by the protocol manager.
Version	Protocol version.
Source ip	Source address of the connection.
Bytes Sent	Number of bytes sent by this connection since it became active.
Msgs Sent	Number of updates sent by this connection since it became active.
Msgs Received	Number of requests received by the connection since it became active. Depending on the protocol, this number can also be zero.
Bytes in queue	Number of bytes currently waiting to be sent to the remote receiver.

show telemetry internal diagnostics

To display telemetry diagnostics information, use the **show telemetry internal diagnostics** command in privileged EXEC mode.

show telemetry internal diagnostics

This command has no arguments or keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Usage Guidelines

This command displays all telemetry logs and operational states. When reporting problems or for troubleshooting, use this command as close to the problem time as possible and also provide the output of the **show running-config | section telemetry** command.

Example

The following is sample output from the **show telemetry internal diagnostics** command:

```
Device# show telemetry internal diagnostics

Using 'chassis active' in show commands for platform.
================================================================================

# show platform software trace message mdt-pubd chassis active R0 reverse


This command is being deprecated. Please use 'show logging process' command.
executing cmd on chassis 1 ...
Not enough available disk space in /bootflash to run this command.
Maximum used disk capacity of 90% for /bootflash exceeded. Aborting ...

================================================================================
Getting configuration database records.

URI = /services;serviceName=mdt/mdt_subscriptions;subscription_id=1
subscription_id: '1'
base.stream: 'NETCONF' (d)
base.filter_type: 'SUB_FILTER_TYPE_NONE' (d)
base.no_filter: '0' (d)
base.xpath: 'null'
base.encoding: 'encode-xml' (d)
base.update_trigger: 'SUB_UPD_TRIG_NONE' (d)
base.no_trigger: '0' (d)
base.period: 'null'
base.no_synch_on_start: 'null'
base.source_vrf: 'null'
base.source_address: 'null'
base.tdl_uri: 'null'
base.transform_name: 'null'
base.nested_uri: 'null'
base.rcvr_type: 'RCVR_TYPE_UNSPECIFIED' (d)
```

```
permanent: 'null'


URI = /services;serviceName=mdt/mdt_subscriptions;subscription_id=1/
mdt_receivers;address=0A010101;port=98
protocol: 'grpc-tcp'
parent_mdt_subscriptions_key: '1'
profile: 'null'
address: '10.1.1.1'
port: '98'


URI = /services;serviceName=mdt/mdt_named_protocol_rcvr;name=p1
name: 'p1'
protocol: 'null'
profile: 'null'
host.type: 'HOST_TYPE_UNSPECIFIED' (d)
host.unspecified: 'false' (d)
host.address: 'null'
host.hostname: 'null'
port: 'null'


URI = /services;serviceName=mdt/mdt_named_protocol_rcvr;name=proto1
name: 'proto1'
protocol: 'PROT_RCVR_TLS_NATIVE'
profile: 'abcd'
host.type: 'HOST_TYPE_HOSTNAME'
host.unspecified: 'null'
host.address: 'null'
host.hostname: 'ancd'
port: '9'

==============================================================================
Getting details for subscription 1...

# show telemetry ietf subscription 1 detail


Telemetry subscription detail:

  Subscription ID: 1
  Type: Configured
  State: Invalid
  Stream: NETCONF
  Filter:
    Filter type: not specified
    <none>
  Update policy:
    Update Trigger: not specified
    <none>
  Encoding: encode-xml
  Source VRF:
  Source Address:
  Notes: Stream not supported

  Legacy Receivers:
    Address                                 Port     Protocol         Protocol Profile

   ------------------------------------------------------------------------------------

    10.1.1.1                                98       grpc-tcp
```

```
# show telemetry ietf subscription 1 receiver


Telemetry subscription receivers detail:

  Subscription ID: 1
  Address: 10.1.1.1
  Port: 98
  Protocol: grpc-tcp
  Profile:
  Connection: 65535
  State: Invalid
  Explanation: Subscription stream invalid



# show telemetry internal sensor subscription 1



==============================================================================
Collecting internal connection information...

# show telemetry internal connection



==============================================================================
Collecting internal subscription information...

# show telemetry internal subscription all stats



==============================================================================
Collecting named receiver information...

  Name: p1
  Profile:
  State: Invalid
  Last State Change: 03/08/21 20:15:02
  Explanation: Value 'unspecified' not supported for parameter 'protocol'.
  Type: protocol
  Protocol: unspecified
  Host:
  Port: 0



  Name: proto1
  Profile: abcd
  State: Valid
  Last State Change: 03/08/21 03:06:47
  Explanation:
  Type: protocol
  Protocol: tls-native
  Host: ancd
  Port: 9


==============================================================================
Collecting stream sensor information...
```

```
# show telemetry internal sensor stream yang-push


# show telemetry internal sensor stream natve


# show telemetry internal sensor stream yang-notif-native


================================================================================

In addition: Please provide output of
"show running-config | section telemetry"

================================================================================
```

