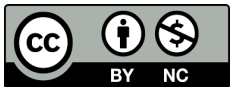
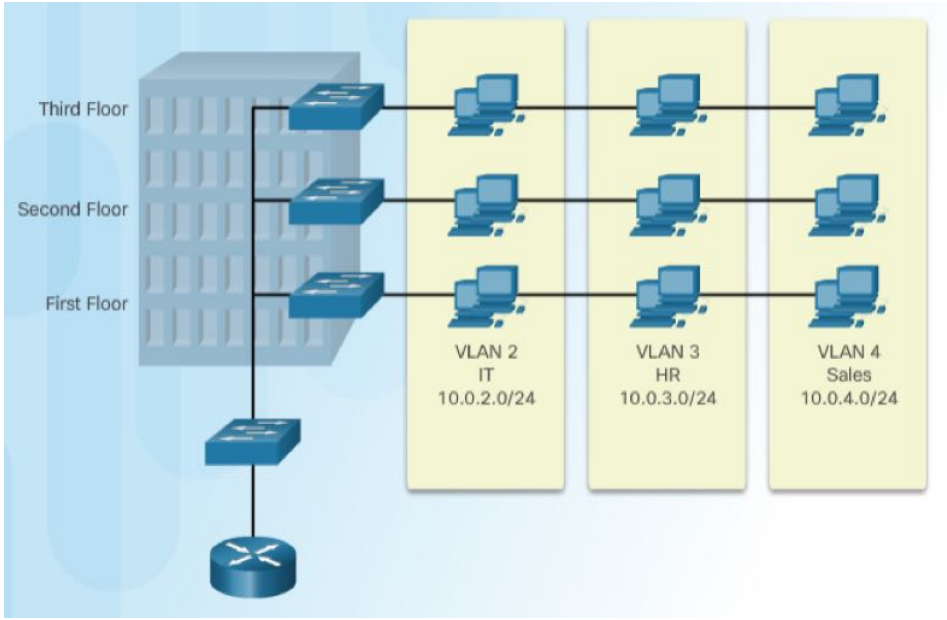
# Virtual Local Area Network (VLAN)

