Programmability

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 lxc_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.

The IOx application must be copied to the local device storage medium using the IOS **copy** command, and then the **app-hosting install** command must be enabled from the local storage to enable application hosting.

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

When the **activate** keyword is enabled, it 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.

During the uninstallation of the application, all packaging 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 lxc_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 lxc_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.

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 lxc_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

To configure a virtual network interface gateway of 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 of 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 lxc_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 AppGigEthernet

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

app-vnic AppGigEthernet {**trunk** | **vlan-access**}
no app-vnic AppGigEthernet {**trunk** | **vlan-access**}

Syntax Description

trunk	Configures the front-panel trunk port for application hosting.
vlan-access	Configures the front-panel VLAN 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 and Cisco Catalyst 9400 Series Switches support front-panel VLAN or 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.

When you configure the **vlan-access** keyword, the command mode changes to application-hosting VLAN-access configuration mode. And when you configure the **trunk** keyword, the command mode changes to application-hosting trunk configuration mode.

Example

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

```
Device# configure terminal
Device(config)# app-hosting appid lxc_app
Device(config-app-hosting)# app-vnic AppGigEthernet vlan-access
Device(config-config-app-hosting-vlan-access)#
```

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

```
Device# configure terminal
Device(config)# app-hosting appid lxc_app
Device(config-app-hosting)# app-vnic AppGigEthernet 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
```

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.

controller (OpenFlow)

To connect to an OpenFlow controller, use the **controller** command in OpenFlow switch configuration mode. To disconnect an OpenFlow controller, use the **no** form of this command.

controller ipv4 *controller-address* [{**port** [{*port-number*}]}][{**security** {**none** | **tls**}}][{**vrf** [{*vrf-name*}]}]
no controller ipv4 *controller-address* [{**port** [{*port-number*}]}][{**security** {**none** | **tls**}}][{**vrf** [{*vrf-name*}]}]

Syntax Description

ipv4 *controller-address*	Configures the IP address of the OpenFlow controller.
port *port-number*	(Optional) Configures the OpenFlow controller TCP port. The default is 6653.
security	(Optional) Configures the OpenFlow controller connection security.
none	(Optional) Configures no authentication or encryption for the controller.
tls	(Optional) Configures the Transport Layer Security (TLS) protocol for the controller.
vrf *vrf-name*	(Optional) Configures a virtual routing and forwarding (VRF) instance for the OpenFlow controller.

Command Default

The controller is not configured.

Command Modes

OpenFlow switch configuration (config-openflow-switch)

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

Usage Guidelines

The OpenFlow controller is an entity that interacts with the OpenFlow switch using the OpenFlow protocol. In most cases, an OpenFlow controller is a software that controls many OpenFlow logical switches. OpenFlow controllers offer a centralized view of the network, and enable administrators to dictate to the underlying systems (switches and routers) on how to handle the network traffic.

Example

The following example shows how to configure an OpenFlow controller:

```
Device# configure terminal
Device(config)# feature openflow
Device(config)# openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)# controller ipv4
10.2.2.2 port 6633 vrf Mgmt-vrf security none
```

Related Commands

Command	Description
feature openflow	Enables the OpenFlow feature.
openflow	Enables OpenFlow configuration and enters OpenFlow configuration mode.
switch	Configures a logical switch and enters OpenFlow switch configuration 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 lxc_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.

datapath-id

To set the OpenFlow logical switch ID, use the **datapath-id** command in OpenFlow switch configuration mode. To disable the logical switch ID, use the **no** form of this command.

datapath-id *value*
no datapath-id

Syntax Description

value	A 64-bit hexadecimal value in the range 0x1 to 0xffffffffffffffff.

Command Default

Datapath ID is not configured.

