OpenFlow

- Finding Feature Information, on page 1
- Prerequisites for OpenFlow, on page 1
- Restrictions for OpenFlow, on page 2
- Information About Open Flow, on page 3
- Configuring OpenFlow, on page 9
- Monitoring OpenFlow, on page 13
- Configuration Examples for OpenFlow, on page 13

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for OpenFlow

The Prerequisites for OpenFlow are as follows:

- A Cisco device and its corresponding operating system that supports the installation of OpenFlow.

  Refer to the corresponding release notes for information about which operating system release supports the features and necessary infrastructure.

  Note Note: Release notes for Cisco Catalyst 2960X/XR Series Switches

- A controller installed on a connected server.

Table 1: Controller Support

OpenFlow Version	Supported Controllers
OpenFlow 1.0	Extensible Network Controller (XNC) 1.0, POX, Cisco Open SDN Controller, or Ixia controllers
OpenFlow 1.3	Ixia, CiscoOpen SDN Controller, or OpenDaylight

Restrictions for OpenFlow

The Restrictions for OpenFlow are as listed below:

- OpenFlow supports only a subset of OpenFlow 1.3 functions. For more information, see the Cisco OpenFlow Feature Support section.
- You cannot configure more than one OpenFlow logical switch. The logical switch ID has a value of 1.
- OpenFlow hybrid model (ships-in-the-night) is supported. VLANs configured for OpenFlow logical switch ports should not overlap with regular device interfaces.
- The OpenFlow logical switch ports must not be configured in a mode other than trunk port.
- You cannot configure a bridge domain, Virtual LANs, virtual routing and forwarding (VRF) or port-channel interfaces on an OpenFlow logical switch. You can only configure physical interfaces.
- You cannot make additional configurations to an interface configured as a port of OpenFlow Logical Switch without removing the configuration as a port of OpenFlow Logical Switch.
- In stack scenarios, consisting of master / member switches, whenever the master switch goes down, all current configuration will exist in newly elected master switch. However, the flows have to program again from the controller.
- MIBs and XMLs are not supported.
- Cisco Catalyst 2960X/XR switch supports 1000 L2 flows with EtherType, 200 L2 flows without EtherType, and 500 L3 flows.
- A maximum of 48 ports can be assigned for Openflow operation.
- In general, the maximum sustained flow programming rate from the controller should not exceed 50 (added or deleted) flows per second. For flows that have more than 1 match criteria (more than input port + 1 match), the sustained controller programming rate should not exceed 40 flows per second.
- The maximum burst flow programming rate from the controller should not exceed 1000 flows, spaced by 30-second time intervals. A minimum of 30-second time interval should be maintained between addition or deletion of flows.
- The rate of PACKET_IN messages sent to the controller should be rate-limited to 300 packets per second, using configuration.

Information About Open Flow

Overview of OpenFlow

OpenFlow is a standard communications interface defined between the control and forwarding plane for direct access to and manipulation of the forwarding plane of network devices such as switches and routers from multiple vendors.

OpenFlow Switch Specification Version 1.0.1 (Wire Protocol 0x01), referred to as OpenFlow 1.0, and OpenFlow Switch Specification Version 1.3.0 (Wire Protocol 0x04), referred to as OpenFlow 1.3, are based on the concept of an Ethernet switch with an internal flow table and standardized interface to allow traffic flows on a device to be added or removed. OpenFlow 1.3 defines the communication channel between OpenFlow and controllers.

A generic OpenFlow controller will interact with an specialized OpenFlow agent that translates the OpenFlow configuration into IOS configurations and configures the data plane.

Support of OpenFlow on catalyst 2960X/XR is limited to only software forwarding (due to ASIC limitations). The software forwarding of flows will happen at the OpenFlow agent with support of 12 tuples matches consisting of single table with both L2 and L3 fields together. The match criteria can be match on all 12 tuple fields or any of the 12 tuple fields.

The corresponding actions to the matching criteria can be:

- Push / Pop of Vlan
- Output the packet to port
- Drop the packet
- Set/Decrement IP TTL value
- Modify of L2/L3/L4 fields of Ethernet frame

The Physical ports can be configured as OpenFlow ports or as normal port. The flows in the flow table will be installed based on the priority of the flow.