awal.ece@gmail.com

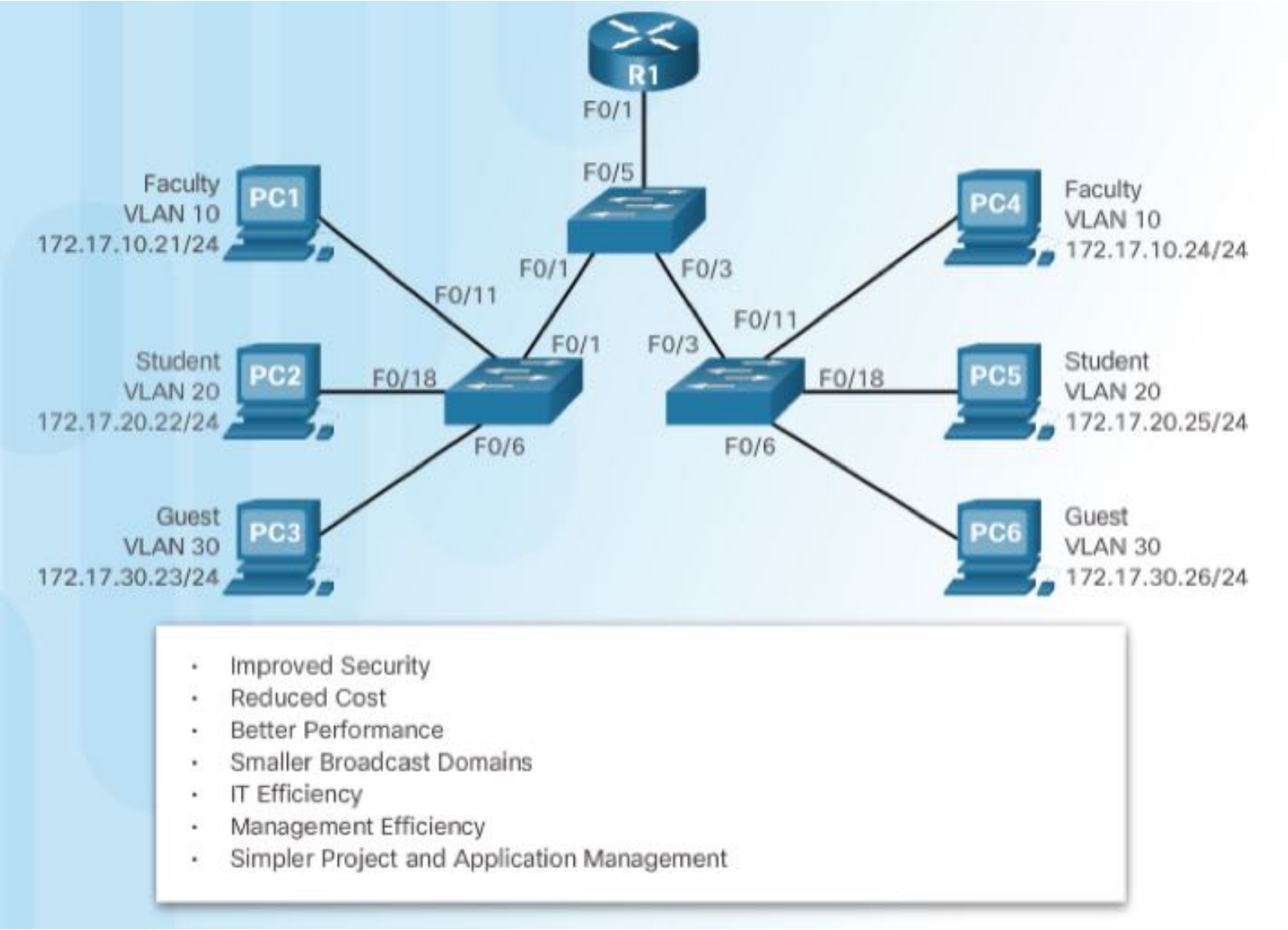


# VLAN

- VLANs can segment LAN devices without regard for the physical location of the user or device.
  - In the figure, IT users on the first, second, and third floors are all on the same LAN segment. The same is true for HR and Sales users.
- A VLAN is a logical partition of a Layer 2 network.
  - Multiple partitions can be created and multiple VLANs can co-exist.
  - The partitioning of the Layer 2 network takes place inside a Layer 2 device, usually via a switch.
  - Each VLAN is a broadcast domain that can span multiple physical LAN segments.
  - Hosts on the same VLAN are unaware of the VLAN's existence.
- 1 VLAN = 1 IP subnet.
- VLANs are mutually isolated and packets can only pass between VLANs via a router.