The output fields are self-explanatory.

show telemetry internal sensor

To display internal telemetry sensor information, use the **show telemetry internal sensor** command in privileged EXEC mode.

show telemetry internal sensor { **stream** *name* | **subscription** *id* }

Syntax Description

stream *name*	Displays telemetry stream information.
subscription *id*	Displays telemetry sensor subscription information.

Command Modes

Privileged EXEC #

Command History

Release	Modification
Cisco IOS XE Cupertino 17.7.1	This command was introduced.

Usage Guidelines

A sensor collects data from a single source. A single subscription might use multiple sensors, if the subscription data comes from multiple sources. This would typically happen when the XPath union operator is used in the subscription filter (for example /path1 or /path2).

A stream defines a set of events that can be subscribed to, and this set of events can be almost anything. For example, yang-push, yang-notif-native, and so on. The **stream** *name* keyword-argument pair in this command will display the sensors for all subscriptions on the specified stream.

Example

The following is sample output from the **show telemetry internal sensor subscription** command:

```
Device# show telemetry internal sensor subscription 2147483658

Subscription ID: 2147483658
Sensor Type: yang-push periodic
Filter type: xpath
Filter selector: /wireless-access-point-oper:access-point-oper-data/radio-oper-data/
    vap-oper-config/ssid
Data Collectors
DC: CEP periodic, SubFilter: /wireless-access-point-oper:access-point-oper-data/
    radio-oper-data/vap-oper-config/ssid
```

The table below lists the significant fields shown in the display.

Table 15: show telemetry internal sensor subscription Field Descriptions

Field	Description
Subscription ID	Subscription identifier.
Sensor Type	Type of sensor used for subscriptions.

Field	Description
Filter type	Type of filter used for subscriptions. Only XPath is supported.
Filter selector	The XPath that specifies the type of data to be sent by the subscription.
Data Collectors DC	Data collector used.

show telemetry internal subscription

To display internal telemetry subscription information, use the **show telemetry internal subscription** command in privileged EXEC mode.

show telemetry internal subscription { **all stats** | **id** *subscription-id* **stats** } [**connection** *ip-ipv6-address peer-port* [*vrf ip-ipv6-address*]]

Syntax Description

all	Displays all subscription information.
stats	Displays all subscription statistics.
id *subscription-id*	Displays information about the specified subscription ID.
connection	(Optional) Displays named receiver connection information.
ip-ipv6-address	(Optional) Peer IPv4 or IPv6 address.
peer-port	(Optional) Peer port number. Valid values are from 1 to 65535.
vrf	(Optional) Virtual routing and forwarding (VRF) name.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Usage Guidelines

If a subscription receiver is connected; but no updates are received, use this command to view whether the message drop count is incrementing.

Example

The following is sample output from the **show telemetry internal subscription all stats** command:

```
Device# show telemetry internal subscription all stats

Telemetry subscription stats:

Subscription ID  Msgs Sent  Msgs Drop  Records Sent Connection Info
---------------- ---------- ---------- ------------ ----------------------------------------
2147483651       2          0          0            admin
```

The output fields are self-explanatory.

show telemetry receiver

To display the state of all telemetry receivers, use the **show telemetry receiver** command in privileged EXEC mode.

show telemetry receiver { **all** | **name** *receiver-name* [**subscription**] }

Syntax Description

all	Displays information about all named receivers.
name *receiver-name*	Displays information about the specified receiver.
subscription	(Optional) Displays all subscriptions that use this named receiver.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.
Cisco IOS XE Cupertino 17.7.1	This command was modified. The **subscription** keyword was added.

Usage Guidelines

Named receiver objects have two different operational states, valid or invalid. If the state is invalid, the output of this command provides an explanation on why the receiver is invalid. When the receiver state is valid, this field is empty.

