

## DirectFlow

---

This chapter describes Arista's DirectFlow implementation. Sections in this chapter include:

- [Section 46.1: Introduction](#)
- [Section 46.2: DirectFlow Configuration](#)
- [Section 46.3: DirectFlow Feature Interactions](#)
- [Section 46.4: DirectFlow Command Descriptions](#)

### 46.1 Introduction

Like OpenFlow, DirectFlow exposes the underlying forwarding ASICs capabilities through a programmable interface like EAPI or the standard CLI.

Unlike OpenFlow, DirectFlow works in conjunction with all other aspects of standard L2/L3 bridging or forwarding, and DirectFlow traffic is subject to the standard packet processing pipeline within the ASIC. You can think of DirectFlow as a stage in packet processing that processes traffic after ingress checks and before any egress actions.

This feature enables you to configure flows that consist of a matching criteria and actions, and to modify how traffic is processed (for example, by overriding the L2 lookup decision or rewriting a mac address or VLAN).

Features like MAC learning, STP state checks, ingress or egress VLAN membership checks on ports, ACLs, QoS and other features are all respected by DirectFlow. Traffic that doesn't match any programmed flow is processed normally while traffic that matches programmed flows is now subject to the actions specified in the flows.

DirectFlow and OpenFlow are mutually exclusive and you can run only one of the two at any given time.

#### **How DirectFlow is different from OpenFlow**

There is no default flow matching all traffic, traffic not matched by other rules is forwarded as normal. This means the configuration/ controller/ application doesn't consume TCAM space programming flows for normal forwarding.

DirectFlow works with other features and so the user can use ACL, rate limiting, STP etc. in their network as normal and not build all of that into the application used to inject flows.

DirectFlow flows can be configured from the CLI or using EAPI, giving users the option of using flow based forwarding without an external controller. This is especially useful where the number of flows is small and static e.g. to process a small subset of the traffic in a different manner to the normal L2/L3 pipeline.

Unlike OpenFlow which requires the switch support OUTPUT NORMAL or re-circulate a packet in order to send a packet from the OpenFlow domain to non-OpenFlow domain, there is just one domain with DirectFlow.

## 46.1.1 DirectFlow Flows

Similar to OpenFlow, you can define a relative priority between flows and define idle or hard timeouts for the flow. DirectFlow also enables you to insert a flow entry that matches on specified criteria, and define actions to be taken on traffic that matches the specified matching conditions. You can define flows to match on TCP flags, IPv6 source and destination addresses, input ports, and more.

For more information, see:

- [Section 46.1.1.1: DirectFlow Non-persistent Flows](#)
- [Section 46.1.1.2: Supported matches](#)
- [Section 46.1.1.3: Supported actions](#)

### 46.1.1.1 DirectFlow Non-persistent Flows

DirectFlow enables you to configure flows that are not visible in the startup or running configurations. This feature is designed to be used for flows that are configured by a custom agent using the EOS SDK or eAPI and age out (expire) after a specified time period.

For example, if you are using a custom agent that reacts to traffic sent to the CPU (the redirect to CPU action), and you want to use a flow that will drop all matching traffic for 5 minutes, the agent can program a non-persistent flow that expires after a hard timeout of 300 seconds.

Using a non-persistent flow for this purpose ensures that other administrator actions (for example, saving the configuration) does not result in the flow being resurrected on startup or reverting to the saved configuration. It also removes the need for the agent to delete the expired flow.

---

#### Note

By default, all direct flow flows are persistent. You must use the **no persistent** command to configure a non-persistent flow.

---

### 46.1.1.2 Supported matches

DirectFlow supports all matches supported on EOS with OpenFlow 1.0.

This includes matches on VLAN, ether type, source or destination MAC address, COS, source or destination IP address, IP protocol, IP TOS, L4 source, destination ports, ICMP type and code).

In addition, DirectFlow also allows matching on:

- TCP flags
- IPv6 source address
- IPv6 destination address
- Traffic injected from the CPU
- Input port

DirectFlow also permits re-using the same flow on multiple input ports, saving valuable TCAM space.