Command Modes

OpenFlow switch configuration mode (config-openflow-switch)

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

Usage Guidelines

Example

The following example shows how to configure a logical switch ID:

```
Device# configure terminal
Device(config)# feature openflow
Device(config)# openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)# datapath 0x12345678
```

Related Commands

Command	Description
feature openflow	Enables the OpenFlow feature.
openflow	Enables OpenFlow configuration and enters OpenFlow configuration mode.
switch	Configures a logical switch and enters OpenFlow switch configuration mode.

dataplane-default

To set the OpenFlow configuration in the dataplane, use the **dataplane-default** command in OpenFlow switch configuration mode. To disable the settings, use the **no** form of this command.

dataplane-default {**secure** | **standalone**}
no dataplane-default {**secure** | **standalone**}

Syntax Description

secure	Configures the dropping of packets that are intended to the controller. This is the default.
standalone	Configures the forwarding of packets that are intended to the controller.

Command Default

Packets are dropped.

Command Modes

OpenFlow switch configuration (config-openflow-switch)

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

Usage Guidelines

Example

The following example shows how to configure the dropping of packets that are intended for the controller:

```
Device# configure terminal
Device(config)# feature openflow
Device(config)# openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)# dataplane-default secure
```

Related Commands

Command	Description
feature openflow	Enables the OpenFlow feature.
openflow	Enables OpenFlow configuration and enters OpenFlow configuration mode.
switch(OpenFlow)	Configures a logical switch and enters OpenFlow switch configuration mode.

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
```

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.

encoding

To configure telemetry encoding for the subscription, use the **encoding** command in telemetry-subscription configuration mode. To disable the configuration, use the **no** form of this command.

encoding encode-kvgpb
no encoding encode-kvgpb

Syntax Description

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

Command Modes

Telemetry-subscription configuration (config)

Command History

Release	Modification
Cisco IOS XE Gibraltor 16.10.1	This command was introduced.

Example

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

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

feature openflow

To enable the OpenFlow feature, use the **feature openflow** command in global configuration mode. To disable the OpenFlow feature, use the **no** form of this command.

feature openflow
no feature openflow

This command has no arguments or keywords.

Command Default

OpenFlow is not configured.

Command Modes

Global configuration (config)

Command History

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

Usage Guidelines

Before configuring this command, you must configure the **boot mode openflow** command to enable OpenFlow forwarding mode on your device.

Example

The following example shows how to enable the OpenFlow configuration:

```
Device# configure terminal
Device(config)# feature openflow
Device(config)#
```

Related Commands

Command	Description
boot mode openflow	Enables OpenFlow forwarding mode.

filter xpath

To configure XPath filter, use the **filter xpath** command in telemetry-subscription configuration mode. To disable the configuration, use the **no** form of this command.

filter xpath *path*
nofilter xpath *path*

Syntax Description

path	Specifies XPath filter.

Command Modes

Telemetry-subscription configuration (config)

Command History

Release	Modification
Cisco IOS XE Gibraltor 16.10.1	This command was introduced.

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

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 AppGigEthernet 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 lxc_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 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
```

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 lxc_app
Device(config-app-hosting)# app-vnic AppGigEthernet vlan-access
Device(config-config-app-hosting-vlan-access)# 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 AppGigEthernet vlan-access	Configures a VLAN port as the front-panel port for an application, and enters application-hosting VLAN-access 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.

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.

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.

logging flow-modify

To enable error logs for flows, use the **logging flow-modify** command in OpenFlow switch configuration mode. To disable logging, use the **no** form of this command.

logging flow-modify
no logging flow-modify

This command has no arguments or keywords.

Command Default

Error logging is not configured.

Command Modes

OpenFlow switch configuration (config-openflow-switch)

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

Usage Guidelines

Example

The following example shows how to enable logging for flows:

```
Device# configure terminal
Device(config)# feature openflow
Device(config)# openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)# logging flow-modify
```

Related Commands

Command	Description
feature openflow	Enables the OpenFlow feature.
openflow	Enables OpenFlow configuration and enters OpenFlow configuration mode.
switch(OpenFlow)	Configures a logical switch and enters OpenFlow switch 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 lxc_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.