# Benefits of VLAN

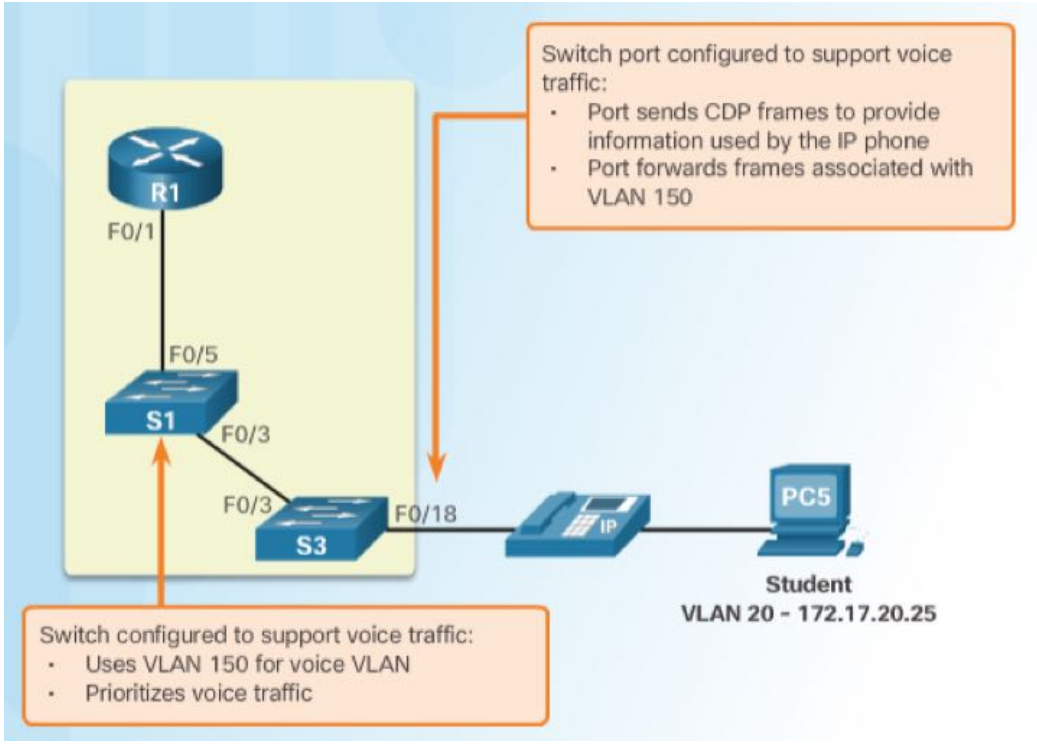


# Types of VLAN

- **Default VLAN** – Also known as VLAN 1. All switch ports are members of VLAN 1 by default.
- **Data VLAN** – Data VLANs are commonly created for specific groups of users or devices. They carry user generated traffic.
- **Native VLAN** – This is the VLAN that carries all untagged traffic. This is traffic that does not originate from a VLAN port (e.g., STP BPDU traffic exchanged between STP enabled switches). The native VLAN is VLAN 1 by default.
- **Management VLAN** – This is a VLAN that is created to carry network management traffic including SSH, SNMP, Syslog, and more. VLAN 1 is the default VLAN used for network management.

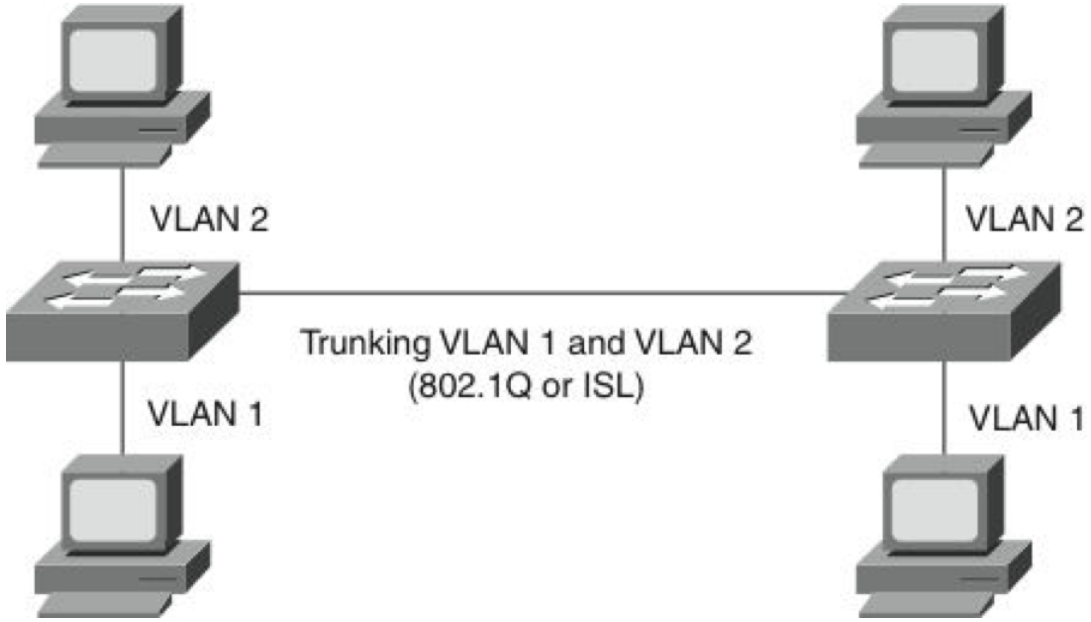
# Voice VLAN

- The voice VLAN feature enables access ports to carry user and IP voice traffic.
  - In the figure, the S3 F0/18 interface has been configured to tag student traffic on VLAN 20 and voice traffic on VLAN 150.