### 46.1.1.3 Supported actions

DirectFlow supports all actions supported on EOS with OpenFlow 1.0, including:

- Setting the source or destination MAC address

- VLAN
- COS
- IP TOS
- Transmit queue
- Output port list and mirroring traffic pre-modification (ingress mirror) and post-modification (egress mirror).
- Redirect to CPU

The redirect to CPU action is useful in cases in which a custom agent is running on EOS and you want to trap specific traffic (matching traffic) and send the trapped traffic to the agent. See the example [“Redirect to CPU” on page 2595](#).

## 46.2 DirectFlow Configuration

Consider the following when using DirectFlow.

- DirectFlow takes effect ONLY after exiting the individual flow configuration sub-mode.
- Match criteria are connected with Boolean AND operators i.e. they must **all** match for the condition to be true and action to be taken.
- CLI is automatically set to match the ethertype to IP if IP fields (such as source or destination address or L4 ports) are chosen as part of other match/ action commands.
- In a single flow, only the following fields can be matched along with IPv6 source and destination addresses:
  - VLAN priority
  - VLAN ID
  - EtherType
  - Source interface
  - Class of Service (CoS)

### 46.2.1 Commands Used to Enable DirectFlow, Configure and Display Flows

A number of different commands are provided for the DirectFlow feature. The different commands enable you to enter the DirectFlow configuration mode, enable DirectFlow, configure flows, and display configured flows.

---

**Important!** ALL match criteria specified in a flow definition must match in the packet for the actions specified to be applied to the traffic.

---

#### Enter the DirectFlow configuration mode

The **directflow** command places the switch in DirectFlow configuration mode.

```
switch(config)#directflow
switch(config-directflow)#
```

#### Enable DirectFlow

The **shutdown (DirectFlow)** command determines if the configuration takes effect or not. To enable DirectFlow, enter the following command.

```
switch(config-directflow)#no shutdown
switch(config-directflow)#
```

#### Create the flow

The **flow (DirectFlow)** command creates a new flow entry. It must be unique or it will be overwritten by an existing entry.

```
switch(config-directflow)#flow Test-1
switch(config-directflow-Test-1)#
```

### Create the DirectFlow match criteria

The **match (DirectFlow-flow mode)** command allows you to configure a rule or a flow which match on L2, L3, L4 fields of a packet and specify a certain action to either modify, drop or redirect the packet.

```
switch(config-directflow)#flow Test1
switch(config-directflow-Test1)#match ethertype ip
switch(config-directflow-Test1)#match source ip 10.10.10.10
```

### Action Set

The **action set (DirectFlow-flow mode)** command allows you to configure a packet to be routed out a layer three interface using a DirectFlow entry.

```
switch(config-directflow)#flow Test1
switch(config-directflow-Test1)#action egress mirror ethernet 7
switch(config-directflow-Test1)#action set destination mac 0000.aaaa.bbbb
```

### Redirect to CPU

The **action output interface cpu (DirectFlow-flow mode)** command allows you to configure flows so that traffic that matches the matching conditions specified in the flow is redirected to the CPU.

```
switch(config)#directflow
switch(config-directflow)#flow redirect-http-cpu
switch(config-directflow-redirect-http-cpu)#match ip protocol tcp
switch(config-directflow-redirect-http-cpu)#match destination port 80
switch(config-directflow-redirect-http-cpu)#action output interface cpu
```

### Configuring a non-persistent flow

Including the **no persistent** command allows you to configure non-persistent direct flow flows.

```
switch (config-directflow)#flow example-non-persistent
switch (config-directflow-example-non-persistent)#match input interface ethernet 25
switch (config-directflow-example-non-persistent)#action drop
switch (config-directflow-example-non-persistent)#no persistent
switch (config-directflow-example-non-persistent)#timeout hard 300
```

### Display details for configured flows

The **show directflow flow <flow name> detail** command enables you to display the details of configured flows. You can use this command to verify that a non-persistent flow is deleted after the timeout period configured for the flow has elapsed.

The following example shows the use of this command to view the configuration of a non-persistent flow before the timeout period has elapsed, and a second time, after the timeout period has expired.

