# VLANs

This chapter describes Arista's VLAN implementation and MAC address tables.

Sections in this chapter include:

## 20.1    VLAN Introduction

Arista switches support industry standard 802.1q VLANs. Arista EOS provides tools to manage and extend VLANs throughout the data center network.

## 20.2    VLAN Conceptual Overview

### 20.2.1    VLAN Definition

A virtual local area network (VLAN) allows a group of devices to communicate as if they were in the same network regardless of their physical location. VLANs are layer 2 structures based on the 802.1Q standard.

These parameters are associated with a VLAN:

- VLAN number (1-4094): VLAN numbers uniquely identify the VLAN within a network. VLAN 1 exists by default; all other VLANs only exist after they are configured.

- VLAN name (optional): The VLAN name is a text string that describes the VLAN.

- VLAN state (*active* or *suspended*): The state specifies the VLAN transmission status within the switch. In the *suspended* state, VLAN traffic is blocked on all switch ports. The default state is *active*.

VLANs define layer 2 broadcast domains in a layer 2 network, in which each device can receive broadcast frames sent by any other within the domain. Switches accommodating multiple broadcast domains serve as multi-port bridges where each broadcast domain is a distinct virtual bridge. Traffic does not pass directly between different VLANs within a switch or between two switches.

### 20.2.2    VLAN Switching

Ethernet and port channel interfaces are configured as switched ports by default. Switched ports are configurable as members of one or more VLANs. Switched ports ignore all IP-level configuration commands, including IP address assignments.

## 20.2.2.1    VLAN Trunking and Trunk Groups

Trunking extends multiple VLANs beyond the switch through a common interface or port channel.

A trunk group is the set of physical interfaces that comprise the trunk and the collection of VLANs whose traffic is carried on the trunk. The traffic of a VLAN that belongs to one or more trunk groups is carried only on ports that are members of trunk groups to which the VLAN belongs, i.e., VLANs configured in a trunk group are pruned of all ports that are not associated with the trunk group. See the Trunk Ports example section for further details.

---

**Important!**   Be cautious when using allowed VLAN lists or trunk groups to ensure that the VLAN topology is consistent with any Layer-2 control protocol topology, or unpredictable results can occur.

---

VLAN traffic is carried through Ethernet or LAG ports. A port's switchport mode defines the number of VLANs for which the port can carry traffic.

- Access ports carry traffic for one VLAN – the access VLAN. Access ports associate untagged frames with the access VLAN. Access ports drop tagged frames that are not tagged with the access VLAN.

- Trunk ports carry traffic for multiple VLANs. Tag frames specify the VLAN for which trunk ports process packets.

## 20.2.2.2    Q-in-Q Trunking

A Q-in-Q network is a multi-tier layer 2 VLAN network. A typical Q-in-Q network is composed of a service provider network (tier 1) where each node connects to a customer network (tier 2).

802.1ad is a networking standard that supports Q-in-Q networks by allowing multiple 802.1Q tags in an Ethernet frame.

Each interface in a customer network is assigned to a customer-VLAN (c-VLAN). Packets in c-VLANs contain 802.1q tags that switch traffic within the network. c-VLANs access the service provider VLAN (s-VLAN) through a provider switch. Customer switch ports connect to an s-VLAN through provider switch edge ports, which are configured as dot1q ports and operate as follows:

- Inbound traffic (from customer switches): adds an s-VLAN tag, then forwards packets to the provider network.

- Outbound traffic (to customer switches): removes the s-VLAN tag, then forwards packets to the customer network.

## 20.2.2.3    TPID (Configurable Ethertypes)

By default, VLAN-tagged packets carry a tag protocol identifier (TPID) of 0x8100. On some Arista platforms, however, the TPID of a switchport can be modified in accordance with IEEE 802.1ad to allow for the use of 802.1q TPIDs other than 0x8100.  Well known and standard tags include:

- **0x8100**    customer VLAN
- **0x88a8**    service VLAN tag used in provider bridging
- **0x9100**    service VLAN tag used in provider bridging (common, but not standardized)

Other non-standard TPID values may also be configured for interoperability with legacy equipment or non-standard systems. Values range from 0x600 (1536) through 0xFFFF (65535).

Non-default TPID values are most commonly used for provider bridging on a network-to-network interface.

## 20.2.3    VLAN Routing

Each VLAN can be associated with a switch virtual interface (SVI), also called a VLAN interface. The VLAN interface functions in a routed network (layer 3) with an assigned IP subnet address. Connecting different VLANs requires layer 3 networking.

### 20.2.3.1    VLAN Interfaces

A switched virtual interface (SVI) connects to the VLAN segment on the switch to provide layer 3 processing for packets from the VLAN. An SVI can be activated only after it is connected to a VLAN. SVIs are typically configured for a VLAN to a default gateway for a subnet to facilitate traffic routing with other subnets.

In a layer 3 network, each VLAN SVI is associated with an IP subnet, with all stations in the subnet members of the VLAN. Traffic between different VLANs is routed when IP routing is enabled.

### 20.2.3.2    Internal VLANs

A routed port is an Ethernet or port channel interface that functions as a layer 3 interface. Routed ports do not bridge frames nor switch VLAN traffic. Routed ports have IP addresses assigned to them and packets are routed directly to and from the port.

The switch allocates an internal VLAN for an interface when it is configured as a routed port. The internal VLAN is assigned a previously unused VLAN ID. The switch prohibits the subsequent configuration of VLANs and VLAN interfaces with IDs corresponding to allocated internal VLANs.

### 20.2.3.3    VLAN Translation

VLAN translation allows you to map packets from one VLAN to another.

# 20.3    VLAN Configuration Procedures

These sections describe basic VLAN configuration tasks.

## 20.3.1    Creating and Configuring VLANs

The CLI provides two methods of creating VLANs.

- Explicitly through the **vlan** command.
- Implicitly through the **switchport access vlan** command.

  The **switchport access vlan** command generates a warning message when it creates a VLAN.

To create a VLAN, use the **vlan** command in global configuration mode. Valid VLAN numbers range between 1 and 4094. To create multiple VLANs, specify a range of VLAN numbers.

To edit an existing VLAN, enter the **vlan** command with the number of the existing VLAN.

**Example**

- This command creates VLAN 45 and enters VLAN configuration mode for the new VLAN.

```
switch(config)#vlan 45
switch(config-vlan-45)#
```

Use the **name (VLAN configuration mode)** command to assign a name to a VLAN.

**Example**

- These commands assign the name Marketing to VLAN 45.

```
switch(config)#vlan 45
switch(config-vlan-45)#name Marketing
switch(config-vlan-45)#show vlan 45
VLAN Name                            Status    Ports
---- -------------------------------- --------- ------------------------
45   Marketing                        active    Et1

switch(config-vlan-45)#
```

To change a VLAN's state, use the **state** command in VLAN configuration mode.

**Examples**

- These commands suspend VLAN 45. VLAN traffic is blocked on all switch ports.

```
switch(config)#vlan 45
switch(config-vlan-45)#state suspend
switch(config-vlan-45)#show vlan 45
VLAN Name                            Status    Ports
---- -------------------------------- --------- ------------------------
45   Marketing                        suspended

switch(config-vlan-45)#
```

- These commands activate VLAN 45.

```
switch(config)#vlan 45
switch(config-vlan-45)#state active
switch(config-vlan-45)#show vlan 45
VLAN Name                              Status    Ports
---- -------------------------------- --------- ------------------------
45   Marketing                        active    Et1

switch(config-vlan-45)#
```

## 20.3.2    Configuring VLAN Switching

The following describe the configuration of VLAN ports.