Note Priority 0 flows are not supported.

Cisco supports a subset of OpenFlow 1.0 and OpenFlow 1.3 functions. A controller can be Extensible Network Controller (XNC) 1.0, or any controller compliant with OpenFlow 1.3.

OpenFlow Controller Operation

OpenFlow controller (referred to as controller) controls the switch and inserts flows with a subset of OpenFlow 1.3 and 1.0 match and action criteria through OpenFlow logical switch.

Cisco OpenFlow Feature Support

The following is a subset of OpenFlow 1.3 and OpenFlow 1.0 functions that are supported by OpenFlow.

Table 2: Cisco OpenFlow Feature Support

Feature	Notes
Configuration of physical interfaces as OpenFlow logical switch ports	Bridge domain, Virtual LANs and Virtual Routing and Forwarding (VRF), and port-channel interfaces are not supported. Only L2 interfaces can be OpenFlow logical switch ports.
Supported OpenFlow message types	Controller to switch: • Handshake • Switch Configuration • Modify State (Port Modification message is not supported) • Read State • Packet-Out • Barrier Asynchronous messages: • Packet-In • Flow Removed • Port Status • Error Symmetric messages: • Hello • Echo Request • Echo Reply • Vendor
Connection to controllers	You can connect up to eight controllers. Connection to the controller through a management interface or a switched virtual interface (SVI) is supported. Connection via TCP and TLS is supported.

Feature	Notes
Multiple actions	If multiple actions are associated with a flow, they are processed in the order specified. The output action should be the last action in the action list. Any action after the output action is not supported, and can cause the flow to fail and return an error to the controller. Flows defined on the controller must follow the these guidelines: • The flow can have only one output action. • Some action combinations which are not supported may be rejected at flow programming time. • The flow should not have an output–to–controller action in combination with other rewrite actions.
Supported OpenFlow counters	Per Table—Active entries, packet lookups, and packet matches. Per Flow—Received Packets, Received bytes, Duration (seconds), Duration (milliseconds). Per Port—Received or transmitted packets, and bytes. Per Controller— Flow addition, modification, deletion, error messages, echo requests or replies, barrier requests or replies, connection attempts, successful connections, packet in or packet out.
Default forwarding rule	All packets that cannot be matched to programmed flows are dropped by default. You can configure sending unmatched packets to the controller. You can modify the default action taken on unmatched packets either using the default-miss command or by the controller.
Idle timeout	A minimum Idle timeout of 14 seconds is supported for 700 flows and 48 ports. The statistics collection interval influences the minimum idle timeout. When the interval is set to 7 seconds, the timeout is a minimum of 14 seconds. 700 flows are supported with the 14-second idle timeout. When using an idle timeout of less than 25 seconds, the number of L3 flows should be limited to 700.

Supported Match and Actions and Pipelines

Table 3: Supported Match and Actions and Pipelines

Feature	Notes
Pipelines	Pipelines are mandatory for logical switch. The logical switch supports only pipeline 1. The logical switch supports only table 0.

Feature	Notes
Forwarding Table	

Feature	Notes
	Match Criteria: • Input Port • Ethernet type • Source Mac Address • Dest Mac Address • VLAN ID • IP TOS (DSCP bits) • IP Protocol (except for lower 8 bits of ARP code) • IPv4 Source Address • IPv4 Destination Address • Layer 4 Source Port • Layer 4 Destination Port • IPv6 Source Address • IPv6 Destination Address Action Criteria: • Forward: Controller • Forward: Port • Forward: Drop • Forward: Controller + Port • Set VLAN ID • New VLAN ID • Replace VLAN ID • Strip VLAN Header • Modify Source MAC • Modify Destination MAC • Modify IPv4 Source Address • Modify IPv4 Destination Address • Modify IPv4 TOS bits • Modify L4 source port • Modify L4 destination port • Decrement TTL

Feature	Notes
Number of flows	1000
Configuration of VLANs	VLAN range is from 1 to 4094.