The initial use of the command displays the flow configuration (before the timeout expires).

```
switch (config-directflow)#show directflow flows example-non-persistent detail
Flow example-non-persistent: (Flow programmed)
  persistent: False
  priority: 0
  hard timeout: 300
  idle timeout: 0
  match:
    ingress interface:
      Et25
  actions:
    drop
  matched: 0 packets, 0 bytes
```

The **second** use of the command displays the flow details (after the timeout expires). The output shows that the flow is no longer programmed.

```
switch (config-directflow)#show directflow flows example-non-persistent detail
Flow example-non-persistent: (Flow not programmed)
persistent: False
priority: 0
hard timeout: 300
idle timeout: 0
match:
  ingress interface:
    Et25
actions:
  drop
matched: 0 packets, 0 bytes
```

## 46.3 DirectFlow Feature Interactions

DirectFlow flow entries can have one of the following actions:

- A set of egress ports for sending a matched packet
- Copy to CPU
- Redirect to CPU
- Drop (default)
- No specified action (in this case, the traffic is output normally).

The only exception is the ingress or egress mirroring action, where the DirectFlow entry causes the packet to be mirrored.

When the ingress or egress packets are mirrored, the original traffic is sent out normally.

### Bridging Features

- DirectFlow entries have precedence over all entries in the MAC table, including static MAC entries and static MAC drop entries. Packets that do not match DirectFlow entries are forwarded based on the MAC address table.
- VLANs: DirectFlow entries can modify the VLAN of a packet. MAC learning takes place in the original VLAN for DirectFlow entries that modify the VLAN. The modified packet will be subject to VLAN membership checks on the egress port. If a packet has no VLAN tag, DirectFlow assumes it came in on the native VLAN for the ingress interface. A VLAN override causes the packet to obey the VLAN rules on the egress port.
- Q-in-Q: Q-in-Q is supported as DirectFlow entries match only on the outer tag.
- Counters: All packets that match DirectFlow entries cause interface counters to increment as usual.

### Spanning Tree

DirectFlow runs alongside MSTP, RSTP, and PVST. DirectFlow entries do not match on packets that ingress an STP discarding port. DirectFlow entries that cause a packet to be forwarded out an STP discarding port will result in the packets being dropped on egress.

When STP is enabled, BPDUs will always be trapped to the CPU. When STP is disabled, BPDUs will be subject to DirectFlow entries and not be copied to the CPU by default.

### LLDP, LAGs, and LACP

- LLDP packets are always trapped to the CPU. DirectFlow entries can never match LLDP packets.
- LAGs are fully supported, and can be part of a match criteria and part of an output action to an interface.
- LACP packets are always trapped to CPU. DirectFlow entries can never match LACP packets.

### sFlow

sFlow is unaffected by DirectFlow.

### IGMP Snooping

IGMP control packets are trapped to the CPU when IGMP Snooping is enabled. DirectFlow entries can match IGMP Snooping control traffic and override the trap to CPU.

Link-local-multicast packets are flooded in hardware in the VLAN via a TCAM entry. DirectFlow entries can match link-local-multicast packets and change the flooding behavior. As DirectFlow entries have to specify output interfaces or drop, the action will conflict and so matching DirectFlow entries will get precedence.

When IGMP snooping is enabled, unknown IPV4 multicast packets are flooded to the multicast-router ports in the VLAN. If DirectFlow entries match unknown IPV4 multicast packets, they will override the flooding behavior.

Data packets in groups under IGMP snooping control are sent to the group members through a MAC table entry. Matching DirectFlow entries override the MAC table entries.

### ACLs

DirectFlow entries are lower priority than any configured Port ACLs (ingress). Packets coming in on a port that match DirectFlow entries obey any configured ACL on that port, and will only apply to packets that have a **permit** action.

DirectFlow entries are higher priority than any configured RACLs. Packets coming in on an L3 interface that match DirectFlow entries ignore any RACLs configured on that interface.

DirectFlow entries are lower priority than any configured Egress ACLs.