### 20.3.2.1    Access Ports

Access ports carry traffic for one VLAN, as designated by a **switchport access vlan** command. Access ports associate untagged frames with the access VLAN. Tagged frames received by the interface are dropped unless they are tagged with the access VLAN.

To configure an interface group as an access port, use the **switchport mode** command.

**Example**

- These commands configure Ethernet interface 1 as an access port.

```
switch(config)#interface ethernet 1
switch(config-if-Et1)#switchport mode access
switch(config-if-Et1)#
```

To specify the port's access VLAN, use the **switchport access vlan** command.

**Examples**

- These commands configure VLAN 15 as the access VLAN for Ethernet interface 5.

```
switch(config)#interface ethernet 5
switch(config-if-Et5)#switchport access vlan 15
switch(config-if-Et5)#
```

- These commands configure Ethernet interface 1 through 3 as access ports that process untagged frames as VLAN 5 traffic.

```
switch(config)#interface Ethernet 1-3
switch(config-if-Et1-3)#switchport mode access
switch(config-if-Et1-3)#switchport access vlan 5
switch(config-if-Et1-3)#show interfaces ethernet 1-3 vlans
Port       Untagged Tagged
Et1        None     23,25
Et2        18       -
Et3        None     14
switch(config-if-Et1-3)#
```

### 20.3.2.2    Trunk Ports

Trunk ports carry traffic for multiple VLANs. Messages use tagged frames to specify the VLAN for which trunk ports process traffic.

- The **vlan trunk list** specifies the VLANs for which the port handles tagged frames. The port drops any packets tagged for VLANs not in the VLAN list.
- The **native vlan** is the VLAN where the port switches untagged frames.

To configure an interface group as a trunk port, use the **switchport mode** command.

**Example**

- These commands configure Ethernet interface 8 as a trunk port.

```
switch(config)#interface ethernet 8
switch(config-if-Et8)#switchport mode trunk
switch(config-if-Et8)#
```

By default all VLANs are permitted on a port configured with 'switchport mode trunk'. To limit the port's VLAN trunk list, use the **switchport trunk allowed vlan** command. Only VLANs in the allowed list will be permitted.

**Examples**

- These commands configure VLAN 15, 20, 21, 22, 40, and 75 as the explicitly permitted VLAN trunk list for Ethernet interface 12-16.

```
switch(config)#interface ethernet 12-16
switch(config-if-Et12-16)#switchport trunk allowed vlan 15,20-22,40,75
switch(config-if-Et12-16)#
```

- These commands explicitly permit VLAN 100 through 120 to the VLAN trunk list for Ethernet interface 14.

```
switch(config)#interface ethernet 14
switch(config-if-Et14)#switchport trunk allowed vlan add 100-120
switch(config-if-Et14)#
```

To specify the port's native VLAN, use the **switchport trunk native vlan** command.

**Example**

- These commands configure VLAN 12 as the native VLAN trunk for Ethernet interface 10.

```
switch(config)#interface ethernet 10
switch(config-if-Et10)#switchport trunk native vlan 12
switch(config-if-Et10)#
```

By default, ports send native VLAN traffic with untagged frames. The **switchport trunk native vlan** command can also configure the port to send native VLAN traffic with tag frames.

**Examples**

- These commands configure Ethernet interface 10 to send native VLAN traffic as tagged.

```
switch(config)#interface ethernet 10
switch(config-if-Et10)#switchport trunk native vlan tag
switch(config-if-Et10)#
```

- These commands configure Ethernet interface 12 as a trunk with VLAN 15 as the native VLAN. The port's trunk list includes all VLANs except 201-300.

```
switch(config)#interface ethernet 12
switch(config-if-Et12)#switchport mode trunk
switch(config-if-Et12)#switchport trunk native vlan 15
switch(config-if-Et12)#switchport trunk allowed vlan except 201-300
switch(config-if-Et12)#
```

**Example**

* Assume that all ports on the switch are configured with switchport mode trunk similar to ethernet 1 and 2 shown below:

```
!
interface ethernet 1
    switchport mode trunk
!
interface ethernet 2
    switchport mode trunk
!
```

Further assume that VLAN 30 is **not** configured as part of a trunk group

```
switch#show vlan
VLAN  Name                             Status    Ports
----- -------------------------------- --------- -------------------------------
1     default                          active    Et1, Et2
30    vlan30                           active    Et1, Et2
```

Now configure VLAN 30 as part of trunk group 30:

```
switch(config)#vlan 30
switch(config-vlan-30)#trunk group 30
```

This updates the VLAN membership for VLAN 30.

```
switch#show vlan
VLAN  Name                             Status    Ports
----- -------------------------------- --------- -------------------------------
1     default                          active    Et1, Et2
30    vlan30                           active
```

**Note:** Vlan 30 is no longer on Et1, Et2 i.e. it has been 'pruned' due to the trunk group command in the vlan configuration.

To permit VLAN 30 on Et1 you need to associate the interface with the trunk group as follows:

```
switch(config-if-Et1)#switchport trunk group 30

Now we see Et1 included in the vlan 30 list

switch#show vlan
VLAN  Name                             Status    Ports
----- -------------------------------- --------- -------------------------------
1     default                          active    Et1, Et2
30    vlan30                           active    Et1
```

The trunk group command is not additive to the allowed vlan command

```
interface ethernet 1
      switchport mode trunk
      switchport trunk allowed vlan 10
      switchport trunk group trunk30

Vlan 30 will not be permitted on the interface as it is not listed in the allowed
vlan list.
```

### 20.3.2.3    Dot1q Tunnel Ports

Dot1q (802.1Q) is a tunneling protocol that encapsulates traffic from multiple customer (c-tag) VLANs in an additional single outer service provider (s-tag) VLAN for transit across a larger network structure that includes traffic from all customers. Tunneling eliminates the service provider requirement that every VLAN be configured from multiple customers, avoiding overlapping address space issues.

Tunneling preserves the inner VLANs through the tunneled network; these inner VLANs are ignored by intermediate devices that make forwarding decisions based only on the outermost VLAN tag (S-Tag)

A dot1q-tunnel port sits at the edge of the tunneled network. Unlike regular access ports, a dot1q-tunnel port does not drop traffic that arrives with 802.1Q tags in place; it ignores existing 802.1Q information and associates arriving traffic (with or without 802.1Q headers) with a new tunnel VLAN ID.

Packets arriving at a tunnel port are encapsulated with an additional 802.1Q tag that can be trunked between multiple devices like any traditional VLAN. When exiting a dot1-tunnel port, the S-Tag is removed to revert the customer traffic to its original tagged or untagged state.

To configure an interface group as a dot1q tunnel port, use the **switchport mode** command.

**Example**

• These commands configure Ethernet interface 12 as a dot1q tunnel port.

```
switch(config)#interface ethernet 12
switch(config-if-Et12)#switchport mode dot1q-tunnel
switch(config-if-Et12)#
```

To specify the dot1q-tunnel port's access VLAN, use the **switchport access vlan** command. The port then handles all inbound traffic as untagged VLAN traffic.

**Example**

• These commands configure VLAN 60 as the access VLAN for Ethernet interface 12.

```
switch(config)#interface ethernet 12
switch(config-if-Et12)#switchport access vlan 60
switch(config-if-Et12)#
```

### 20.3.2.4    TPID Configuration

The default tag protocol identifier (TPID, also called dot1q ethertype) on all switch ports is 0x8100. To configure a different TPID on a port, use the **switchport dot1q ethertype** command. This feature is available only on 7280E and 7500E platforms.