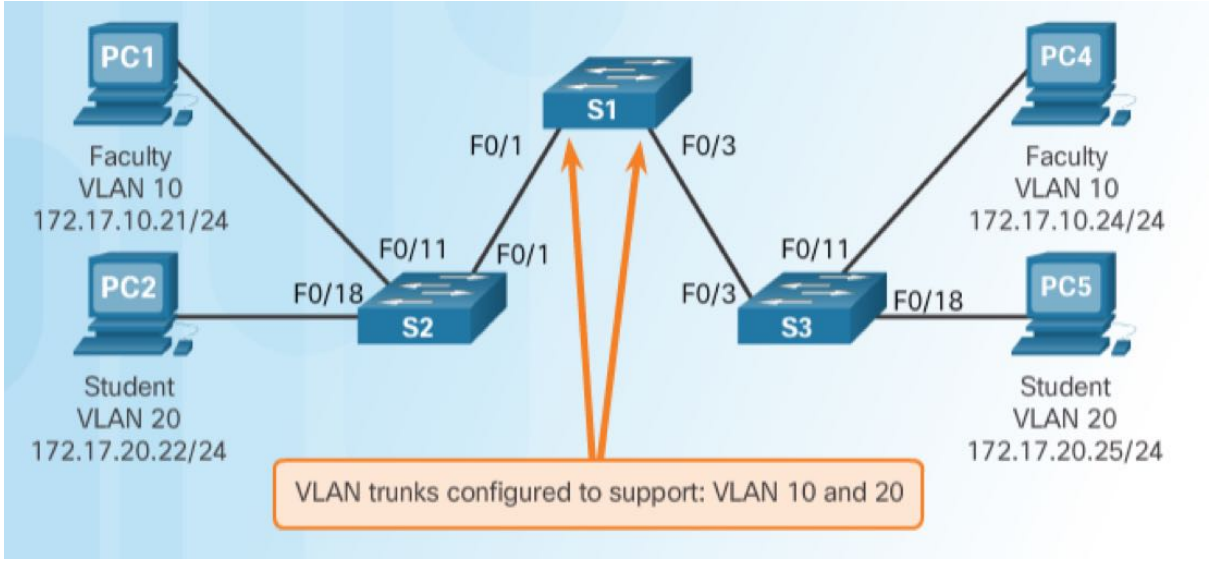
# VLAN Trunks

- A VLAN trunk is a point-to-point link that carries more than one VLAN.
  - Usually established between switches to support intra VLAN communication.
  - A VLAN trunk or trunk ports are not associated to any VLANs.



# Controlling Broadcast Domains with VLANs

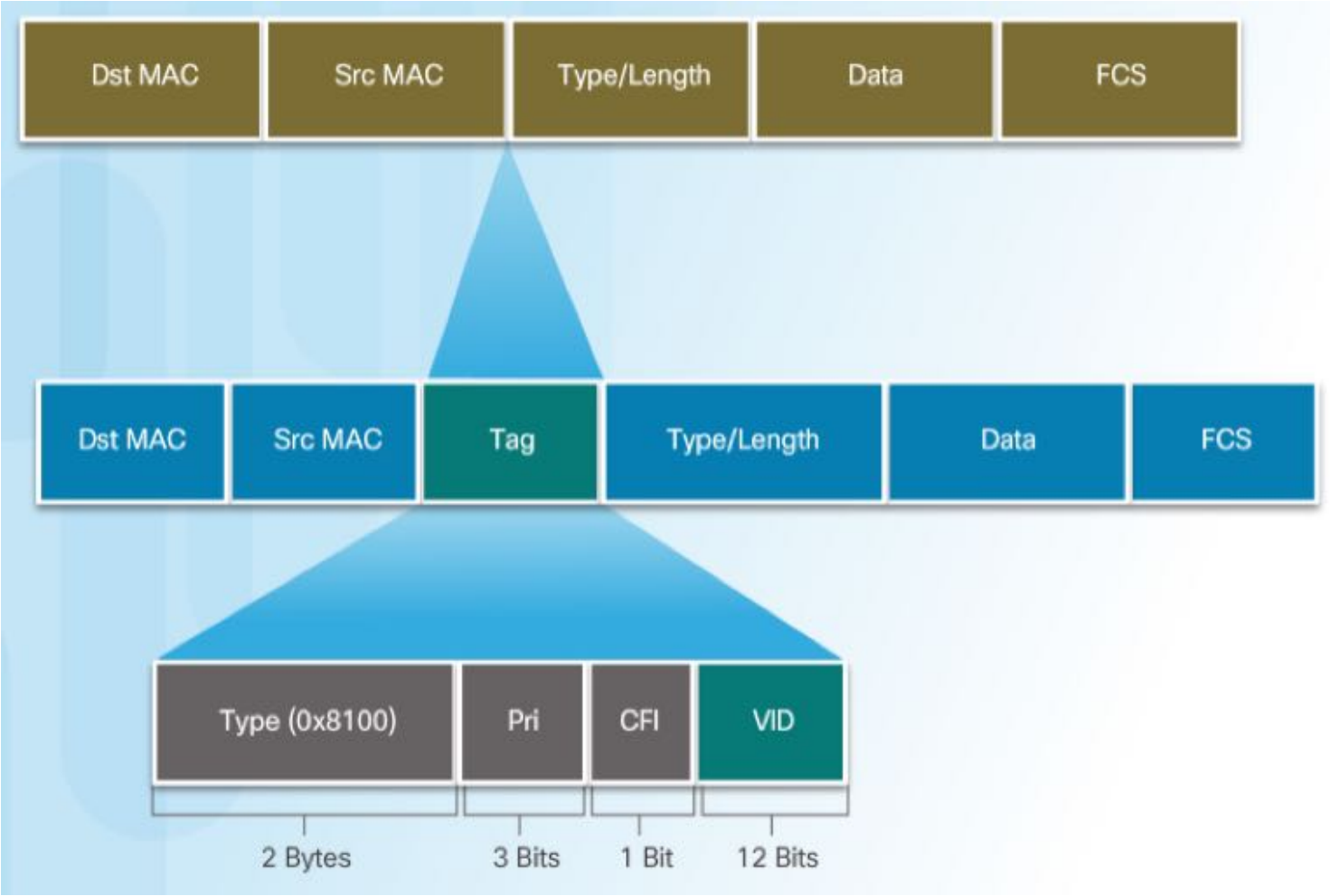
- If a switch port receives a broadcast frame, it forwards it out all ports except the originating port.
  - Eventually the entire network receives the broadcast because the network is one broadcast domain.
- VLANs can be used to limit the reach of broadcast frames because each VLAN is a separate broadcast domain.
  - VLANs help control the broadcast frames and their impact in a network.