## 46.3.1 Layer Three Features and DirectFlow

DirectFlow runs alongside IP routing. If a packet is routed out a layer three interface using a DirectFlow entry, the actions associated with the entry will have to specify the new source MAC and destination MAC for the packet, as well as the physical port or LAG. If there are no output ports specified in an entry, packets that match that entry will be dropped.

### Unicast Routing

When unicast routing is enabled, DirectFlow entries that match take precedence for all packets that would have been otherwise been routed. The three exceptions are the ingress mirror, egress mirror and copy-to-CPU actions where the packets will be routed normally in addition to the action being performed. Routed packets that do not match DirectFlow entries are forwarding based on the L3 lookup.

### Multicast Routing

When multicast routing is enabled, DirectFlow entries that match take precedence for all packets that would have otherwise been multicast routed. The packets are not replicated based on the hardware multicast tables, but are forwarded strictly according to the actions specified by the DirectFlow entry. The entry can specify a set of output interfaces, which will result in the packet being replicated based on the DirectFlow entry.



### 46.3.2 Displaying DirectFlow Configurations

To **show directflow flows** command displays the contents of the flow table, showing each entry with its match rules, actions, and packet counters.

- This example shows the status of a default (persistent) flow.

```
switch(config-directflow)#show directflow flows
Flow Test1:
  priority: 0
  match:
    ingress interface: Ethernet1
    ethertype ip
    source ip address: 10.10.10.10
  actions:
    output mirror: Ethernet2
  matched: 0 packets, 0 bytes
switch(config-directflow)#
```

- This example shows the status of a non-persistent flow. The flow will be deleted once 5 minutes have elapsed.

```
switch(config-directflow)#show directflow flows example-non-persistent
Flow example-non-persistent:
  persistent: False
  priority: 0
  hard timeout: 300
  idle timeout: 0
  match:
    ingress interface:
      Et25
  actions:
    drop
  matched: 0 packets, 0 bytes
```

## 46.4 DirectFlow Command Descriptions

### DirectFlow Global Configuration Mode

- **directflow**

### DirectFlow Configuration Command

- **action drop (DirectFlow-flow mode)**
- **action mirror (DirectFlow-flow mode)**
- **action output (DirectFlow-flow mode)**
- **action output interface cpu (DirectFlow-flow mode)**
- **action set (DirectFlow-flow mode)**
- **flow (DirectFlow)**
- **match (DirectFlow-flow mode)**
- **priority (DirectFlow-flow mode)**
- **shutdown (DirectFlow)**
- **timeout (DirectFlow-flow mode)**

### DirectFlow and Clear Commands

- **show directflow**
- **show directflow flows**

## action drop (DirectFlow-flow mode)

The **action drop** command configures packets that match an entry to be dropped.

The **no action drop** and **default action drop** commands remove the statement from the DirectFlow configuration mode.

### Command Mode

Directflow-flow Configuration

### Command Syntax

```
action drop
no action drop
default action drop
```

### Example

- This command sets the action for packets from Test-1 to be dropped.

```
switch(config-directflow-Test-1)#action drop
switch#
```

## action mirror (DirectFlow-flow mode)

The **action mirror** command can be used to ingress or egress mirror traffic to a mirror destination. This requires a mirror destination to be setup on the switch. If a packet comes in or goes out an interface that is part of another mirror session, then the destination for that destination as well as the DirectFlow destination will receive a copy of the packet.

The **no action mirror** and **default action mirror** commands remove the statement from DirectFlow configuration mode.

### Command Mode

Directflow-flow Configuration

### Command Syntax

```
action DIRECTION mirror INT_NAME
no action DIRECTION mirror INT_NAME
default action DIRECTION mirror INT_NAME
```

### Parameters

- ***DIRECTION*** transmission direction of traffic to be mirrored.
  - **ingress** mirrors before any rewrites.
  - **egress** mirrors after rewrites.
- ***INT\_NAME*** Source interface for the mirroring session.
  - **ethernet *e\_range*** Ethernet interfaces specified by *e\_range*.
  - **port-channel *p\_range*** Port channel interfaces specified by *p\_range*.