**Important!**    If dot1q tunneling is enabled on the interface, a TPID configured on the interface becomes irrelevant.

**Example**

• In this provider bridging example, Ethernet interface 1 is the user network interface and Ethernet interface 2 is the network-to-network interface. These commands configure dot1q tunneling on Ethernet interface 1 and set the TPID of Ethernet interface 2 to 0x9100.

```
switch(config)#interface ethernet 1
switch(config-if-Et1)#switchport mode dot1q-tunnel
switch(config-if-Et1)#interface ethernet 2
switch(config-if-Et2)#switchport mode trunk
switch(config-if-Et2)#switchport dot1q ethertype 0x9100
switch(config-if-Et2)#
```

In the above configuration, packets from Et1 to Et2 will undergo dot1q-tunneling (stacking of an additional dot1q tag), with an outer TPID of 0x9100 at egress, while packets with outer TPID 0x9100 going from Et2 to Et1 will have the outer tag removed at egress.

### 20.3.2.5    Layer 2 802.1Q Encapsulation

Layer 2 traffic encapsulation is enabled on the configuration mode interface for a specified VLAN through **l2-protocol encapsulation dot1q vlan**.

**Example**

- These commands enable traffic encapsulation for VLAN 200 traffic passing through Ethernet interface 2/5.

  ```
  switch(config)#interface ethernet 5/2
  switch(config-if-Et5/2)#l2-protocol encapsulation dot1q vlan 200
  ```

## 20.3.3    Creating and Configuring VLAN Interfaces

The **interface vlan** command places the switch in VLAN-interface configuration mode for modifying an SVI. An SVI provides a management address point and Layer 3 processing for packets from all VLAN ports.

**Example**

- This command enters VLAN-interface configuration mode for VLAN 12. The command also creates VLAN 12 interface if it was not previously created.

  ```
  switch#config t
  switch(config)#interface vlan 12
  switch(config-if-Vl12)#
  ```

## 20.3.4    Allocating Internal VLANs

The **vlan internal order** command specifies the VLANs that the switch allocates as internal VLANs when configuring routed ports and the order of their allocation. By default, the switch allocates VLANs in ascending order. The default allocation range is between VLAN 1006 and VLAN 4094.

The **no switchport** command converts an Ethernet or port channel interface into a routed port, disabling layer 2 switching for the interface.

**Examples**

- This command configures the switch to allocate internal VLANs in ascending order starting with 1006.

  ```
  switch(config)#vlan internal order ascending
  switch(config)#
  ```

- This command configures the switch to allocate internal VLANs in descending order starting with 4094.

  ```
  switch(config)#vlan internal order descending
  switch(config)#
  ```

- This command configures the switch to allocate internal VLANs in descending order from 4094 through 4000.

  ```
  switch(config)#vlan internal order descending range 4000 4094
  switch(config)#
  ```

## 20.3.5    VLAN Translation

VLAN translation allows you to map packets from one VLAN to another. This can be carried out only on packets having a dot1q header (tagged frames). The translation rewrites the VID field (VLAN ID) in dot1q headers on packets passing through a switched port without changing any other fields.

VLAN translation also supports the ability to translate packets with a dot1q header to the internal VLAN for a routed port. The VLAN in the incoming packets is mapped to the internal VLAN of the routed port and packets egressing the routed port are encapsulated with a dot1q header for the specified VLAN. For egress packets, no priority information is added to the dot1q header and the priority from the incoming encapsulation will be retained.

When configuring the VLAN translation mode, consider the following:

- VLAN translation is only supported for tagged packets.
- BPDUs from STP, LLDP and other protocols are not affected by this mapping.
- VLAN translation is not applicable for access ports.
- Untagged packets entering the switch on the trunk native VLAN are not mapped.
- TPID and VLAN priority does not get re-written during the translation.

### Per-port VLAN Translation on Switched Ports

The **switchport vlan translation** command allows translation of the VLAN tag of traffic entering or exiting a switched port.

To use VLAN translation on a switched port, the port must be configured as a trunk port using the **switchport mode** command.

### Example

- This command configures Ethernet interface 5 as a trunk port.

```
switch(config)#interface ethernet 5
switch(config-if-Et5)#switchport mode trunk
switch(config-if-Et5)#
```

By default, the translation is bidirectional: packets ingressing an interface through VLAN A are internally mapped to VLAN B; VLAN B packets egressing the same interface are mapped to VLAN A.

### Examples

- These commands map Ethernet interface 5 traffic with dot1q tag 50 to bridging VLAN 60.

```
switch(config)#interface ethernet 5
switch(config-if-Et5)# switchport vlan translation 50 60
switch(config-if-Et5)#
```

- These commands provides multiple 1:1 VLAN mappings under an interface.

```
switch(config)#interface ethernet 5
switch(config-if-Et5)# switchport vlan translation 50 60
switch(config-if-Et5)# switchport vlan translation 61 71
switch(config-if-Et5)# switchport vlan translation 62 72
switch(config-if-Et5)#
```

- These commands translate only incoming packets.

```
switch(config)#interface ethernet 5
switch(config-if-Et5)# switchport vlan translation in 50 60
switch(config-if-Et5)#
```

- These commands translate only egress packets.

```
switch(config)#interface ethernet 5
switch(config-if-Et5)#switchport vlan translation out 60 50
switch(config-if-Et5)#
```

### Per-port VLAN Translation on Routed Ports

On routed ports, the **encapsulation dot1q vlan** command (permitted only on routed ports) configures the VLAN on the interface to act as the native VLAN. This command will map packets ingressing with the specified VLAN ID to the internal VLAN ID of the routed port. All traffic egressing out of the routed port will be tagged with the VLAN ID specified in the command.

### Examples

- These commands translate between VLAN 50 and the internal VLAN for Ethernet interface 5 (a routed port).

```
switch(config)#interface ethernet 5
switch(config-if-Et5)# no switchport
switch(config-if-Et5)# encapsulation dot1q vlan 50
switch(config-if-Et5)#
```

## 20.4      VLAN Configuration Commands

**Global VLAN Configuration Commands**

- interface vlan
- vlan
- vlan internal order

**VLAN Configuration Mode Commands**

- name (VLAN configuration mode)
- state
- trunk group

**Layer 2 Interface (Ethernet and Port Channel) Configuration Commands**

- switchport access vlan
- switchport mode
- switchport trunk allowed vlan
- switchport trunk group
- switchport trunk native vlan
- switchport vlan translation

**VLAN Interface Configuration Mode Commands**

- autostate
- encapsulation dot1q vlan
- l2-protocol encapsulation dot1q vlan
- name (VLAN configuration mode)
- pvlan mapping

**Show Commands**

- show dot1q-tunnel
- show interfaces switchport
- show interfaces switchport backup-link
- show interfaces trunk
- show interfaces vlans
- show pvlan mapping interfaces
- show vlan
- show vlan brief count
- show vlan dynamic
- show vlan internal allocation policy
- show vlan internal usage
- show vlan trunk group

# autostate

When autostate is *enabled*, the VLAN interface will be up when:

- the corresponding VLAN exists and is in the active state.
- one or more layer 2 ports in the VLAN are up and in spanning-tree forwarding state.
- the VLAN interface exists and is not in a **shutdown** state.

Autostate is *enabled* by default. When autostate is *disabled*, the VLAN interface is forced to be active.

- The **no autostate** command disables autostate on the configuration mode interface. The **no autostate** command is stored to *running-config*.
- The **autostate** command enables the autostate function on the configuration mode VLAN SVI by removing the corresponding **no autostate** statement from *running-config*.
- The **default autostate** command restores the autostate default state of *enabled* by removing the corresponding **no autostate** statement from *running-config*.

## Command Mode

Interface-VLAN Configuration

## Command Syntax

```
autostate
no autostate
default autostate
```

## Guidelines

Autostate should be disabled on SVIs configured as an MLAG local interface.