# VLAN Tagging in Ethernet Frames

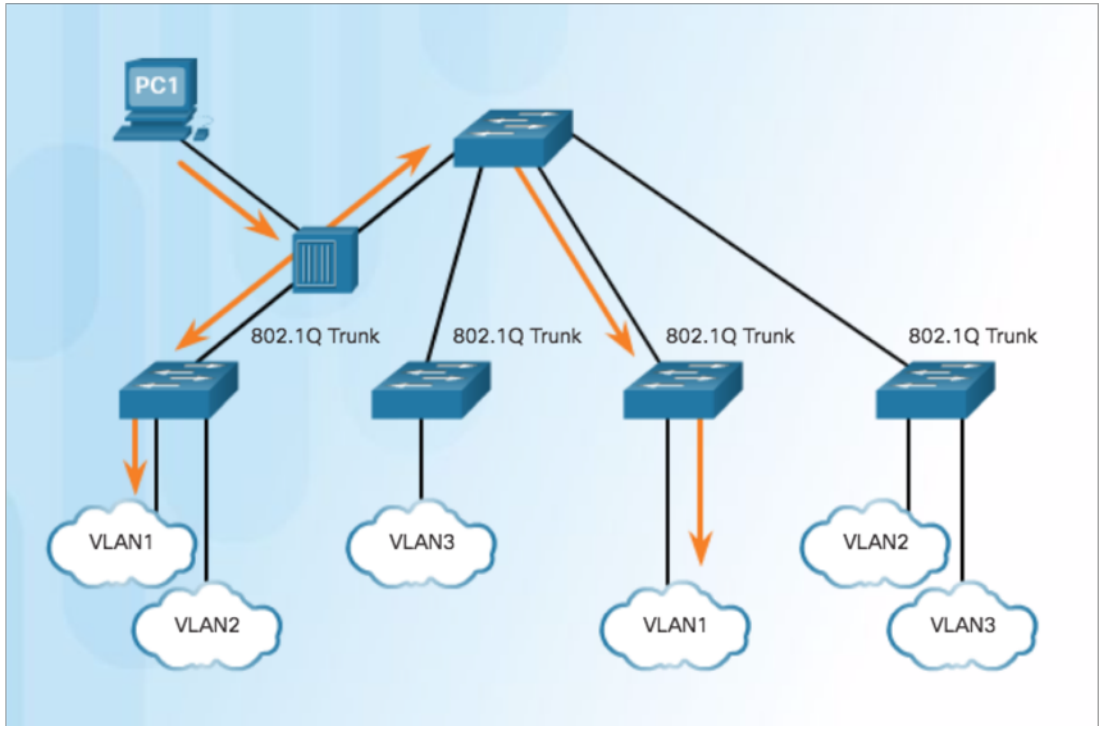
- Before a frame is forwarded across a trunk link, it must be tagged with its VLAN information.
  - Frame tagging is the process of adding a VLAN identification header to the frame.
  - It is used to properly transmit multiple VLAN frames through a trunk link.
- IEEE 802.1Q is a vey popular VLAN trunking protocol that defines the structure of the tagging header added to the frame.
  - Switches add VLAN tagging information after the Source MAC address field.
  - The fields in the 802.1Q VLAN tag includes VLAN ID (VID).
  - Trunk links add the tag information before sending the frame and then remove the tags before forwarding frames through non-trunk ports.