### Example

- This command configures mirror traffic to Ethernet 2.

```
switch(config-directflow)# flow Test1
switch(config-directflow-Test1)#match ethertype ip
switch(config-directflow-Test1)#match source ip 10.10.10.10
switch(config-directflow-Test1)#action egress mirror ethernet 2
switch(config-directflow-Test1)#
```

## action output (DirectFlow-flow mode)

The **action output** command configures an Ethernet or port channel interface as the output of a specified port mirroring session.

The **no action output** and **default action output** commands remove the statement from DirectFlow configuration mode.

### Command Mode

Directflow-flow Configuration

### Command Syntax

```
action output DESTINATION
no action output DESTINATION
default action output DESTINATION
```

### Parameters

- **DESTINATION** transmission direction of traffic to be mirrored.
  - **all** mirrors transmitted and received traffic.
  - **flood** mirrors received traffic only.
  - **interface ethernet *e\_range*** Ethernet interfaces specified by *e\_range*.
  - **interface port-channel *p\_range*** Port channel interfaces specified by *p\_range*.

### Example

- This command configures Ethernet interface 7 as the output for the mirroring session.

```
switch(config-directflow-Test1)#action output interface ethernet 7
switch(config-directflow-Test1)#
```

## action output interface cpu (DirectFlow-flow mode)

The **action output interface cpu** command configures the action (other commands are used to define the traffic matching conditions).

The **no action output interface cpu** and **default action output** commands remove the statement from DirectFlow configuration mode.

### Command Mode

Directflow-flow Configuration

### Command Syntax

```
action output DESTINATION
no action output DESTINATION
default action output DESTINATION
```

### Parameters

- **DESTINATION** transmission direction of traffic to be mirrored.
  - **all** mirrors transmitted and received traffic.
  - **flood** mirrors received traffic only.
  - **interface cpu** Ethernet interfaces specified by *e\_range*.

### Example

- This command configures Ethernet interface 7 as the output for the mirroring session.

```
switch(config-directflow-Test1)#action output interface ethernet 7
switch(config-directflow-Test1)#
```

- These commands configure the action to redirect traffic matching the flow to the CPU and the matching conditions for the flow.

```
switch (config)#directflow
switch (config-directflow)#flow redirect-http-cpu
switch (config-directflow-redirect-http-cpu)#match ip protocol tcp
switch (config-directflow-redirect-http-cpu)#match destination port 80
switch (config-directflow-redirect-http-cpu)#action output interface cpu
```

## action set (DirectFlow-flow mode)

The **action set** command allows you to configure a packet to be routed out a layer three interface using a DirectFlow entry. The actions associated with the entry will have to specify the new source MAC and destination MAC for the packet, as well as the physical port or LAG. If there are no output ports specified in an entry, packets that match that entry will be dropped.

The **no action set** and **default action set** commands remove **action set** statement from DirectFlow configuration mode.

### Command Mode

Directflow-flow Configuration

### Command Syntax

```
action set CONDITION
no action set CONDITION
default action set CONDITION
```

### Parameters

- **CONDITION** specifies parameter and value. Options include:
  - **cos** <0 to 7> cost of service.
  - **destination mac** *mac\_addr* Dotted hex notation.
  - **ip tos** <0 to 255> Type of service.
  - **source mac** *mac\_addr* Dotted hex notation.
  - **traffic-class** <0 to 7> Dotted hex notation.
  - **vlan** <1 to 4094> Number of VLAN.

The **no action set** and **default action set** commands require only the **CONDITION** type without a specific condition value.

### Example

- These commands change the destination MAC of the frame.

```
switch(config-directflow)#flow Test1
switch(config-directflow-Test1)#action egress mirror ethernet 7
switch(config-directflow-Test1)#action set destination mac 0000.aaaa.bbbb
```

## directflow

The **directflow** command places the switch in DirectFlow configuration mode.

The **no directflow** and **default directflow** commands delete the DirectFlow configuration mode statements from *running-config*.

DirectFlow configuration mode is not a group change mode; *running-config* is changed immediately upon entering commands. Exiting OpenFlow configuration mode does not affect *running-config*. The **exit** command returns the switch to global configuration mode.