## Examples

- These commands disable autostate on VLAN 100.

```
switch(config)#interface vlan 100
switch(config-if-Vl100)#no autostate
switch(config-if-Vl100)#
```

- These commands enable autostate on VLAN 100.

```
switch(config)#interface vlan 100
switch(config-if-Vl100)#autostate
switch(config-if-Vl100)#
```

# encapsulation dot1q vlan

### Routed Port VLAN Translation

In the configuration mode for an Ethernet or port channel interface, the **encapsulation dot1q vlan** translates packets with a dot1q header to the internal VLAN for a routed port. The VLAN in the incoming packets is mapped to the internal VLAN of the routed port, and packets egressing the routed port are encapsulated with a dot1q header for the specified VLAN. For egress packets, no priority information is added to the dot1q header and the priority from the incoming encapsulation will be retained.

### Subinterface VLAN Assignment

When used in the configuration mode for an Ethernet or port channel subinterface, however, the **encapsulation dot1q vlan** command assigns a dot1q tag to the subinterface. Traffic ingressing on the parent interface with that dot1q tag will then be sent to the configured subinterface. See **Subinterfaces** and **Subinterface Configuration** for details.

The **no encapsulation dot1q vlan** and **default encapsulation dot1q vlan** commands restore the default VLAN to the configuration mode interface by removing the corresponding **encapsulation dot1q vlan** command from *running-config*.

### Command Mode

Interface-Ethernet Configuration
Interface-port-channel Configuration
Subinterface-Ethernet Configuration
Subinterface-port-channel Configuration

### Command Syntax

```
encapsulation dot1q vlan vlan_id
no encapsulation dot1q vlan
default encapsulation dot1q vlan
```

### Parameters

- *vlan_id*    For VLAN translation, the ID of the external VLAN to be translated; for subinterface configuration, the VLAN of the subinterface. Values range from 1 to 4094.

### Example

- These commands translate between VLAN 50 and the internal VLAN for Ethernet interface 5 (a routed port).

```
switch(config)#interface ethernet 5
switch(config-if-Et5)# no switchport
switch(config-if-Et5)# encapsulation dot1q vlan 50
switch(config-if-Et5)#
```

- These commands assign packets ingressing on Ethernet interface 1/1 with VLAN ID 100 to Ethernet subinterface 1/1.1.

```
switch(config)#interface ethernet1/1.1
switch(config-if-Et1/1.1)# no switchport
switch(config-if-Et1/1.1)#encapsulation dot1q vlan 100
switch(config-if-Et1/1.1)#
```

## interface vlan

The **interface vlan** command places the switch in VLAN-interface configuration mode for modifying parameters of the switch virtual interface (SVI). An SVI provides Layer 3 processing for packets from all ports associated with the VLAN. There is no physical interface for the VLAN.

When entering configuration mode to modify existing SVIs, the command can specify multiple interfaces. The command creates an SVI if the specified interface does not exist prior to issuing the command. When creating an SVI, the command can only specify a single interface.

The **no interface vlan** command deletes the specified SVI interfaces from *running-config*. The **default interface vlan** commands remove all configuration statements for the specified SVI interfaces from *running-config* without deleting the interfaces.

### Command Mode

Global Configuration

### Command Syntax

```
interface vlan v_range
no interface vlan v_range
default interface vlan v_range
```

### Parameter

- *v_range*    VLAN interfaces (number, range, or comma-delimited list of numbers and ranges).

  VLAN number ranges from 1 to 4094.

### Restrictions

Internal VLANs: A VLAN interface cannot be created or configured for internal VLAN IDs. The switch rejects any **interface vlan** command that specifies an internal VLAN ID.

### Example

- This example creates an SVI for VLAN 12:

```
switch#config
switch(config)#interface vlan 12
switch(config-if-Vl12)#
```

# l2-protocol encapsulation dot1q vlan

The **l2-protocol encapsulation dot1q vlan** command enables Layer 2 802.1Q traffic encapsulation on the configuration mode interface for a specified VLAN. The default VLAN for all interfaces is VLAN 1.

The **no l2-protocol encapsulation dot1q vlan** and **default l2-protocol encapsulation dot1q vlan** commands disable the specified encapsulation on the configuration mode interface by removing the corresponding **l2-protocol encapsulation dot1q vlan** command from *running-config*.

## Command Mode

Interface-Ethernet Configuration
Interface-Port-channel Configuration

## Command Syntax

```
l2-protocol encapsulation dot1q vlan vlan_id
no l2-protocol encapsulation dot1q vlan
default l2-protocol encapsulation dot1q vlan
```

## Parameters

* *vlan_id*    the ID of the native VLAN. Values range from 1 to 4094.

## Example

* These commands enable 802.1Q encapsulation of traffic on VLAN 200.

```
switch(config)#interface ethernet 5/2
switch(config-if-Et5/2)#l2-protocol encapsulation dot1q vlan 200
switch(s1)(config-if-Et5/2)#show active
interface Ethernet5/2
   l2-protocol encapsulation dot1q vlan 200
switch(config-if-Et5/2)#
```

# name (VLAN configuration mode)

The **name** command configures the VLAN name. The name can have up to 32 characters. The default name for VLAN 1 is *default*. The default name for all other VLANs is VLANxxxx, where xxxx is the VLAN number. The default name for VLAN 55 is VLAN0055. The **show vlan** command displays the VLAN name.

The **name** command accepts all characters except the space.

The **no name** and **default name** commands restore the default name by removing the **name** command from *running-config*.

## Command Mode

VLAN Configuration

## Command Syntax

```
name label_text
no name
default name
```

## Parameters

- *label_text*    character string assigned to name attribute. Maximum length is 32 characters. The space character is not permitted in the name string.

## Examples

- These commands assign corporate_100 as the name for VLAN 25, then displays the VLAN name.

```
switch(config)#vlan 25
switch(config-vlan-25)#name corporate_100
switch(config-vlan-25)#show vlan 25
VLAN  Name                             Status    Ports
----- -------------------------------- --------- -------------------------------
25    corporate_100                    active

switch(config-vlan-25)#
```

# pvlan mapping

The **pvlan mapping** command maps a switch virtual interface (SVI) available in the primary VLAN to the secondary VLAN or VLANs in the VLAN configuration mode. The **show pvlan mapping interfaces** command displays the list of mapped VLANs.

The **no pvlan mapping** and **default pvlan mapping** commands restore the default state of the private VLAN mapping.

## Command Mode

VLAN Configuration

## Command Syntax

```
pvlan mapping {add | remove | vlan ID}
no pvlan mapping {add | remove | vlan ID}
default pvlan mapping {add | remove | vlan ID}
```

## Parameters

*   **add**    adding VLANs to the PVLAN mapping of the current VLAN interface.
*   **remove**    removing VLANs from the PVLAN mapping of the current VLAN interface.
*   *vlan ID*    The secondary VLAN IDs of the private VLAN mapping. The IDs range from 1 to 4094.

## Related Commands

*   show pvlan mapping interfaces

## Examples

*   These commands assign a secondary VLAN ID of 50 to the primary VLAN.

```
switch(config)#vlan 25
switch(config-vlan-25)#pvlan mapping 50
switch(config-vlan-25)#
```

# show dot1q-tunnel

The **show dot1q-tunnel** command displays the ports that are configured in dot1q-tunnel switching mode. The **switchport mode** command configures the switching mode for the configuration mode interface.