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
```

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 lxc_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-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-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-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>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 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.

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, 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.
len	

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:dead:beef:cafe::5 6 1000

Ping host 2001:dead:beef:cafe::5, 6 times, 1000 bytes
Pinging 2001:dead:beef:cafe::5 ... reply in 0 ms
Pinging 2001:dead:beef:cafe::5 ... reply in 1 ms
Pinging 2001:dead:beef:cafe::5 ... reply in 1 ms
Pinging 2001:dead:beef:cafe::5 ... reply in 0 ms
Pinging 2001:dead:beef:cafe::5 ... reply in 0 ms
Pinging 2001:dead:beef:cafe::5 ... reply in 0 ms
```

Related Commands

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

probe-interval

To configure the OpenFlow probe interval, use the **probe-interval** command in OpenFlow switch configuration mode. To disable the probe interval, use the **no** form of this command.

probe-interval *seconds*
no probe-interval

Syntax Description

seconds	Probe interval in seconds. The default is 5.

Command Default

5 seconds

Command Modes

OpenFlow switch configuration (config-openflow-switch)

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

Usage Guidelines

Example

The following example shows how to configure the probe interval:

```
Device# configure terminal
Device(config)# feature openflow
Device(config)# openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)# probe-interval 10
```

Related Commands

Command	Description
feature openflow	Enables the OpenFlow feature.
openflow	Enables OpenFlow configuration and enters OpenFlow configuration mode.
switch(OpenFlow)	Configures a logical switch and enters OpenFlow switch configuration mode.

protocol-version

To configure an OpenFlow protocol to connect to the controller, use the **protocol-version** command in OpenFlow switch configuration mode. To disable the selected protocol, use the **no** form of this command.

protocol-version {**1.0** | **1.3** | **negotiate**}
no protocol-version {**1.0** | **1.3** | **negotiate**}

Syntax Description

1.0	Configures OpenFlow 1.0 protocol to connect to the controller.
1.3	Configures OpenFlow 1.3 protocol to connect to the controller.
negotiate	Configures protocol negotiation with the controller.

Command Default

Protocol is not configured.

Command Modes

OpenFlow switch configuration (config-openflow-switch)

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

Usage Guidelines

Example

The following example shows how to configure an OpenFlow protocol:

```
Device# configure terminal
Device(config)# feature openflow
Device(config)# openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)# protocol-version 1.3
```

Related Commands

Command	Description
feature openflow	Enables the OpenFlow feature.
openflow	Enables OpenFlow configuration and enters OpenFlow configuration mode.
switch(OpenFlow)	Configures a logical switch and enters OpenFlow switch configuration mode.

receiver

To configure 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 receiver IP address.
ipv4-address **|** *ipv6-address*	IPv4 or IPv6 receiver adress.
port	Configures receiver port.
protocol*protocol*	Configures protocol for notification. The following protocol is supported: • **grpc-tcp**

Command Modes

Telemetry-subscription configuration (config)

Command History

Release	Modification
Cisco IOS XE Gibraltor 16.10.1	This command was introduced.

Usage Guidelines

A receiver is a network element that receives the telemetry data. Configured subscriptions are created by management operations on the publisher by controllers, and explicitly include the specification of the receiver of the telemetry data defined by the subscription. These subscriptions persist across reboots of the publisher.

Configured subscriptions can be configured with multiple receivers, however; only the first valid receiver is used. Other receivers are not be tried, if a receiver is already connected, or in the process of being connected. If that receiver is deleted, another receiver is connected.