The output of the **subscription** keyword displays a table of all the subscriptions that use the specified receiver. The output of this command should match the output of the **show telemetry ietf subscription brief** command.

Example

The following is sample output from the **show telemetry receiver all** command:

```
Device# show telemetry receiver all

Telemetry receivers

  Name       <…>     Type        Profile               State       Explanation
  -----------<…>-------------------------------------------------------------------------
  receiver1  <…>     protocol    tls-trustpoint        Valid
```

The following is sample output from the **show telemetry receiver name** command:

```
Device# show telemetry receiver name receiver1

  Name: receiver1
  Profile: tls-trustpoint
  State: Valid
```

```
  Last State Change: 08/12/20 19:55:54
  Explanation:
  Type: protocol
  Protocol: tls-native
  Host: rcvr.test.com
  Port: 45000
```

The following is sample output from the **show telemetry receiver name subscription** command:

```
Device# show telemetry receiver name grpc-tcp subscription

  ID         Type         State      State Description
  1003       Configured   Valid
  1004       Configured   Valid
```

The output fields are self-explanatory.

Related Commands

Command	Description
receiver ip-address	Configures telemetry subscription.
receiver name	Configures a named receiver in a subscription.
show telemetry ietf subscription brief	Displays a brief summary of the subscription information.
telemetry receiver protocol	Configures a named protocol receiver.

source-address (telemetry)

To configure a source address for a subscription, use the **source-address** command in telemetry-subscription configuration mode. To remove the source address, use the **no** form of this command.

source-address { *ip-address* *ipv6-address* }
no source-address [*ip-address* *ipv6-address*]

Syntax Description

ip-address	IPv4 address of the source.
ipv6-address	IPv6 address of the source.

Command Default

Source address is not configured.

Command Modes

Telemetry subscription configuration (config-mdt-subs)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Example

The following example shows how to configure a source address for a subscription:

```
Device> enable
Device# configure terminal
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# source-address 2001:DB8::2
```

Related Commands

Command	Description
show telemetry receiver	Displays the state of all telemetry receivers.
telemetry ietf subscription	Configures telemetry subscription.

source-vrf (telemetry)

To configure a source virtual routing and forwarding (VRF) instance for a subscription, use the **source-vrf** command in telemetry-subscription configuration mode. To remove the source VRF instance, use the **no** form of this command.

source-vrf *vrf-name*
no source-vrf [*vrf-name*]

Syntax Description

vrf-name	Name of the VRF.

Command Default

Source VRF is not configured.

Command Modes

Telemetry subscription configuration (config-mdt-subs)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Usage Guidelines

Example

The following example shows how to configure a source VRF for a subscription:

```
Device> enable
Device# configure terminal
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# source-vrf vrf1
```

Related Commands

Command	Description
show telemetry receiver	Displays the state of all telemetry receivers.
telemetry ietf subscription	Configures telemetry subscription.

start (App Hosting)

To start or run an application, use the **start** command in application-hosting configuration mode. To stop the application, use the **no** form of this command.

start
no start

This command has no arguments or keywords.

Command Default

Starting of applications are not enabled.

Command Modes

Application-hosting configuration mode (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

You can either use the **start** command in privileged EXEC mode or the **app-hosting start appid** *application-name* command in application-hosting configuration mode.

To stop the app, you can either use the **no start** command in privileged EXEC mode or the **app-hosting stop appid** *application-name* command in application-hosting configuration mode.

Example

The following example shows how to start an application:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# start
Device(config-app-hosting)# end
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-hosting start appid *application-name*	Starts the application.

stream

To configure a telemetry stream for a subscription, use the **stream** command in telemetry-subscription configuration mode.

stream { **native** | **yang-notif-native** | **yang-push** }

Syntax Description

native	Configures a native stream.
yang-notif-native	Configures a YANG-NOTIF-NATIVE stream.
yang-push	Configures a YANG-push stream.

Command Modes

Telemetry-subscription configuration (config-mdt-subs)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.10.1	This command was introduced.
Cisco IOS XE Bengaluru 17.6.1	This command was modified. The **native**, and **yang-notif-native** keywords were added.

Usage Guidelines

Sources of telemetry data in a subscription are specified by the use of a stream and a filter. The term stream refers to a related set of events. RFC 5277 defines an event stream as a set of event notifications matching some forwarding criteria.