## Command Mode

EXEC

## Command Syntax

```
show dot1q-tunnel [INTERFACE]
```

## Parameters

- *INTERFACE*    Interface type and numbers. Options include:
    - \<no parameter\>    Display information for all interfaces.
    - **ethernet** *e_range*    Ethernet interface range specified by *e_range*.
    - **loopback** *l_range*    Loopback interface specified by *l_range*.
    - **management** *m_range*    Management interface range specified by *m_range*.
    - **port-channel** *p_range*    Port-Channel Interface range specified by *p_range*.
    - **vlan** *v_range*    VLAN interface range specified by *v_range*.
    - **vxlan** *vx_range*    VXLAN interface range specified by *vx_range*.

    Valid *range* formats include number, number range, or comma-delimited list of numbers and ranges.

## Example

- This command displays the ports that are configured in dot1q-tunnel switching mode.

```
switch>show dot1q-tunnel
dot1q-tunnel mode LAN Port (s)
------------------------------
Po4
Po21
Po22
switch>
```

# show interfaces switchport

The **show interfaces switchport** command displays the switching configuration and operational status of the specified ports.

## Command Mode

EXEC

## Command Syntax

`show interfaces [`***INTERFACE***`] switchport`

## Parameters

- ***INTERFACE***    Interface type and numbers. Options include:
  - <no parameter>    Display the switching status for all interfaces.
  - **ethernet** *e_range*    Ethernet interface range specified by *e_range*.
  - **loopback** *l_range*    Loopback interface specified by *l_range*.
  - **management** *m_range*    Management interface range specified by *m_range*.
  - **port-channel** *p_range*    Port-Channel Interface range specified by *p_range*.
  - **vlan** *v_range*    VLAN interface range specified by *v_range*.

    Valid *e_range*, *l_range*, *m_range*, *p_range*, and *v_range* formats include number, number range, or comma-delimited list of numbers and ranges.

## Example

- This command displays the switching status for all interfaces.

```
switch(config)#show interface switchport
Default switchport mode: access

Name: Et5/1
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
MAC Address Learning: enabled
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: disabled
Trunking VLANs Enabled: ALL
Static Trunk Groups:
Dynamic Trunk Groups:

Name: Et5/2
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
MAC Address Learning: enabled
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: disabled
Trunking VLANs Enabled: ALL
Static Trunk Groups:
Dynamic Trunk Groups:

[...]

switch(config)#
```

- This command displays the switching status of port channel interfaces 21 and 22.

```
switch>show interface port-channel 21-22 switchport
Name: Po21
Switchport: Enabled
Administrative Mode: tunnel
Operational Mode: tunnel
Access Mode VLAN: 1 (inactive)
Trunking Native Mode VLAN: 100 (VLAN0100)
Administrative Native VLAN tagging: disabled
Trunking VLANs Enabled: ALL
Trunk Groups: foo

Name: Po22
Switchport: Enabled
Administrative Mode: tunnel
Operational Mode: tunnel
Access Mode VLAN: 1 (inactive)
Trunking Native Mode VLAN: 1 (inactive)
Administrative Native VLAN tagging: disabled
Trunking VLANs Enabled: ALL
Trunk Groups:

switch>
```

# show interfaces switchport backup-link

The **show interfaces switchport backup-link** command displays interfaces that are configured as switchport backup pairs and the operational status of each interface. For each pair, the command displays the names, roles, status, and VLAN traffic of each interface.

## Command Mode

EXEC

## Command Syntax

```
show interfaces [INTERFACE] switchport backup-link
show interfaces switchport backup-link [module {Fabric f_num | Linecard lc_num |
Supervisor svr_num | Switchcard | <1-2> | <3-6>}]
```

## Parameters

* *INTERFACE*   Interface type and numbers. Options include:
  * <no parameter>   Display information for all interfaces.
  * **ethernet** *e_range*   Ethernet interface range specified by *e_range*.
  * **loopback** *l_range*   Loopback interface specified by *l_range*.
  * **management** *m_range*   Management interface range specified by *m_range*.
  * **port-channel** *p_range*   Port-Channel Interface range specified by *p_range*.
  * **vlan** *v_range*   VLAN interface range specified by *v_range*.

    Valid *e_range*, *l_range*, *m_range*, *p_range*, and *v_range* formats include number, number range, or comma-delimited list of numbers and ranges.

* **module**   Displays interfaces of the specified module. Options include:
  * **Fabric** *f_num*   Displays interfaces of the specified fabric module. Value ranges from 1 to 6.
  * **Linecard** *lc_num*   Displays interfaces of the specified linecard module. Value ranges from 3 to 6.
  * **Supervisor** *svr_num*   Displays interfaces of the specified supervisor module. Accepted values are 1 and 2.
  * **Switchcard**   Displays interfaces of switchcard modules.
  * *<1-2>*   Displays interfaces of the specified supervisor module.
  * *<3-6>*   Displays interfaces of the specified linecard module.

## Display Values

* **State**   Operational status of the interface. Values include:
  * *Up*   Spanning tree mode is *backup*, interface status is *up*.
  * *Down*   Spanning tree mode is *backup*, interface status is *down*.
  * *Inactive Configuration*   The spanning tree mode is not *backup*.
* **Forwarding vlans**   VLANs forwarded by the interface. Depends on interface operation status and prefer option specified by the **switchport backup** command.

### Example

• This command displays the configured switchport primary-backup pairs.

```
switch>show interfaces switchport backup-link
Switch backup interface pair: Ethernet3/17, Ethernet3/8
Primary Interface: Ethernet3/17     State:  Inactive Configuration
Backup Interface:  Ethernet3/8      State:  Inactive Configuration
Preemption delay: 0 milliseconds
Mac move burst size: 0
Mac move burst interval: 20 milliseconds
Mac move destination: ff:ff:ff:ff:ff:ff
```

• This command displays interfaces of the module for linecard 4.

```
switch(config)#show int switchport backup-link module Linecard 4
Switch backup interface pair: Ethernet4/19/1, Ethernet4/19/2
Primary Interface: Ethernet4/19/1   State:  Inactive Configuration
Backup Interface:  Ethernet4/19/2   State:  Inactive Configuration
Preemption delay: 0 milliseconds
Mac move burst size: 0
Mac move burst interval: 20 milliseconds
Mac move destination: ff:ff:ff:ff:ff:ff
```

# show interfaces trunk

The **show interfaces trunk** command displays configuration and status information for interfaces configured in switchport trunk mode.

## Command Mode

EXEC

## Command Syntax

`show interfaces [INTERFACE] trunk`

## Parameters

*   *INTERFACE*    Interface type and numbers. Options include:

    *   <no parameter>    Display information for all interfaces.
    *   **ethernet** *e_range*    Ethernet interface range specified by *e_range*.
    *   **management** *m_range*    Management interface range specified by *m_range*.
    *   **port-channel** *p_range*    Port-Channel Interface range specified by *p_range*.

        Valid *e_range*, *m_range*, and *p_range* formats include number, number range, or comma-delimited list of numbers and ranges.

## Example

*   This command displays the trunk status for all interfaces configured in switchport trunk mode.

```
switch>show interfaces trunk
Port            Mode            Status          Native vlan
Po1             trunk           trunking        1
Po2             trunk           trunking        1

Port            Vlans allowed
Po1             1-15
Po2             16-30

Port            Vlans allowed and active in management domain
Po1             1-10
Po2             21-30


Port            Vlans in spanning tree forwarding state
Po1             1-10
Po2             21-30

switch>
```

# show interfaces vlans

The **show interfaces vlans** command displays a table that lists the VLANs that are carried by the specified interfaces. Interfaces that do not carry VLANs are not listed in the table. The table lists the untagged (native or access) and tagged VLANs for each interface.

## Command Mode

  EXEC

## Command Syntax

  `show interfaces [`*`INT_NAME`*`] vlans`

## Parameters

•  *INT_NAME*  Interface type and number. Values include

  •  **ethernet** *e_num*  Ethernet interface specified by *e_num*.