Configuring OpenFlow

To configure OpenFlow logical switch and the IP address of a controller, perform this task:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**feature openflow** **Example:** `Device(config)# feature openflow`	Enables Open Flow Agent support on the switch.
Step 4	**openflow** **Example:** `Device(config)# openflow`	Enables Open Flow Agent support on the switch.
Step 5	**switch** *logical-switch-id* **pipeline** *logical-id* **Example:** `Device(config-ofa-switch)# switch 1 pipeline 1`	Specifies an ID for a logical switch that is used for OpenFlow switching and enters logical switch configuration mode. The only logical switch ID supported is 1. Configures a pipeline. This step is mandatory for a logical switch configuration. The only pipeline ID supported is 1.

	Command or Action	Purpose
Step 6	**controller** [**ipv4** *ip-address*] [**port** *tcp-port*] [**vrf** *vrf-name*] [**security** {**none** | **tls**}] **Example:** `Device(config-ofa-switch)# controller ipv4 10.1.1.1 tcp 6633`	Specifies the IPv4 address, port number used by the controller to connect to the logical switch Repeat this step if you need to configure additional controllers. You can configure up to eight controllers. If TLS is used in this step, configure TLS trustpoints in the next step. If unspecified, by default, Controllers use TCP port 6633. A connection to a controller is initiated by the logical switch.

Step 7	**of-port interface** *interface-name*
	Example:
	`Device(config-ofa-switch)# of-port interface GigabitEthernet1/0/23`
	`Device(config-ofa-switch)# of-port interface TenGigabitEthernet1/1/2`
		Adds interfaces to the logical switch configuration.
		Observe these guidelines:
		• Do not abbreviate the interface type. Ensure that the interface type is spelled out completely and is as shown in the examples.
		• If the keyword is abbreviated, the interface is not configured.
		• The interface must be designated for the OpenFlow logical switch only.
		Repeat this step to configure additional interfaces.

Step 8	**default-miss** *action-for-unmatched-flows* **Example:** `Device(config-ofa-switch)# default-miss continue-controller`	Configures the action to be taken for packets that do not match any of the flow defined. The supported options are: • forward the packets using the normal routing tables • forward the packets to the controller • drop the packets The default option is to forward the packets using the normal routing tables.
Step 9	**protocol-version {1.1 | 1.3 | negotiate}** **Example:** `Device(config-ofa-switch)# protocol-version negotiate`	Configures the protocol version. Supported values are: • 1.0—Configures device to connect to 1.0 controllers only. • 1.3—Configures device to connect to 1.3 controllers only..

	Command or Action	Purpose
		• negotiate—Negotiates the protocol version with the controller. Device uses 1.3 for negotiation. The default value is 1.0.
Step 10	**shutdown** **Example:** `Device(config-ofa-switch)# shutdown`	Disables a logical switch, bringing down the tcp/ip connection and removing flows from the dataplane.
Step 11	**datapath-id** *datapath-id* **Example:** `Device(config-ofa-switch)# datapath-id 0x222`	Configures a unique datapath ID for the switch. This step is mandatory for a logical switch configuration. Enter a 64-bit hexadecimal value.
Step 12	**tls trust-point local** *local-trust-point* **remote** *remote-trust-point* **Example:** `Device(config-ofa-switch)# tls trust-point local myCA remote myCA`	(Optional) Specifies the local and remote TLS trustpoints to be used for the controller connection.
Step 13	**probe-interval** *probe-interval* **Example:** `Device(config-ofa-switch)# probe-interval 7`	(Optional) Configures the interval (in seconds) at which the controller is probed. After the configured interval of time passes, if the switch has not received any messages from the controller, the switch sends an echo request (echo_request) to the controller. It should normally receive an echo reply (echo_reply). If no message is seen for the duration of another probe interval, the switch presumes that the controller is down and disconnects the controller connection. The switch tries to reconnect periodically. The default value is 5 seconds; the range is from 5 to 65535 seconds.
Step 14	**rate-limit packet_in** *controllet-packet-rate* **burst** *maximum-packets-to-controller* **Example:** `Device(config-ofa-switch)# rate-limit packet_in 300 burst 50`	(Optional) Configures the maximum packet rate sent to the controller and the maximum packets burst sent to the controller in a second. The default value is zero, that is, an indefinite packet rate and packet burst is permitted. This rate limit is for OpenFlow. It is not related to the rate limit of the device (data plane) configured by COPP.