### Command Mode

Global Configuration

### Command Syntax

```
directflow
no directflow
default directflow
```

### Commands Available in DirectFlow-Flow configuration mode:

- **flow (DirectFlow)**
- **shutdown (DirectFlow)**

### Example

- This command places the switch in DirectFlow configuration mode:

```
switch(config)#directflow
switch(config-directflow)#
```

- This command returns the switch to global management mode:

```
switch(config-directflow)#exit
switch(config)#
```



## flow (DirectFlow)

The **flow** command places the switch in flow configuration mode.

The **flow** command specifies the name of the flow that subsequent commands modify and creates a newflow definition if it references a nonexistent flow. All changes in a flow configuration mode edit session are pending until the session ends:

- The **exit** command saves pending changes to **running-config** and returns the switch to DirectFlow configuration mode. Changes are also saved by entering a different configuration mode.
- The **abort** command discards pending changes, returning the switch to DirectFlow configuration mode.

The **no flow** and **default flow** commands delete the specified role by removing the role and its statements from **running-config**.

### Command Mode

DirectFlow Configuration

### Command Syntax

```
flow flow_name
no flow flow_name
default flow flow_name
```

### Parameters

- *flow\_name* Name of flow.

### Commands Available in DirectFlow-Flow configuration mode:

- **action drop (DirectFlow-flow mode)**
- **action mirror (DirectFlow-flow mode)**
- **action output (DirectFlow-flow mode)**
- **action set (DirectFlow-flow mode)**
- **match (DirectFlow-flow mode)**

## match (DirectFlow-flow mode)

The **match** command allows you to configure a rule or a flow which could match on L2, L3, L4 fields of a packet and specify a certain action to modify, drop or redirect the packet.

All traffic ingressing on the switch will be matched against the flows installed. In cases where none of the packets match, normal switching or routing behavior will take over. When multiple entries match a packet, precedence is given to the entry that was installed first.

The **no match** and **default match** commands remove the **match** statement from the configuration mode.

### Command Mode

Directflow-flow Configuration

### Command Syntax

```
match CONDITION
no match CONDITION
default match CONDITION
```

### Parameters

- **CONDITION** specifies criteria for evaluating a route. Options include:
  - **cos** *<0 to 7>* cost of service.
  - **destination ip** *ipv4\_sub* destination IPv4 subnet. L3 fields valid only if ethertype is IP (0x0800).
  - **destination mac** *mac\_addr* Add to the existing community. Dotted hex notation.
  - **destination mac** *mac\_addr mask mac\_mask* Add to the sting community. Dotted hex notation.
  - **destination port** *<0 to 65535>* Fields accepted only if protocol is TCPI UDP
  - **ethertype** *<0 to 65535>* Layer 4 destination port.
  - **ethertype ARP** Layer 4 destination port.
  - **ethertype IP** Layer 4 destination port.
  - **icmp code** *<0 to 255>* Fields accepted only if protocol is ICMP
  - **icmp type** *<0 to 255>* Fields accepted only if protocol is ICMP
  - **input interface ethernet** *e\_num* Ethernet interface specified by *e\_num*.
  - **input interface port-channel** *p\_num* Port channel interface specified by *p\_num*.
  - **ip protocol** *<0 to 255>* Type of service.
  - **ip protocol icmp** L3 fields valid only if ethertype is IP (0x0800).
  - **ip protocol tcp** L3 fields valid only if ethertype is IP (0x0800).
  - **ip protocol udp** L3 fields valid only if ethertype is IP (0x0800).
  - **ip tos** *<0 to 255>* L3 fields valid only if ethertype is IP (0x0800).
  - **source ip** *ipv4\_subnet* L3 fields valid only if ethertype is IP (0x0800).
  - **source mac** *mac\_addr* Add to the existing community. Dotted hex notation.
  - **source mac** *mac\_addr mask mac\_mask* Add to the sting community. Dotted hex notation.
  - **source port** *<0 to 65535>* Fields accepted only if protocol is TCPI UDP
  - **tcp flag ack** Layer 4 destination port.
  - **tcp flag fin** Layer 4 destination port.