  •  **management** *m_num*  Management interface specified by *m_num*.

  •  **port-channel** *p_num*  Port-Channel Interface specified by *p_num*.

## Example

•  This command displays the VLANs carried by all L2 ports.

```
switch>show interfaces vlans
Port       Untagged Tagged
Et9        3910     -
Et11       3912     -
Et16       500      -
Et17       3908     -
Et18       3908     -
Po1        1        101-102,500,721,3000,
Po2        101      -
Po4        3902     -
Po5        3903     -
Po6        3992     -
Po7        661      -
Po8        3911     -
```

## show pvlan mapping interfaces

The **show interfaces vlans** command displays information about the private VLAN mapping interfaces.

### Command Mode

EXEC

### Command Syntax

```
show pvlan mapping interfaces
```

### Example

• This command displays information about the private VLAN mapping interfaces.

```
switch(config)#int vlan 50
switch(config-if-Vl50)#pvlan mapping 70
switch(config-if-Vl50)#show pvlan mapping interfaces
Interface    Secondary Vlans
---------    ---------------
Vlan50       70
```

## show vlan

The **show vlan** command displays the VLAN ID, name, status, and member ports of all configured VLANs. The command only displays active ports by default; by specifying *configured-ports*, the command displays all ports that are members of a configured VLAN regardless of their activity status, including Ethernet ports that are members of a port channel.

### Command Mode

EXEC

### Command Syntax

```
show vlan [VLAN_LIST] [PORT_ACTIVITY]
```

### Parameters

*   *VLAN_LIST*    List of VLANs displayed by command. Options include:
    *   <no parameter>    all VLANs.
    *   *v_range*    VLANs specified by *v_range*.
    *   **id** *v_range*    VLANs specified by *v_range*.
    *   **name** *v_name*    VLANs specified by the VLAN name *v_name*.

    *v_range* formats include number, number range, or comma-delimited list of numbers and ranges.
*   *PORT_ACTIVITY*    Ports listed in table. Options include:
    *   <no parameter>    table displays only active ports (same as **active-configuration** option).
    *   **active-configuration**    table displays only active ports.
    *   **configured-ports**    table displays all configured ports.

### Display Values

*   **VLAN**    The VLAN ID.
*   **Name**    The name of the VLAN.
*   **Status**    The status of the VLAN.
*   **Ports**    The ports that are members of the VLAN.

### Example

*   This command displays status and ports of VLANs 1-1000.

```
switch>show vlan 1-1000
VLAN  Name                             Status    Ports
----- -------------------------------- --------- --------------------------------
1     default                          active    Po1
184   fet.arka                         active    Cpu, Po1, Po2
262   mgq.net                          active    PPo2, Po1
512   sant.test                        active    Cpu, Et16, Po1
821   ipv6.net                         active    Cpu, Po1, Po7

switch>
```

*   This command displays the list of all the member interfaces under each SVI.

```
switch# show vlan
VLAN  Name                             Status    Ports
----- -------------------------------- --------- --------------------------------
1     default                          active
2148  VLAN2148                         active    Cpu, Et1, Et26
2700  VLAN2700                         active    Cpu, Et18
```

## show vlan brief count

The **show vlan brief count** command displays the number of VLANs that are configured on the switch.

### Command Mode

EXEC

### Command Syntax

`show vlan brief count`

### Example

• This command displays the number of VLANs on the switch.

```
switch>show vlan brief count
Number of existing VLANs          : 18

switch>
```

## show vlan dynamic

The **show vlan dynamic** command displays the source and quantity of dynamic VLANs on the switch. Dynamic VLANs support VM Tracer monitoring sessions.

### Command Mode

EXEC

### Command Syntax

`show vlan dynamic`

### Example

• This command displays the source and quantity of dynamic VLANs on the switch.

```
switch>show vlan dynamic
Dynamic VLAN source     VLANS
vmtracer-poc            88
switch>
```

## show vlan internal allocation policy

The **show vlan internal allocation policy** command displays the method the switch uses to allocate VLANs to routed ports. The **vlan internal order** command configures the allocation method.

The allocation method consists of two configurable components:

- range: the list of VLANs that are allocated to routed ports.
- direction: the direction by which VLANs are allocated (ascending or descending).

### Command Mode

EXEC

### Command Syntax

```
show vlan internal allocation policy
```

### Example

- This command displays the internal allocation policy.

```
switch>show vlan internal allocation policy
Internal VLAN Allocation Policy: ascending
Internal VLAN Allocation Range: 1006-4094
switch>
```

## show vlan internal usage

The **show vlan internal usage** command shows the VLANs that are allocated as internal VLANs for routed ports.

A routed port is an Ethernet or port channel interface that is configured as a layer 3 interface. Routed ports do not bridge frames and are not members of any VLANs. Routed ports can have IP addresses assigned to them and packets are routed directly to and from the port.

When an interface is configured as a routed port, the switch allocates an SVI with a previously unused VLAN ID. The switch prohibits the configuration of VLANs with numbers corresponding to internal VLAN interfaces allocated to a routed port. VLAN interfaces corresponding to SVIs allocated to a routed port cannot be configured by VLAN interface configuration mode commands.

### Command Mode

EXEC

### Command Syntax

```
show vlan internal usage
```

### Example

- This command displays the VLANs that are allocated to routed ports.

```
switch>show vlan internal usage
1006  Ethernet3
1007  Ethernet4
switch>
```

## show vlan trunk group

The **show vlan trunk group** command displays the trunk group membership of the specified VLANs.

### Command Mode

EXEC

### Command Syntax

```
show vlan [VLAN_LIST] trunk group
```

### Parameters

* *VLAN_LIST*      VLAN list. Options include:
  * <no parameter>       all VLANs.
  * *v_range*     VLANs specified by *v_range*.
  * **id** *v_range*     VLANs specified by *v_range*.
  * **name** *v_name*     VLANs specified by the VLAN name *v_name*.

### Display Values

* **VLAN**     VLAN ID.
* **Trunk Group**s      Trunk groups associated with the listed VLANs.

### Example

* This command displays the trunk group membership of all configured VLANs.

```
switch>show vlan trunk group
VLAN      Trunk Groups
----      --------------------------------------------------------------------
5
10        first_group
12
40        second_group
100       third_group
101       middle_group
102
200

switch>
```

## state

The **state** command configures the VLAN transmission state of the configuration mode VLAN.

• **Active** state: Ports forward VLAN traffic.

• **Suspend** state: Ports block VLAN traffic.

The default transmission status is *active*.

The **no state** command restores the default VLAN transmission state to the configuration mode VLAN by removing the corresponding **state** command from *running-config*.

### Command Mode

VLAN Configuration

### Command Syntax

```
state OPERATION_STATE
no state
default state
```

### Parameters

• *OPERATION_STATE*    VLAN transmission state. Options include:

• **active**    VLAN traffic is forwarded

• **suspend**    LAN traffic is blocked.

### Example

• These commands suspend VLAN traffic on VLANs 100-102.

```
switch(config)#vlan 100-102
switch(config-vlan-100-102)#state suspend
switch(config-vlan-100-102)#
```

# switchport dot1q ethertype

The **switchport dot1q ethertype** command configures the tag protocol identifier (TPID, also known as a dot1q ethertype), of the configuration mode interface. By default, all switch ports use the standard TPID of 0x8100.

The **no switchport dot1q ethertype** and **default switchport dot1q ethertype** commands restore the TPID to 0x8100 by removing the corresponding **switchport dot1q ethertype** statement from *running-config*.