Example

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

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

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 Fuji 16.8.1	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 lxc_app
Device(config-app-hosting)# resource profile custom cpu 7400 memory 2048 vcpu 2
```

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.

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 lxc_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 AppGigEthernet

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

show controller ethernet-controller AppGigEthernet *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 AppGigEthernet** *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 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 1: 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.

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.

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

After configuring this command, you can update the application hosting configuration.

Example

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

```
Device# show iox-service

IOx Infrastructure   Summary:
------------------------------
IOx service (CAF)    : Running
IOx service (HA)     : Running
IOx service (IOxman) : Running
Libvirtd             : Running
```

The table below lists the fields shown in the display.

Table 2: show iox-service Field Descriptions

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

Field	Description
Libvirtd	Status of the Linux Library Virtual daemon.

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

```
Device# show iox-service detail

IOx Infrastructure Summary:
---------------------------
IOx service (CAF)    : Running
IOx service (HA)     : Running
IOx service (IOxman) : Running
Libvirtd             : Running



------------------ show platform software process list switch active r0 name caf
------------------
Name: run_ioxn_caf.sh
  Process id       : 28445
  Parent process id: 28155
  Group id         : 28445
  Status           : S
  Session id       : 9123
  User time        : 5
  Kernel time      : 2
  Priority         : 20
  Virtual bytes    : 19939328
  Resident pages   : 1036
  Resident limit   : 18446744073709551615
  Minor page faults: 4833
  Major page faults: 0




------------------ show platform software process list switch active r0 name libvirtd
------------------
Name: libvirtd.sh
  Process id       : 5757
  Parent process id: 1
  Group id         : 5757
  Status           : S
  Session id       : 5757
  User time        : 0
  Kernel time      : 0
  Priority         : 20
  Virtual bytes    : 18661376
  Resident pages   : 692
  Resident limit   : 18446744073709551615
  Minor page faults: 208
  Major page faults: 0

Name: libvirtd
  Process id       : 5782
  Parent process id: 5757
  Group id         : 5757
  Status           : S
  Session id       : 5757
  User time        : 6
  Kernel time      : 74
  Priority         : 20
```

```
Virtual bytes    : 883945472
Resident pages   : 2122
Resident limit   : 18446744073709551615
Minor page faults: 2398
Major page faults: 59
```

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-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 3: 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 4: 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 5: 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 openflow hardware capabilities

To display information about OpenFlow hardware capabilities, use the **show openflow hardware capabilities** command in privileged EXEC mode.

show openflow hardware capabilities [{**pipeline 1**}]

Syntax Description

pipeline 1	Displays information about the OpenFlow pipeline ID.

Command Modes

Privileged EXEC (#)

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

Usage Guidelines

Example

The following is sample output from the **show openflow hardware capabilities** command:

```
Device# show openflow hardware capabilities

Max Interfaces: 1000
Aggregated Statistics: YES
Pipeline ID: 1
Pipeline Max Flows: 2322
Max Flow Batch Size: 100
Statistics Max Polling Rate (flows/sec): 10000
Pipeline Default Statistics Collect Interval: 5
Flow table ID: 0
Max Flow Batch Size: 100
Max Flows: 1022
Bind Subintfs: FALSE
Primary Table: TRUE
Table Programmable: TRUE
Miss Programmable: TRUE
Number of goto tables: 1
Goto table id: 1
Number of miss goto tables: 1
Miss Goto table id: 1
Stats collection time for full table (sec): 1
!
!
!
```

The following is sample output from the **show openflow hardware capabilities pipeline 1** command:

```
Device# show openflow hardware capabilities pipeline 1

  Max Interfaces: 1000
  Aggregated Statistics: YES

  Pipeline ID: 1
    Pipeline Max Flows: 128
    Max Flow Batch Size: 100
    Statistics Max Polling Rate (flows/sec): 10000
```

```
   Pipeline Default Statistics Collect Interval: 5

   Flow table ID: 0

   Max Flow Batch Size: 100
   Max Flows: 32
   Bind Subintfs: FALSE
   Primary Table: TRUE
   Table Programmable: TRUE
   Miss Programmable: TRUE
   Number of goto tables: 1
   Goto table id: 1
   Number of miss goto tables: 1
   Miss Goto table id: 1
   Stats collection time for full table (sec): 1