- **tcp flag psh** Layer 4 destination port.
- **tcp flag rst** Layer 4 destination port.
- **tcp flag syn** Layer 4 destination port.
- **tcp flag urg** Layer 4 destination port.
- **tcp flag urg** Layer 4 destination port
- **vlan <1 to 4094> mask <1 to 4095>** Number of VLAN.

The **no match** and **default match** commands require only the **CONDITION** type without a specific condition value.

### Example

- This command creates the rules to match on Ethertype IP and Source IP 10.10.10.10.

```
switch(config-directflow)# flow Test1
switch(config-directflow-Test1)#persistent
switch(config-directflow-Test1)#match ethertype ip
switch(config-directflow-Test1)#match source ip 10.10.10.10
```

## priority (DirectFlow-flow mode)

The **priority** command sets the priority for the flow match rules. Each flow-table entry has an optional priority field, with a higher number indicating a higher priority. Flows with the same priority may be loaded in any order, and the order may be changed at any time. If multiple entries match a packet, precedence is given to the entry that was installed first.

Priority numbers range from 0 to 65535. The default is 0. The higher priority rules match first.

The **no priority** and **default priority** commands remove **priority** statement from the DirectFlow configuration mode.

### Command Mode

Directflow-flow Configuration

### Command Syntax

```
priority priority_value
no priority
default priority
```

### Parameters

- *priority\_level* priority xxx. Value ranges from 0 to 65535. Default is 0.

### Example

- These commands assign the priority of 150 to flow Test-1.

```
switch(config-directflow-Test-1)#priority 150
switch(config-directflow-Test-1)#
```

## show directflow

The **show directflow** command shows the effective DirectFlow configuration parameters.

### Command Mode

EXEC

### Command Syntax

```
show directflow
```

### Example

- This command displays the actual hardware state of DirectFlow.

```
switch# show directflow
DirectFlow configuration: Enabled
Total matched: 23 packets
switch#
```

## show directflow flows

The **show directflow flows** command displays the contents of the flow table, showing each entry with its match rules, actions, and packet counters.

### Command Mode

EXEC

### Command Syntax

```
show directflow flows
```

### Example

- This command displays the contents of the flow table.

```
switch# show directflow flows
Flow Test-1:
  priority: 0
  match:
    VLAN ID: 0xa/0x1
    Ethernet type: IPv4
    source IPv4 address: 10.10.10.1
  actions:
    set destination Ethernet address to: 00:00:aa:aa:bb:bb
    output interfaces: Port-Channel100
  matched: 0 packets, 0 bytes
switch#
```

## shutdown (DirectFlow)

The **shutdown** command, in DirectFlow mode, disables DirectFlow on the switch. DirectFlow is disabled by default.

The **no shutdown** command re-enables DirectFlow.

### Command Mode

Directflow Configuration

### Command Syntax

```
shutdown
no shutdown
default shutdown
```

### Example

- These commands enable DirectFlow on the switch.

```
switch(config)#directflow
switch(config-directflow)#no shutdown
switch(config-directflow)#
```

- This command disables DirectFlow Flow.

```
switch(config-directflow-Test1)#shutdown
```

## timeout (DirectFlow-flow mode)

The **timeout** command, in DirectFlow mode, command configures the connection timeout period for connection sessions. The connection timeout period defines the interval between a user's most recently entered command and an automatic connection shutdown. Automatic connection timeout is disabled by setting the idle-timeout to zero, which is the default setting.

### Command Mode

Directflow-flow Configuration

### Command Syntax

```
no priority
no timeout hard
no timeout idle
```

### Parameters

- *idle* session idle timeout length.
  - *0* Automatic connection timeout is disabled
  - *<1-4294967295>* Automatic timeout period (seconds).
- *hard* session hard timeout length.
  - *0* Automatic connection timeout is disabled.
  - *<1-4294967295>* Automatic timeout period (seconds).

### Example

- These commands enable a hard timeout period of 5 seconds on the switch.

```
switch(config)#directflow
switch(config-directflow-Test1)#timeout hard 5
switch(config-directflow-Test1)#
```

- These commands enable DirectFlow on the switch.

```
switch(config)#directflow
switch(config-directflow-Test1)#no timeout hard
switch(config-directflow-Test1)#
```