## Command Mode

Interface-Ethernet Configuration

## Command Syntax

```
switchport dot1q ethertype ethertype
no switchport dot1q ethertype
default switchport dot1q ethertype
```

## Parameters

*   *ethertype*    ethertype number (TPID). Value ranges from 0x600 (1536) through 0xFFFF (65535), and can be entered in decimal or hexadecimal notation. Value is stored and displayed in hexadecimal form; the default value is 0x8100.

## Example

*   These commands configure 0x9100 as the TPID of Ethernet interface 5.

```
switch(config)#interface ethernet 5
switch(config-if-Et5)#switchport dot1q ethertype 0x9100
switch(config-if-Et5)#
```

## switchport access vlan

The **switchport access vlan** command specifies the access VLAN of the configuration mode interface. Ethernet or port channel interfaces that are in access mode are members of only the access VLAN. Untagged frames that the interface receives are associated with the access VLAN. Frames tagged with the access VLAN are also associated with the access VLAN. The interface drops all other tagged frames that it receives. By default, VLAN 1 is the access VLAN of all Ethernet and port channel interfaces.

An interface's access mode is effective only when the interface is in access mode or dot1q-tunnel mode, as specified by the switchport mode command. Interfaces in dot1q-tunnel mode handle inbound traffic as untagged traffic and associate all traffic with the access VLAN. Interfaces configured to switchport trunk mode maintain and ignore existing switchport access commands.

The **no switchport access vlan** and **default switchport access vlan** commands restore VLAN 1 as the access VLAN of the configuration mode interface by removing the corresponding **switchport access vlan** statement from *running-config*.

### Command Mode

Interface-Ethernet Configuration
Interface-Port-channel Configuration

### Command Syntax

```
switchport access vlan v_num
no switchport access vlan
default switchport access vlan
```

### Parameters

- *v_num*    number of access VLAN. Value ranges from 1 to 4094. Default is 1.

### Example

- These commands assign VLAN 100 as the access VLAN to Ethernet interface 5.

```
switch(config)#interface ethernet 5
switch(config-if-Et5)#switchport access vlan 100
switch(config-if-Et5)#
```

## switchport mode

The **switchport mode** command specifies the switching mode of the configuration mode interface. The switch supports five switching modes: access, trunk, dot1q-tunnel, tap, and tool.

- **Access switching mode**: The interface is a member of one VLAN, called the access VLAN, as specified by the **switchport access vlan** command. Tagged frames received on the interface are dropped unless they are tagged with the access VLAN. Frames transmitted from the interface are always untagged.

- **Trunk switching mode**: The interface may be a member of multiple VLANs, as configured by the **switchport trunk allowed vlan** command. Untagged traffic is associated with the interface's native VLAN, as configured with the **switchport trunk native vlan** command.

- **Dot1q-tunnel switching mode**: The interface treats all inbound packets as untagged traffic and handles them as traffic of its access VLAN, as specified by the **switchport access vlan** command.

- **Tap mode**: The interface operates as a tap port. Tap ports receive traffic for replication on one or more tool ports.The interface may be a member of multiple VLANs, as configured by the **switchport tap allowed vlan** command. Untagged traffic is associated with the interface's native VLAN, as configured with the **switchport tap native vlan** command.

  Tap ports are in STP forwarding state and prohibit egress traffic. MAC learning, control plane interaction and traps for inbound traffic are disabled.

- **Tool mode**: The interface operates as a tool port. Tool ports replicate traffic received by tap ports. The interface may be a member of multiple VLANs, as configured by the **switchport tool allowed vlan** command. MAC learning, control plane interaction and traps for inbound traffic are disabled.

  Tool ports are in STP forwarding state and prohibit ingress traffic that uses port settings.

The status of switchport configured ports depends on the switch's tap aggregation mode (which can be viewed by using the **mode (tap-agg configuration mode)** command):

- tap aggregation mode enabled: tap and tool ports are enabled. Switching ports are errdisabled.

- tap aggregation mode disabled: tap and tool ports are errdisabled. Switching ports are enabled.

The **no switchport mode** and **default switchport mode** commands return the configuration mode interface to its default setting as an access port by deleting the corresponding **switchport mode** command from *running-config*.

### Command Mode

Interface-Ethernet Configuration
Interface-Port-channel Configuration

### Command Syntax

```
switchport mode MODE_TYPE
no switchport mode
default switchport mode
```

### Parameters

- *MODE_TYPE*    switching mode of the configuration mode interfaces. Options include:
  - **access**    access switching mode.
  - **dot1q-tunnel**    dot1q-tunnel switching mode.
  - **tap**    tap switching mode.
  - **tool**    tool switching mode.
  - **trunk**    trunk switching mode.

### Restrictions

Dot1q-tunnel switching mode is not available on Petra platform switches.

Tap aggregation (tap and tool modes) is available on FM6000 and Arad platform switches.

### Example

* These commands configure Ethernet 4 interface as a trunk port.

```
switch(config)#interface ethernet 4
switch(config-if-Et4)#switchport mode trunk
switch(config-if-Et4)#
```

# switchport trunk allowed vlan

The **switchport trunk allowed vlan** command creates or modifies the list of VLANs for which the configuration mode interface, in trunk mode, handles tagged traffic. By default, interfaces handle tagged traffic for all VLANs. Command settings persist in *running-config* without taking effect when the switch is in tap aggregation mode or the interface is not in trunk mode.

The **no switchport trunk allowed vlan** and **default switchport trunk allowed vlan** commands restore the trunk mode default allowed VLAN setting of *all* by removing the corresponding **switchport trunk allowed vlan** statement from *running-config*.

## Command Mode

Interface-Ethernet Configuration
Interface-Port-channel Configuration

## Command Syntax

```
switchport trunk allowed vlan EDIT_ACTION
no switchport trunk allowed vlan
default switchport trunk allowed vlan
```

## Parameters

* *EDIT_ACTION*     modifications to the VLAN list.

  * *v_range*     Creates VLAN list from *v_range*.

  * **add** *v_range*     Adds specified VLANs to current list.

  * **all**     VLAN list contains all VLANs.

  * **except** *v_range*     VLAN list contains all VLANs except those specified.

  * **none**     VLAN list is empty (no VLANs).

  * **remove** *v_range*     Removes specified VLANs from current list.

    Valid *v_range* formats include number, range, or comma-delimited list of numbers and ranges.

## Example

* These commands create the trunk mode allowed VLAN list of 6-10 for Ethernet interface 14, then verifies the VLAN list.

```
switch(config)#interface ethernet 14
switch(config-if-Et14)#switchport trunk allowed vlan 6-10
switch(config-if-Et14)#show interfaces ethernet 14 switchport
Name: Et14
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Access Mode VLAN: 1 (inactive)
Trunking Native Mode VLAN: 1 (inactive)
Administrative Native VLAN tagging: disabled
Trunking VLANs Enabled: 6-10
Trunk Groups:

switch(config-if-Et14)#
```

## switchport trunk group

The **switchport trunk group** command assigns the configuration mode interface to the specified trunk group. Trunk group ports handle traffic of the VLANs assigned to the group.

The **no switchport trunk group** and **default switchport trunk group** commands remove the configuration mode interface from the specified trunk group by deleting the corresponding statement from *running-config*. If the command does not specify a trunk group, the interface is removed from all trunk groups to which it is assigned.

**Note**    On platforms which support the use of port channels as mirror destinations, a port channel which is being used as a mirror destination *must not* be assigned to an MLAG.

### Command Mode

Interface-Ethernet Configuration
Interface-Port-channel Configuration

### Command Syntax

```
switchport trunk group group_name
no switchport trunk group [group_name]
default switchport trunk group [group_name]
```

### Parameters

- *group_name*    trunk group name.