       Match Capabilities   Match Types
--------------------------------------  -----------
                     ethernet mac destination      bitmask
                                ethernet type     optional
                                      VLAN ID     optional
                in port (virtual or physical)     optional

Actions              Count Limit  Order
--------------------------------------  -----------  -----
                                  set vlan id          1           10
                                push vlan tag          1           10
                                 pop vlan tag          1           10
                                  drop packet          1           10
perform another lookup in the specified table          1           10
           forward pkt via the specific group          1           10
                          specified interface         64           10
                                   controller          1           10
                               set input port          1           10

Miss actions         Count Limit  Order
--------------------------------------  -----------  -----
                                  set vlan id          1           10
                                push vlan tag          1           10
                                 pop vlan tag          1           10
                                  drop packet          1           10
perform another lookup in the specified table          1           10
           forward pkt via the specific group          1           10
                          specified interface         64           10
                                   controller          1           10
                               set input port          1           10


   Flow table ID: 1

   Max Flow Batch Size: 100
   Max Flows: 32
   Bind Subintfs: FALSE
   Primary Table: FALSE
   Table Programmable: TRUE
   Miss Programmable: TRUE
   Number of goto tables: 2
   Goto table id: 2 3
   Number of miss goto tables: 1
   Miss Goto table id: 2
   Stats collection time for full table (sec): 1

       Match Capabilities   Match Types
--------------------------------------  -----------
```

```
                      ethernet mac destination      bitmask
                           ethernet mac source     optional
                                 ethernet type     optional
                                       VLAN ID     optional
                 in port (virtual or physical)     optional

 Actions              Count Limit  Order
 -------------------------------------  -----------  -----
                       set eth destination mac          1           10
                                   set vlan id          1           10
                                 push vlan tag          1           10
                                  pop vlan tag          1           10
                                   drop packet          1           10
 perform another lookup in the specified table          1           10
            forward pkt via the specific group          1           10
                           specified interface         64           10
                                    controller          1           10
                                set input port          1           10
!
!
!
```

The output fields are self-explanatory.

show openflow interface

To display information about OpenFlow interfaces, use the **show openflow interface** command in privileged EXEC mode.

show openflow interface{[**detail**]}

Syntax Description

detail	Displays detailed administrative and operational state information.

Command Modes

Privileged EXEC(#)

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

Usage Guidelines

Example

The following is sample output from the **show openfow interface detail** command:

```
Device# show openflow interface detail

GigabitEthernet1/0/1, admin up, oper up
 GigabitEthernet1/0/2, admin up, oper up
 GigabitEthernet1/0/3, admin up, oper up
 GigabitEthernet1/0/4, admin up, oper up
 GigabitEthernet1/0/5, admin up, oper down
 GigabitEthernet1/0/6, admin up, oper down
 GigabitEthernet1/0/7, admin up, oper down
 GigabitEthernet1/0/8, admin up, oper down
 GigabitEthernet1/0/9, admin up, oper up
 GigabitEthernet1/0/10, admin up, oper up
 GigabitEthernet1/0/11, admin up, oper up
 GigabitEthernet1/0/12, admin up, oper up
 GigabitEthernet1/0/13, admin up, oper down
 GigabitEthernet1/0/14, admin up, oper down
 GigabitEthernet1/0/15, admin up, oper down
 GigabitEthernet1/0/16, admin up, oper down
 GigabitEthernet1/0/17, admin up, oper down
 GigabitEthernet1/0/18, admin up, oper down
 GigabitEthernet1/0/19, admin up, oper up
 GigabitEthernet1/0/20, admin up, oper up
 GigabitEthernet1/0/21, admin up, oper up
 GigabitEthernet1/0/22, admin up, oper up
 GigabitEthernet1/0/23, admin up, oper down
 GigabitEthernet1/0/24, admin up, oper down
 GigabitEthernet1/1/1, admin up, oper down
 GigabitEthernet1/1/2, admin up, oper down
 GigabitEthernet1/1/3, admin up, oper down
 GigabitEthernet1/1/4, admin up, oper down
 TenGigabitEthernet1/1/1, admin up, oper down
 TenGigabitEthernet1/1/2, admin up, oper down
 TenGigabitEthernet1/1/3, admin up, oper down
 TenGigabitEthernet1/1/4, admin up, oper down
 TenGigabitEthernet1/1/5, admin up, oper down
 TenGigabitEthernet1/1/6, admin up, oper down
```