# VLAN Tagging in Ethernet Frames



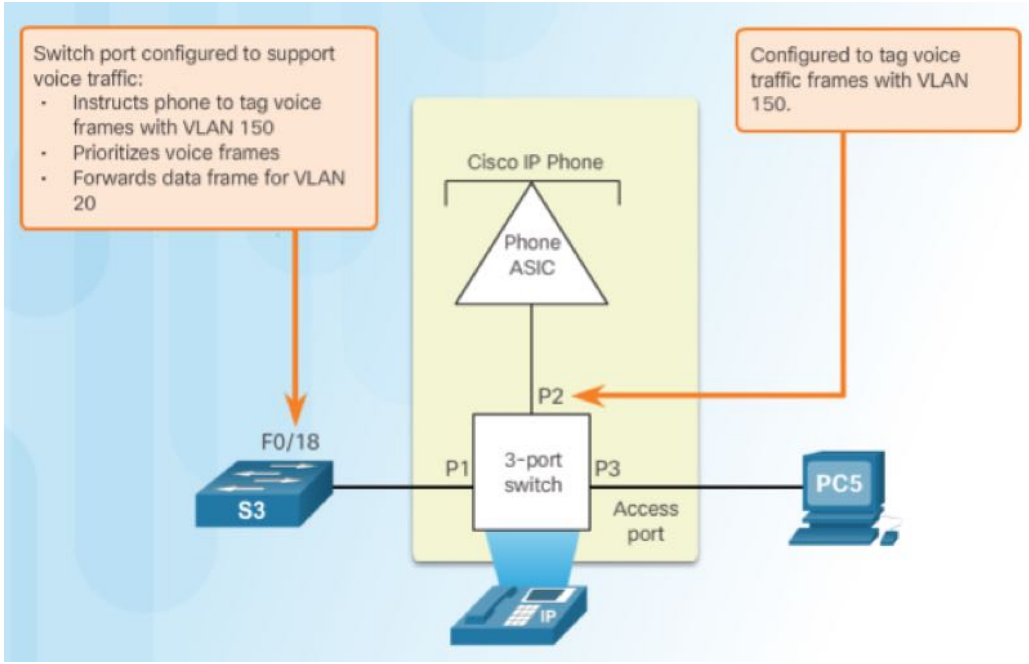
# Native VLANs and 802.1Q Tagging

- Control traffic sent on the native VLAN should not be tagged.
- Frames received untagged, remain untagged and are placed in the native VLAN when forwarded.
- If there are no ports associated to the native VLAN and no other trunk links, an untagged frame is dropped.
- When configuring a switch port on a Cisco switch, configure devices so that they do not send tagged frames on the native VLAN.



# Voice VLAN Tagging

- An access port connecting a Cisco IP phone can be configured to use two separate VLANs:
  - A VLAN for voice traffic
  - A VLAN for data traffic from a device attached to the phone.
- The link between the switch and the IP phone behaves like a trunk to carry traffic from both VLANs.



# VLAN Range

- Normal range VLANs
  - VLAN numbers from 1 to 1,005
  - Configurations stored in the vlan.dat (in the flash memory)
  - IDs 1002 through 1005 are reserved for legacy Token Ring and Fiber Distributed Data Interface (FDDI) VLANs, automatically created and cannot be removed.
- Extended Range VLANs
  - VLAN numbers from 1,006 to 4,096
  - Configurations stored in the running configuration (NVRAM)
  - VLAN Trunking Protocol (VTP) does not learn extended VLANs