### Example

- These commands assign port channel 4 to trunk group *fe-1*.

```
switch(config)#interface port-channel 4
switch(config-if-Po4)#switchport trunk group fe-1
switch(config-if-Po4)#
```

## switchport trunk native vlan

The **switchport trunk native vlan** command specifies the trunk mode native VLAN for the configuration mode interface. Interfaces in trunk mode associate untagged frames with the native VLAN. Trunk mode interfaces can also be configured to drop untagged frames. The default native VLAN for all interfaces is VLAN 1.

The **no switchport trunk native vlan** and **default switchport trunk native vlan** commands restore VLAN 1 as the trunk mode native VLAN to the configuration mode interface by removing the corresponding **switchport trunk native vlan** command from *running-config*.

### Command Mode

    Interface-Ethernet Configuration
    Interface-Port-channel Configuration

### Command Syntax

```
switchport trunk native vlan VLAN_ID
no switchport trunk native vlan
default switchport trunk native vlan
```

### Parameters

*   *VLAN_ID*    the ID of the native VLAN. Options include

    *   v_num    VLAN number. Value ranges from 1 to 4094

    *   **tag**    interface drops all untagged frames.

### Example

*   These commands configure VLAN 100 as the native VLAN for port channel 21.

```
switch(config)#interface port-channel 21
switch(config-if-Po21)#switchport trunk native vlan 100
switch(config-if-Po21)#
```

# switchport vlan translation

The **switchport vlan translation** command allows you to map packets from one VLAN to another using VLAN translation. This can be carried out only on packets having a dot1q header (tagged frames). The translation rewrites the VID field (VLAND ID) in dot1q headers on packets passing through a switched port without changing any other fields.

By default, the translation is bidirectional: packets ingressing an interface through VLAN A are internally mapped to VLAN B; VLAN B packets egressing the same interface are mapped to VLAN A.

To use VLAN translation on a switched port, the port must be configured as a trunk port using the **switchport mode** command.

VLAN translation on routed ports is accomplished through the **encapsulation dot1q vlan** command.

The **no switchport vlan translation** and **default switchport vlan translation** commands remove VLAN mapping by removing the switchport vlan translation command from *running-config*.

## Command Mode

Interface-Ethernet Configuration
Interface-Port-channel Configuration

## Command Syntax

```
switchport vlan translation [DIRECTION] incoming_vlanid new_vlanid
no switchport vlan translation incoming_vlanid new_vlanid
no switchport vlan translation DIRECTION incoming_vlanid
default switchport vlan translation incoming_vlanid new_vlanid
default switchport vlan translation DIRECTION incoming_vlanid
```

## Parameters

* *DIRECTION*    transmission direction of traffic to be translated.

    * <no parameter>    translates the specified VLAN IDs for transmitted and received traffic.

    * **in**    translates the specified VLAN IDs for received traffic only.

    * **out**    translates the specified VLAN IDs for transmitted traffic only.

* *incoming_vlanid*    The VLAN ID to be translated. Value ranges from 1 to 4094.

* *new_vlanid*    The new VLAN ID or bridging VLAN ID which will be used internally. Value ranges from 1 to 4094.

## Example

* These commands translate only incoming packets, changing the VID to 2008 in the dot1q header of packets ingressing on VLAN 201.

```
switch(config)# interface ethernet 5
switch(config-if-Et5)# switchport vlan translation in 201 2008
switch(config-if-Et5)#
```

* These commands translate multiple VLAN mappings under an interface.

```
switch(config)#interface ethernet 5
switch(config-if-Et5)# switchport vlan translation 50 60
switch(config-if-Et5)# switchport vlan translation 61 71
switch(config-if-Et5)# switchport vlan translation 62 72
switch(config-if-Et5)#
```

# trunk group

The **trunk group** command assigns the configuration mode VLAN to a specified trunk group.

A trunk group is the set of physical interfaces that comprise the trunk and the collection of VLANs whose traffic is carried on the trunk. The traffic of a VLAN that belongs to one or more trunk groups is carried only on ports that are members of trunk groups to which the VLAN belongs. Switchport commands specify the physical interfaces that carry trunk group traffic.

The **no trunk group** and **default trunk group** commands remove the configuration mode VLAN from the specified trunk group by removing the corresponding **trunk group** statement from *running-config*. If a trunk group is not specified, the commands remove the configuration mode VLAN from all trunk groups.

## Command Mode

VLAN Configuration

## Command Syntax

```
trunk group name
no trunk group [name]
default trunk group [name]
```

## Parameters

- *name*    a name representing the trunk group.

## Example

- These commands assigns VLAN 49 to the trunk group *mlagpeer*:

```
switch(config)#vlan 49
switch(config-vlan-49)#trunk group mlagpeer
switch(config-vlan-49)#
```

## vlan

The **vlan** command places the switch in VLAN configuration mode to configure a set of virtual LANs. The command creates the specified VLANs if they do not exist prior to issuing the command. A VLAN that is in use as an internal VLAN may not be created or configured. The switch rejects any **vlan** command that specifies an internal VLAN ID.

The **default vlan** and **no vlan** commands removes the VLAN statements from *running-config* for the specified VLANs.

The **exit** command returns the switch to global configuration mode.

### Command Mode

Global Configuration

### Command Syntax

```
vlan vlan_range
no vlan vlan_range
default vlan vlan_range
```

### Parameters

- *vlan_range*    VLAN list.

  Formats include a name, number, number range, or comma-delimited list of numbers and ranges.

### Commands Available in VLAN configuration mode

- **name (VLAN configuration mode)**
- **state**
- **trunk group**

### Guidelines

In MLAG configurations, VLANs operate as follows:

- The VLAN must be configured identically on both MLAG peer switches.
- The port-specific bridging configuration originates on the switch where the port is physically located. This configuration includes the switchport access VLAN, switchport mode (trunk or access), trunk-allowed VLANs, the trunk native VLAN, and the switchport trunk groups.

### Example

- This command creates VLAN 49 and enters VLAN configuration mode for the new VLAN:

```
switch(config)#vlan 49
switch(config-vlan-49)#
```

# vlan internal order

The **vlan internal order** command specifies the range that the switch can allocate as internal VLANs when configuring routed ports and the order of their allocation. By default, the switch allocates VLANs in ascending order from VLAN 1006 to VLAN 4094.

The **no vlan internal order** and **default vlan internal order** commands revert the policy to its default.

## Command Mode

Global Configuration

## Command Syntax

```
vlan internal order DIRECTION [RANGE_VLAN]
no vlan internal order
default vlan internal order
```

## Parameters

- *DIRECTION*    VLAN allocation number direction. Options include:

    - *ascending*    allocates internal VLANs from lower VLAN bound to upper VLAN bound.

    - *descending*    allocates internal VLAN from upper VLAN bound to lower VLAN bound.

- *RANGE_VLAN*    allocation range. Options include:

    - <no parameter>    1006 (lower bound) to 4094 (upper bound).

    - *range lower upper*    specifies lower bound (*lower*) and upper bound (*upper*).

## Examples

- This command configures the switch to allocate internal VLANS from 3000 through 3999.

```
switch(config)#vlan internal order ascending range 3000 3999
switch(config)#
```

- This command configures the switch to allocate internal VLANS from 4094 through 1006.

```
switch(config)#vlan internal order descending
switch(config)#
```

- This command configures the switch to allocate internal VLANS from 4094 down through 4000.

```
switch(config)#vlan internal order descending range 4000 4094
switch(config)#
```

- This command reverts the allocation policy to its default (*ascending*, between *1006* and *4094*).

```
switch(config)#no vlan internal order
switch(config)#
```