```
TenGigabitEthernet1/1/7, admin up, oper down
TenGigabitEthernet1/1/8, admin up, oper down
FortyGigabitEthernet1/1/1, admin up, oper down
FortyGigabitEthernet1/1/2, admin up, oper down
TwentyFiveGigE1/1/1, admin up, oper down
TwentyFiveGigE1/1/2, admin up, oper down
```

The output fields are self-explanatory.

show openflow switch flows

To display OpenFlow switch flows, use the **show openflow switch flows** command in privileged EXEC mode.

show openflow switch *number* **flows** [{**brief** | **list** | **summary** }][{{[**controller**] | **default** | **del-pending** | **fixed** | **pending**} | [{**brief** | **list** | **summary** }]}]

Syntax Description

number	OpenFlow switch number.
brief	(Optional) Displays brief information about OpenFlow switch flows.
list	Displays all flows; one flow entry per line.
summary	Displays the count of flows.
configured	Displays information about the configured flows.
controller	Displayed information about the controller-programmed flows.
default	Displays information about the default flows.
del-pending	Displays information about flows that are scheduled to be deleted.
fixed	Displays information about fixed flows.
pending	Displays all pending flows.

Command Modes

Privileged EXEC (#)

Command History

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

Example

The following is sample output from the **show openflow switch** *number* **flows brief** command:

```
Device# show openflow switch 1 flows brief

Logical Switch Id: 1
Total flows: 10

Flow: 1 Match: any Actions: drop
```

```
      Priority: 0, Table: 0, Cookie: 0x0, Duration: 140.088s, Packets: 2, Bytes: 500

Flow: 2 Match: tcp,in_port=0,tp_src=1 Actions: output:1
      Priority: 11111, Table: 0, Cookie: 0x1, Duration: 130.642s, Packets: 0, Bytes: 0

Flow: 3 Match: any Actions: drop
      Priority: 0, Table: 1, Cookie: 0x0, Duration: 140.088s, Packets: 0, Bytes: 0

Flow: 4 Match: any Actions: drop
      Priority: 0, Table: 2, Cookie: 0x0, Duration: 140.088s, Packets: 0, Bytes: 0

Flow: 5 Match: any Actions: drop
      Priority: 0, Table: 3, Cookie: 0x0, Duration: 140.087s, Packets: 0, Bytes: 0

Flow: 6 Match: any Actions: drop
      Priority: 0, Table: 4, Cookie: 0x0, Duration: 140.087s, Packets: 0, Bytes: 0

Flow: 7 Match: any Actions: drop
      Priority: 0, Table: 5, Cookie: 0x0, Duration: 140.086s, Packets: 0, Bytes: 0

Flow: 8 Match: any Actions: drop
      Priority: 0, Table: 6, Cookie: 0x0, Duration: 140.086s, Packets: 0, Bytes: 0

Flow: 9 Match: any Actions: drop
      Priority: 0, Table: 7, Cookie: 0x0, Duration: 140.085s, Packets: 0, Bytes: 0

Flow: 10 Match: any Actions: drop
      Priority: 0, Table: 8, Cookie: 0x0, Duration: 140.085s, Packets: 0, Bytes: 0

Device#
```

The following is sample from the **show openflow switch** *number* **flows summary** command:

```
Device# show openflow switch 1 flows summary

Logical Switch Id: 1
Switch flow count: 10
```

The output fields are self-explanatory.

Related Commands

Command	Description
feature openflow	Enables the OpenFlow feature.
openflow	Enables OpenFlow configuration and enters OpenFlow configuration mode.

show platform yang-management process

To display the status of the software processes required to support NETCONF-YANG, use the **show platform yang management process** in privileged EXEC mode.

show platform 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.

Usage Guidelines

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 6: 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 inteface (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 7: 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

show telemetry ietf subscription

To display information about telemetry subscriptions on a device, use the **show telemetry ietf subscription** command in user EXEC or privileged EXEC mode.

show telemetry ietf subscription {*subscription-ID* | **all** | **configured** | **dynamic**} [{**brief** | **detail** | **receiver** }]

Syntax Description

subscription-ID	Subscription ID. Valid values are from 0 to 4294967295.
all	Displays all subscription information.
configured	Displays a list of subscriptions configured through the CLI or NETCONF set config.
dynamic	Displays information about dynamic subscriptions created using the <establish-subscription> RPC.
brief	(Optional) Displays a brief summary of the subscription information.
detail	(Optional) Displays the subscription information in detail.
receiver	(Optional) Displays the receiver details for a subscription, including the IP address and port of the remote client, the transport protocol, and the connection state (connected, disconnected, or connecting).

Command Modes

User EXEC (>)

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.

Usage Guidelines

Use the **show telemetry ietf subscription** command or the <get> RPC to retrieve the list of current subscription details on a device.

Example

The following is sample output from the **show telemetry ietf subscription** *subscription-ID* **detail** command:

```
Device#  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 table below lists the significant fields shown in the display.

Table 8: show telemetry ietf subscription Field Descriptions

Field	Description
Subscription ID	Subscription identifier.

Field	Description
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 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 is always be empty.
ID	Subscription ID.

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.

statistics collection-interval

To configure the collection interval for OpenFlow flow statistics, use the **statistics collection-interval** command in OpenFlow switch configuration mode. To disable the collection interval, use the **no** form of this command.

statistics collection-interval *collection-interval*
no statistics collection-interval

Syntax Description

collection-interval	Flow statistics collection interval in seconds. Valid values are from 0 to 10.

Command Default

Collection interval is not set.

Command Modes

OpenFlow switch configuration (config-openflow-switch)

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

Usage Guidelines

A value of zero seconds mean that flow statistics collection is disabled.

Example

The following example shows how to configure the flow statistics collection interval:

```
Device# configure terminal
Device(config)# feature openflow
Device(config)# openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)# statistics collection-interval 9
```

Related Commands

Command	Description
feature openflow	Enables the OpenFlow feature.
openflow	Enables OpenFlow configuration and enters OpenFlow configuration mode.
switch(OpenFlow)	Configures a logical switch and enters OpenFlow switch configuration mode.

stream

To configure telemetry stream for the subscription, use the **stream** command in telemetry-subscription configuration mode. To disable the configuration, use the **no** form of this command.

stream yang-push
no stream yang-push

Syntax Description

yang-push	Configures a yang-push stream.

Command Modes

Telemetry-subscription configuration (config)

Command History

Release	Modification
Cisco IOS XE Gibraltor 16.10.1	This command was introduced.

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-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 telemetry stream for the subscription:

```
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# stream yang-push
```

switch (OpenFlow)

To configure a logical switch and enter OpenFlow switch configuration mode, use the **switch** command in OpenFlow configuration mode. To disable the logical switch configuration, use the **no** form of this command.

switch 1 pipeline 1
no switch 1 pipeline 1

Syntax Description

1	Configures the OpenFlow logical switch ID.
pipeline 1	Configures the OpenFlow pipeline ID.

Command Default

The OpenFlow logical switch is not configured.