	Command or Action	Purpose
Step 15	**max-backoff** *backoff-timer* **Example:** `Device(config-ofa-switch)# max-backoff 8`	(Optional) Configures the duration (in seconds) for which the device must wait before attempting to initiate a connection with the controller. The device initially tries to initiate connection frequently, as the number of unsuccessful attempts increases, the device tries less frequently, that is, the waiting period between attempts also increases. The backoff timer configures the maximum period that the device waits in-between each retry. The default value is 8 seconds; the range is from 1 to 65535 seconds.
Step 16	**logging flow-mod** **Example:** `Device(config-ofa-switch)# logging flow-mod`	(Optional) Enables logging of flow changes, including addition, deletion, and modification of flows. Logging of flow changes is a CPU intensive activity and should not be enabled for a large number of flows. Logging of flow changes is disabled by default. Flow changes are logged in syslog and can be viewed using the **show logging** command.
Step 17	**statistics collection-interval** *interval* **Example:** `Device(config-ofa-switch)# statistics collection-interval 7`	Configures the statistics collection interval (in seconds) for all configured flows of OpenFlow. Observe these guidelines: • The default interval value is 7 seconds. • The minimum interval is 7 seconds; the maximum is 82 seconds. • You can also specify a value of 0, this disables statistics collection. • Flows with an idle timeout value less than 2 * interval are rejected. Configured interval value is displayed in the output of the **show openflow switch 1** command.
Step 18	**end** **Example:** `Device(config-ofa-switch)# end`	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

	Command or Action	Purpose
Step 19	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring OpenFlow

You can monitor OpenFlow parameters using the following commands:

Commands	Description
show openflow switch *switch-id*	Displays information related to OpenFlow on the logical switch.
show openflow switch *switch-id* **controllers [stats]**	Displays information related to the connection status between an OpenFlow logical switch and connected Controllers.
show openflow switch *switch-id* **ports**	Displays the mapping between physical device interfaces and ports of OpenFlow logical switch.
show openflow *switch-id* **flows**	Displays flows defined for the device by controllers.
show openflow switch *switch-id* **stats**	Displays send and receive statistics for each port defined for an OpenFlow logical switch.
show running-config | section openflow	Displays configurations made for OpenFlow.
show openflow hardware capabilities	Displays OpenFlow hardware configurations.

Configuration Examples for OpenFlow

This example shows how you can view information related to OpenFlow on the logical switch.

```
Device#show openflow switch 1

Logical Switch Context
Id: 1
Switch type: Forwarding
Pipeline id: 1
Data plane: secure
Table-Miss default: drop
Configured protocol version: Negotiate
Config state: no-shutdown
Working state: enabled
Rate limit (packet per second): 0
```

```
Burst limit: 0
Max backoff (sec): 8
Probe interval (sec): 5
TLS local trustpoint name: not configured
TLS remote trustpoint name: not configured
Logging flow changes: Disabled
Stats collect interval (sec): 7
Stats collect Max flows: 1000
Stats collect period (sec): 1
Minimum flow idle timeout (sec): 14
OFA Description:
 Manufacturer: Cisco Systems, Inc.
 Hardware: WS-C2960X-48LPS-L
 Software: Cisco IOS Software, C2960X Software (C2960X-UNIVERSALK9-M),
Version 15.2(5.1.50)E, TEST ENGINEERING ESTG_WEEKLY BUILD, synced to
V152_4_1_20_E1| openvswitch 2.1
 Serial Num: FCW1910B5QR
 DP Description: 2960xr:sw1
OF Features:
 DPID: 0x0000000000000251
 Number of tables: 1
 Number of buffers: 256
 Capabilities: FLOW_STATS TABLE_STATS PORT_STATS
Controllers:
 10.106.253.118:6653, Protocol: TCP, VRF: default
Interfaces:
 GigabitEthernet1/0/1
 GigabitEthernet1/0/2

-------------------------------------------------------------------------------------------
```