The *yang-notif-native* stream is any YANG notification in the publisher where the underlying source of events for the notification uses Cisco IOS XE native technology. This stream supports an XPath filter that specifies which notifications are of interest. Update notifications for this stream are sent only when events that the notifications are for occur.

The *yang-push* stream is the data in configuration and operational databases that is described by a supported YANG model. This stream supports an XPath filter to specify what data is of interest within the stream, and where the XPath expression is based on the YANG model that defines the data of interest. Update notifications for this stream may be sent either when data changes or at fixed periods, but not for both, for a given subscription. Subscriptions for data that does not currently exist are permitted, and these run as normal subscriptions.

Example

The following example shows how to configure a telemetry stream for a subscription:

```
Device> enable
Device# configure terminal
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# stream yang-push
```

Related Commands

Command	Description
telemetry ietf subscription	Configures telemetry subscription.

telemetry ietf subscription

To configure telemetry subscription, use the **telemetry ietf subscription** command in global configuration mode. To disable the configuration, use the **no** form of this command.

telemetry ietf { **subscription** *sub-id* }
no telemetry ietf { **subscription** *sub-id* }

Syntax Description

subscription *sub-id*	Configures a telemetry subscription. Valid values are from 0 to 2147483647.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.10.1	This command was introduced.

Example

The following example shows how to configure an telemetry subscription:

```
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)#
```

telemetry protocol grpc profile

To configure a profile for the Google Remote Procedure Call (gRPC) telemetry connection, use the **telemetry protocol grpc profile** command in global configuration mode. To remove the profile, use the **no** form of this command.

telemetry protocol grpc profile *profile-name*
no telemetry protocol grpc profile *profile-name*

Syntax Description

profile-name	Name of the Certificate Authority (CA) trustpoint.

Command Default

The profile for the gRPC telemetry protocol is enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Cupertino 17.9.1	This command was introduced.

Usage Guidelines

To use the client ID certificate for mutual authentication, when using the gRPC-TLS protocol, a new gRPC-TLS profile that contains a pair of trustpoints is added to the telemetry configuration.

If the server is configured to require mutual authentication, and there is no client ID trustpoint in the profile, the client authentication will not happen, nor will the connection succeed.

Example

The following example shows how to configure a profile for a gRPC telemetry connection:

```
Device> enable
Device# configure terminal
Device(config)# telemetry protocol grpc profile myprofile
Device(config-mdt-protocol-grpc-profile)#
```

Related Commands

Command	Description
ca-trustpoint	Configures the server CA trustpoint for a gRPC telemetry connection.
id-trustpoint	Configures a client ID trustpoint for a gRPC telemetry connection.

telemetry receiver protocol

To configure a named protocol receiver, use the **telemetry receiver protocol** command in global configuration mode. To remove a named protocol receiver, use the **no** form of this command.

telemetry receiver protocol *receiver-name*
no telemetry receiver protocol *receiver-name*

Syntax Description

receiver-name	Name of the receiver by which it is identified in the system.

Command Default

A named protocol receiver is not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	This command was introduced.

Usage Guidelines

Named protocol receivers are used to specify telemetry transports that use protocols.

When a named protocol receiver is created, it is not automatically connected to the receiver. The named protocol receiver must be requested by at least one subscription to create a connection to the receiver.

After you configure the **telemetry receiver protocol** command, the command mode changes to telemetry protocol-receiver configuration mode. You can configure the host and protocol name for the named receiver in this mode.

Example

The following example shows how to configure a named protocol receiver:

```
Device> enable
Device# configure terminal
Device(config)# telemetry receiver protocol my-receiver
Device(config-mdt-protocol-receiver)#
```

Related Commands

Command	Description
host	Specifies named receiver host details.
protocol	Specifies a protocol for the named receiver.
show telemetry receiver	Displays the state of all telemetry receivers.

update-policy

To configure an update policy for a subscription, use the **update-policy** command in telemetry-subscription configuration mode.

update-policy {**on-change** | **periodic** *period*}

Syntax Description

on-change	Enables on-change updates.
periodic *period*	Enable periodic updates. Valid values are from 100 to 4294967295.

Command Default

Update policy is not configured.

Command Modes

Telemetry-subscription configuration (config-mdt-subs)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.10.1	This command was introduced.