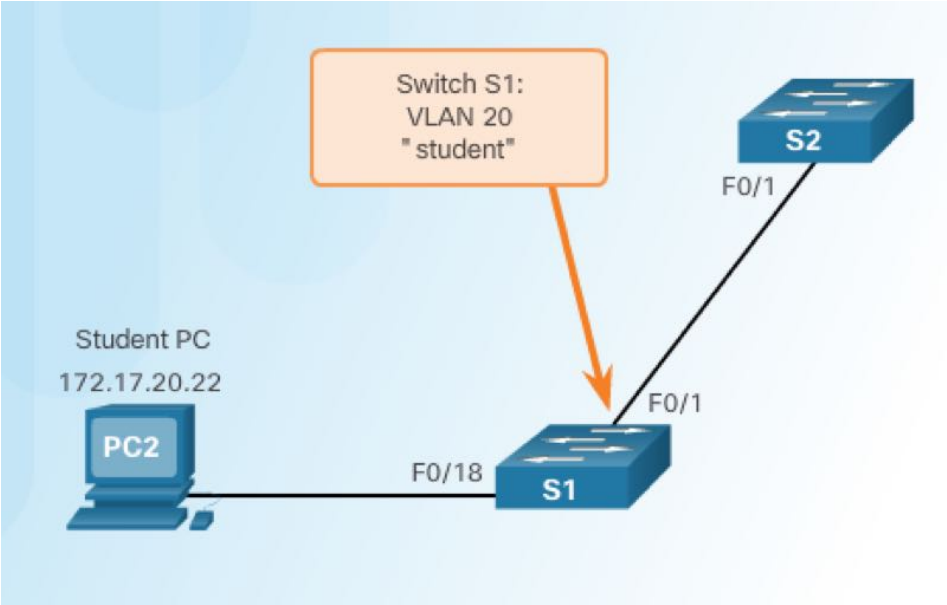
```
Switch# show vlan brief

VLAN Name                Status    Ports
---- ------------------- --------- -------------------------------
1    default             active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                   Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                   Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                   Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                   Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                   Fa0/21, Fa0/22, Fa0/23, Fa0/24
                                   Gi0/1, Gi0/2
1002 fddi-default        act/unsup
1003 token-ring-default  act/unsup
1004 fddinet-default     act/unsup
1005 trnet-default       act/unsup
```

# Configuring VLAN

# Creating a VLAN

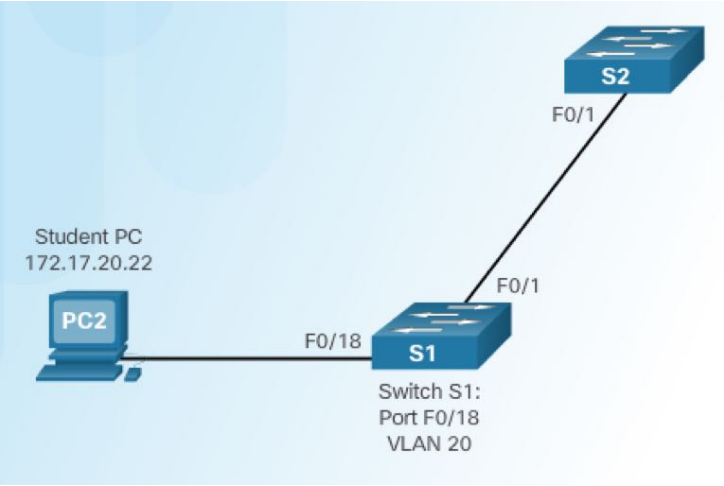
| | |
|---|---|
| Enter global configuration mode. | `S1# configure terminal` |
| Create a VLAN with a valid id number. | `S1(config)# vlan vlan-id` |
| Specify a unique name to identify the VLAN. | `S1(config-vlan)# name vlan-name` |
| Return to the privileged EXEC mode. | `S1(config-vlan)# end` |



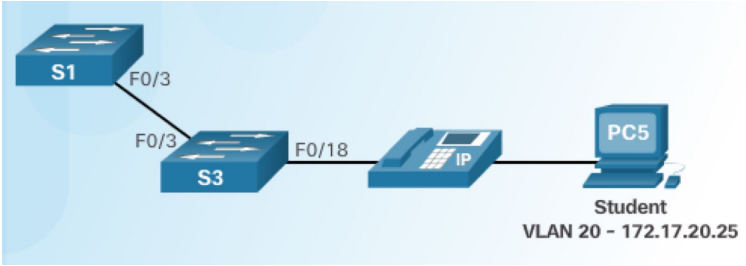

```
S1# configure terminal
S1(config)# vlan 20
S1(config-vlan)# name student
S1(config-vlan)# end
```

# Assigning Ports to VLANs

| | |
|---|---|
| Enter global configuration mode. | `S1# configure terminal` |
| Enter interface configuration mode. | `S1(config)# interface` *interface_id* |
| Set the port to access mode. | `S1(config-if)# switchport mode access` |
| Assign the port to a VLAN. | `S1(config-if)# switchport access vlan` *vlan_id* |
| Return to the privileged EXEC mode. | `S1(config-if)# end` |