Command Modes

OpenFlow configuration (config-openflow)

Command History

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

Example

The following example shows how to configure a logical switch and pipeline:

```
Device# configure terminal
Device(config)# feature openflow
Device(config)# openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)#
```

Related Commands

Command	Description
feature openflow	Enables the OpenFlow feature.
openflow	Enables OpenFlow configuration and enters OpenFlow configuration mode.

switch pipeline

To enable the OpenFlow logical switch and pipeline configuration, use the **switch pipeline** command in OpenFlow configuration mode. To disable the logical switch and pipeline configuration, use the **no** form of this command.

switch *switch_ID* **pipeline** *pipeline_ID*
no switch *switch_ID* **pipeline** *pipeline_ID*

Syntax Description

switch_ID	OpenFlow switch ID.
pipeline_ID	OpenFlow pipleline ID

Command Default

The logical switch and pipeline are not enabled.

Command Modes

OpenFlow configuration (config-openflow)

Command History

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

Usage Guidelines

Can we explain what a logical switch and pipeline are?

When you configure the command, the configuration mode changes to OpenFlow switch configuration mode.

Example

The following example shows how to enable a logical switch and pipeline configuration:

```
Device# configure terminal
Device(config)# feature openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)#
```

Related Commands

Command	Description
feature openflow	OpenFlow Plugin configuration and enter OpenFlow configuration mode.

telemetry ietf subscription

To configure telemetry subscription, use the **telemetry ietf subscriptiom** command in global configuration mode. To disable the configuration, use the **no** form of this command.

telemetry ietf {**subscriptiom** *sub-id*}
no telemetry ietf {**subscriptiom** *sub-id*}

Syntax Description

subscriptiom *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 Gibraltor 16.10.1	This command was introduced.

Example

The following example shows how to configure an telemetry subscription:

```
Device(config)# telemetry ietf subscription 101
```

tls trustpoint

To configure the OpenFlow switch Transport Layer Security (TLS) trustpoint, use the **tls trustpoint** command in OpenFlow switch configuration mode. To remove the TLS trustpoint, use the **no** form of this command.

tls trustpoint local *name* **remote** *name*
no tls trustpoint local *name* **remote** *name*

Syntax Description

local *name*	Configures a local trustpoint.
remote *name*	Configures a remote trustpoint.

Command Default

TLS trustpoint is not configured.

Command Modes

OpenFlow switch configuration (config-openflow-switch)

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

Usage Guidelines

Example

The following example shows how to configure a TLS trustpoint for an OpenFlow switch:

```
Device# configure terminal
Device(config)# feature openflow
Device(config)# openflow
Device(config-openflow)# switch 1 pipeline 1
Device(config-openflow-switch)# tls trustpoint local local-tls remote remote-tls
```

Related Commands

Command	Description
feature openflow	Enables the OpenFlow feature.
openflow	Enables OpenFlow configuration and enters OpenFlow configuration mode.
switch(OpenFlow)	Configures a logical switch and enters OpenFlow switch configuration mode.

update-policy

To configure update policy for subscription, use the **update-policy** command in telemetry-subscription configuration mode. To disable the configuration, use the **no** form of this command.

update-policy{**on-change** | **periodic** *period*}
no update-policy{**on-change** | **periodic** *period*}

Syntax Description

on-change	Enables on-change updates.
periodic *period*	Enable on-change updates. Valid values are from 100 to 4294967295.

Command Modes

Telemetry-subscription configuration (config)

Command History

Release	Modification
Cisco IOS XE Gibraltor 16.10.1	This command was introduced.

Usage Guidelines

Example

The following example shows how to configure a periodic update policy for a subscription:

```
Device(config)# telemetry ietf subscription 101
Device(config-mdt-subs)# update-policy periodic 6000
```

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

What is the default vCPU?

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
```

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 VLAN-access configuration (config-config-app-hosting-vlan-access)

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 AppGigEthernet vlan-access
Device(config-config-app-hosting-vlan-access)# 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 AppGigEthernet vlan-access	Configures the front-panel VLAN port for application hosting and enters application-hosting VLAN-access 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 Fuji 16.8.1	This command was introduced.

Example

The following example shows how to configure a vNIC gateway:

```
Device# configure terminal
Device(config)# app-hosting appid lxc_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 Fuji 16.8.1	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 lxc_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.