Example

The following example shows how to configure a periodic update policy for a subscription:

```
Device> enable
Device# configure terminal
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# update-policy periodic 6000
Device(config-mdt-subs)#
```

The following example shows how to configure an on-change update policy for a subscription:

```
Device> enable
Device# configure terminal
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# update-policy on-change 4000
Device(config-update-onchange)#
```

Related Commands

Command	Description
telemetry ietf subscription	Configures telemetry subscription.

vcpu (App Hosting)

To change the virtual CPU (vCPU) allocated by the application, use the **vcpu** command in custom application resource profile configuration mode. To revert to the application-provided CPU quota, use the **no** form of this command.

vcpu *number*
no vcpu {[*number*]}

Syntax Description

number	The vCPU count. Valid values are from 0 to 65535.

Command Default

Command Modes

Custom application resource profile configuration (config-app-resource-profile-custom)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.1	This command was introduced.

Usage Guidelines

Within each application package, an application-specific resource profile is provided that defines the recommended CPU load, memory size, and number of virtual CPUs (vCPUs) required for the application. Use this command to change the allocation of resources for specific processes in the custom resource profile.

Reserved resources specified in the application package can be changed by setting a custom resource profile. Only the CPU, memory, and vCPU resources can be changed. For the resource changes to take effect, stop and deactivate the application, then activate it and start it again.

Note Resource values are application-specific, and any adjustment to these values must ensure that the application can run reliably with the changes.

Example

The following example shows how to override the application-provided vCPU quota using a custom resource profile:

```
Device# configure terminal
Device(config)# app-hosting appid lxc_app
Device(config-app-hosting)# app-resource profile custom
Device(config-app-resource-profile-custom)# vcpu 2
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-resource profile	Overrides the application-provided resource profile.

vlan (App Hosting)

To configure a VLAN guest interface and enter application-hosting VLAN-access IP configuration mode, use the **vlan** command in application-hosting VLAN-access configuration mode. To remove the configuration, use the **no** form of this command.

vlan *vlan-ID* **guest-interface** *interface-number*
no vlan *vlan-ID* **guest-interface** *interface-number*

Syntax Description

vlan-ID	VLAN ID of the front-panel port. Valid values are from 0 to 4094.
guest-interface *interface-number*	Configures the guest interface. Valid values are for the *interface-number* argument are from 0 to 63.

Command Default

Guest interface is not configured.

Command Modes

Application-hosting trunk configuration (config-app-hosting-trunk)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

When using the front-panel port as a VLAN interface, the application is connected to a specific VLAN network. A VLAN interface is created on the host and it is associated with the front-panel port *eth0* interface.

Example

The following example shows how to configure a guest-interface for a front-panel trunk port:

```
Device# configure terminal
Device(config)# app-hosting appid lxc_app
Device(config-app-hosting)# app-vnic AppGigabitEthernet trunk
Device(config-config-app-hosting-trunk)# vlan 1 guest-interface 9
Device(config-config-app-hosting-vlan-access-ip)# guest-ipaddress 192.168.0.1
netmask 255.255.255.0
Device(config-config-app-hosting-vlan-access-ip)# end
```

Related Commands

Command	Description
app-hosting appid	Configures an application and enters application hosting configuration mode.
app-vnic AppGigabitEthernet trunk	Configures a front-panel trunk port for application hosting and enters application-hosting trunk configuration mode.
guest-ipaddress	Configures a guest IP address.

vnic gateway

To configure a gateway for a virtual network interface (vNIC), use the **vnic gateway** command in application hosting configuration mode. To remove the configuration, use the **no** form of this command.

vnic gateway VirtualPortGroup *number* **guest-interface** *network-interface* [**guest-ipaddress** *ip-address*]**netmask** *netmask* **gateway** *ip-address* [**name-server** *ip-address*] [**default**]
no vnic gateway [**VirtualPortGroup** *number* **guest-interface** *network-interface*]

Syntax Description

VirtualPortGroup *number*	Configures a VirtualPortGroup interface for the gateway.
guest-interface *network-interface*	Configures a guest interface for the gateway.
guest-ipaddress *ip-address*	(Optional) Configures an IP address for the guest interface.
netmask *netmask*	(Optional) Specifies the subnet mask for the guest IP address.
gateway *ip-address*	(Optional) Configures an IP address for the vNIC gateway.
name-server *ip-address*	(Optional) Configures an IP address for the Domain Name System (DNS) server.
default	(Optional) Configures the default gateway.

Command Default

vNIC gateway is not configured.

Command Modes

Application hosting configuration (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Example

The following example shows how to configure a vNIC gateway:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# vnic gateway1 VirtualPortGroup 0 guest-interface 1
guest-ipaddress 10.0.0.3 netmask 255.255.255.0 gateway 10.0.0.1 name-server 10.2.2.2
```