This example shows how you can view information related to the connection status between an OpenFlow logical switch and connected Controllers.

```
Device#show openflow switch 1 controllers

Logical Switch Id: 1
Total Controllers: 1
 Controller: 1
  10.106.253.118:6653
  Protocol: tcp
  VRF: default
  Connected: Yes
  Role: Equal
  Negotiated Protocol Version: OpenFlow 1.3
  Last Alive Ping: 2016-04-03 18:40:48 UTC
  state: ACTIVE
  sec_since_connect: 192038

Device#show openflow switch 1 controllers stats

Logical Switch Id: 1
Total Controllers: 1
```

```
Controller: 1
address : tcp:10.106.253.118:6653
connection attempts : 9
successful connection attempts : 1
flow adds : 1
flow mods : 0
flow deletes : 0
flow removals : 0
flow errors : 0
flow unencodable errors : 0
total errors : 0
echo requests : rx: 0, tx:0
echo reply : rx: 0, tx:0
flow stats : rx: 64004, tx:64004
barrier : rx: 0, tx:0
packet-in/packet-out : rx: 0, tx:0
```

This example shows how you can view the mapping between physical device interfaces and ports of OpenFlow logical switch.

```
Device#show openflow switch 1 ports

Logical Switch Id: 1
Port Interface Name Config-State Link-State Features
1    Gi1/0/1        PORT_UP      LINK_UP    1GB-FD
2    Gi1/0/2        PORT_UP      LINK_UP    1GB-FD
```

This example shows how you can view flows defined for the device by controllers.

```
Device#show openflow switch 1 flows

Logical Switch Id: 1
Total flows: 2

Flow: 1
 Match:
  Actions: drop
  Priority: 0
  Table: 0
  Cookie: 0x0
  Duration: 4335.022s
  Number of packets: 18323
  Number of bytes: 1172672

Flow: 2
 Match: ipv6
  Actions: output:2
  Priority: 1
  Table: 0
  Cookie: 0x0
```

```
  Duration: 727.757s
  Number of packets: 1024
  Number of bytes: 131072
```

This example shows how you can view the send and receive statistics for each port defined for an OpenFlow logical switch.

```
Device#show openflow switch 1 stats

Logical Switch Id: 1
Total ports: 2
 Port 1: rx
    tx
 Port 2: rx
    tx
Total tables: 1
 Table 0: Main
 Wildcards = 0x00000
 Max entries = 1000
 Active entries = 2
 Number of lookups = 0
 Number of matches = 0
```

This example shows how you can view configurations made for OpenFlow.

```
Device#show running-config | section openflow

feature openflow
 mode openflow
 mode openflow
openflow
 switch 1 pipeline 1
 controller ipv4 10.106.253.118 port 6653 security none
 of-port interface GigabitEthernet1/0/1
 of-port interface GigabitEthernet1/0/2
 datapath-id 0x251
```

This example shows how you can view OpenFlow hardware configurations.

```
Device#show openflow hardware capabilities

Max Flow Batch Size: 100
Statistics Max Polling Rate (flows/sec): 1024
Max Interfaces: 1000
Aggregated Statistics: YES
Pipeline ID: 1
Pipeline Max Flows: 1000
Pipeline Default Statistics Collect Interval: 7
Flow table ID: 0
```

```
Max Flow Batch Size: 100
Max Flows: 1000
Bind Subintfs: FALSE
Primary Table: TRUE
Table Programmable: TRUE
Miss Programmable: TRUE
Number of goto tables: 0
Goto table id:
Stats collection time for full table (sec): 1
Match Capabilities              Match Types
------------------              -----------
ethernet mac destination        optional
ethernet mac source             optional
ethernet type                   optional
VLAN ID                         optional
IP DSCP                         optional
IP protocol                     optional
IPv4 source address             lengthmask
IPv4 destination address        lengthmask
ipv6 source addresss            lengthmask
ipv6 destination address        lengthmask
source port                     optional
destination port                optional
in port (virtual or physical)   optional

Actions                            Count Limit Order
---------------------------        ----------- -----
set eth source mac                     1         10
set eth destination mac                1         10
set vlan id                            1         10
set IPv4 source address                1         10
set IPv4 destination address           1         10
set IP dscp                            1         10
set TCP source port                    1         10
set TCP destination port               1         10
set UDP source port                    1         10
set UDP destination port               1         10
pop vlan tag                           1         10
set qos group                          1         10
drop packet                            1         100
specified interface                    1         100
controller                             1         100
divert a copy of pkt to application    1         100

Miss actions                       Count Limit Order
---------------------------        ----------- -----
drop packet                            1         100
controller                             1         100
```