Related Commands

Command	Description
app-hosting appid	Enables application hosting and enters application hosting configuration mode.

vnic management

To configure an application management network for a virtual network interface (vNIC), use the **vnic management** command in application hosting configuration mode. To remove the configuration, use the **no** form of this command.

vnicmanagementguest-interface *network-interface* {**guest-ipaddress** *ip-address*} **netmask** *netmask* **gateway** *ip-address* [**name-server** *ip-address*] [**default**]
no vnic management [**guest-interface** *network-interface*]

Syntax Description

guest-interface *network-interface*	Configures a guest interface for the gateway.
guest-ipaddress *ip-address*	(Optional) Configures an IP address for the guest interface.
netmask *netmask*	(Optional) Specifies the subnet mask for the guest IP address.
gateway *ip-address*	(Optional) Configures an IP address for the vNIC gateway.
name-server *ip-address*	(Optional) Configures an IP address for the Domain Name System (DNS) server.
default	(Optional) Configures the default gateway.

Command Default

An application management network is not configured.

Command Modes

Application hosting configuration (config-app-hosting)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.12.1	This command was introduced.

Usage Guidelines

Example

The following example shows how to configure a vNIC application management network:

```
Device# configure terminal
Device(config)# app-hosting appid iox_app
Device(config-app-hosting)# vnic management guest-interface 0 guest-ipaddress
172.19.0.24 netmask 255.255.255.0 gateway 172.19.0.23 default
```

Command	Description
app-hosting appid	Enables application hosting and enters application hosting configuration mode.

yang-interfaces aaa

To configure a method-list for authentication, authorization, and accounting (AAA), use the **yang-interfaces aaa** command in global configuration mode. To remove the AAA method-list, use the **no** form of this command.

yang-interfaces aaa { **authentication** | **authorization** } **method-list** *method-list-name*
no yang-interfaces aaa { **authentication** | **authorization** } **method-list** *method-list-name*

Syntax Description

authentication	Configures authentication.
authorization	Configures authorization.
method-list *named-method-list*	Configures a named method-list.

Command Default

The default method list is configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Cupertino 17.9.1	This command was introduced.

Usage Guidelines

A method list is a named list that describes the authorization methods to be queried, such as, AAA, Lightweight Directory Access Protocol (LDAP), RADIUS, or TACACS+. Method lists defines the method and the sequence in which authorization is performed. Method lists enables one or more security protocols for authorization, ensuring a backup system in case of a failure. Both the default method-list and named method-lists are supported.

Method lists are processed by the Cisco IOS software serially. If the first configured method-list fails, the next one is processed. This process continues until a successful authentication or authorization, or until all configured methods are exhausted. Named method-lists are supported on gNMI, NETCONF, and RESTCONF interfaces.

Example

The following example shows how to configure a named method-list:

```
Device> enable
Device# configure terminal
Device(config)# netconf-yang
Device(config)# yang-interfaces aaa authentication method-list netconf-authn
Device(config)# yang-interfaces aaa authorization method-list netconf-authr
Device(config)# end
```

Related Commands

Command	Description
gnxi	Starts the gNxI process.
netconf-yang	Enables NETCONF-YANG.

Command	Description
restconf	Enables the RESTCONF interface on a device.

yang-interfaces feature deprecated disable

To disable all deprecated YANG models, use the **yang-interfaces feature deprecated disable** command in global configuration mode. To re-enable the deprecated YANG models, use the **no** form of this command.

yang-interfaces feature deprecated disable
no yang-interfaces feature deprecated disable

This command has no arguments or keywords.

Command Default

All deprecated models are enabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE 17.15.1	This command was introduced.

Usage Guidelines

Deprecated YANG models can also be disabled by using the Cisco-IOS-XE-Yang-interfaces-cfg.yang model.

Example

The following example shows how to disable deprecated YANG models:

```
Device> enable
Device# configure terminal
Device(config)# yang-interfaces feature deprecated disabled
Device(config)#
```