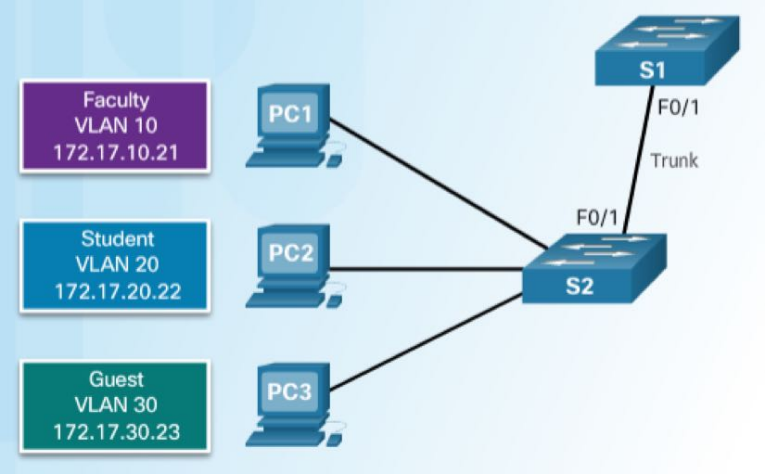

```
S1# configure terminal
S1(config)# interface F0/18
S1(config-if)# switchport mode access
S1(config-if)# switchport access vlan 20
S1(config-if)# end
```




```
S3(config)# vlan 20
S3(config-vlan)# name student
S3(config-vlan)# vlan 150
S3(config-vlan)# name VOICE
S3(config-vlan)# exit
S3(config)#
S3(config)# interface fa0/18
S3(config-if)# switchport mode access
S3(config-if)# switchport access vlan 20
S3(config-if)#
S3(config-if)# mls qos trust cos
S3(config-if)# switchport voice vlan 150
S3(config-if)# end
S3#
```

# Configuring IEEE 802.1q Trunk Links

| | |
|---|---|
| Enter global configuration mode. | `S1# configure terminal` |
| Enter interface configuration mode. | `S1(config)# interface` *interface_id* |
| Force the link to be a trunk link. | `S1(config-if)# switchport mode trunk` |
| Specify a native VLAN for untagged frames. | `S1(config-if)# switchport trunk native vlan` *vlan_id* |
| Specify the list of VLANs to be allowed on the trunk link. | `S1(config-if)# switchport trunk allowed vlan` *vlan-list* |
| Return to the privileged EXEC mode. | `S1(config-if)# end` |




```
S1(config)# interface FastEthernet0/1
S1(config-if)# switchport mode trunk
S1(config-if)# switchport trunk native vlan 99
S1(config-if)# switchport trunk allowed vlan 10,20,30,99
S1(config-if)# end
```

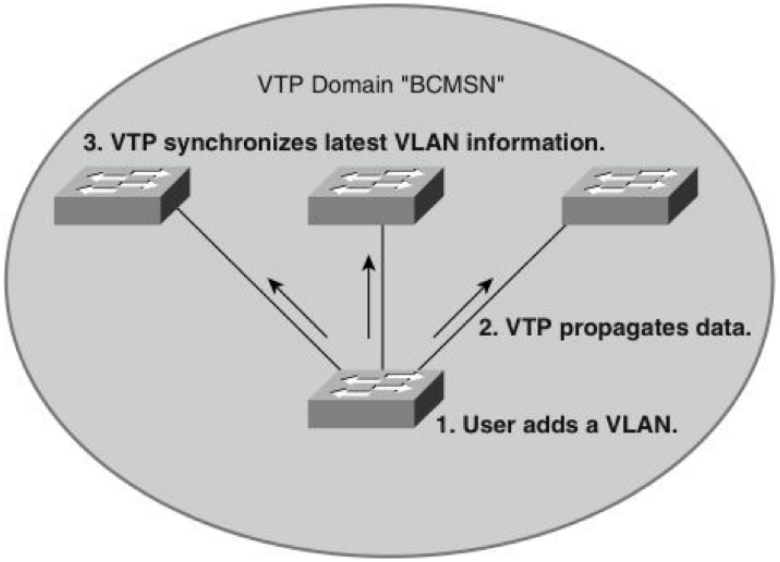
# Verifying Trunk Configuration

```
S1# show interfaces f0/1 switchport
Name: Fa0/1
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 99 (VLAN0099)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
<output omitted>
```

# VLAN Trunking Protocol (VTP)

# VTP Overview

- VTP is a Cisco-proprietary protocol that automates the propagation of VLAN information between switches via trunk links. This minimizes misconfigurations and configuration inconsistencies.
- VTP does not configure switch ports for VLAN membership.



# VTP Modes

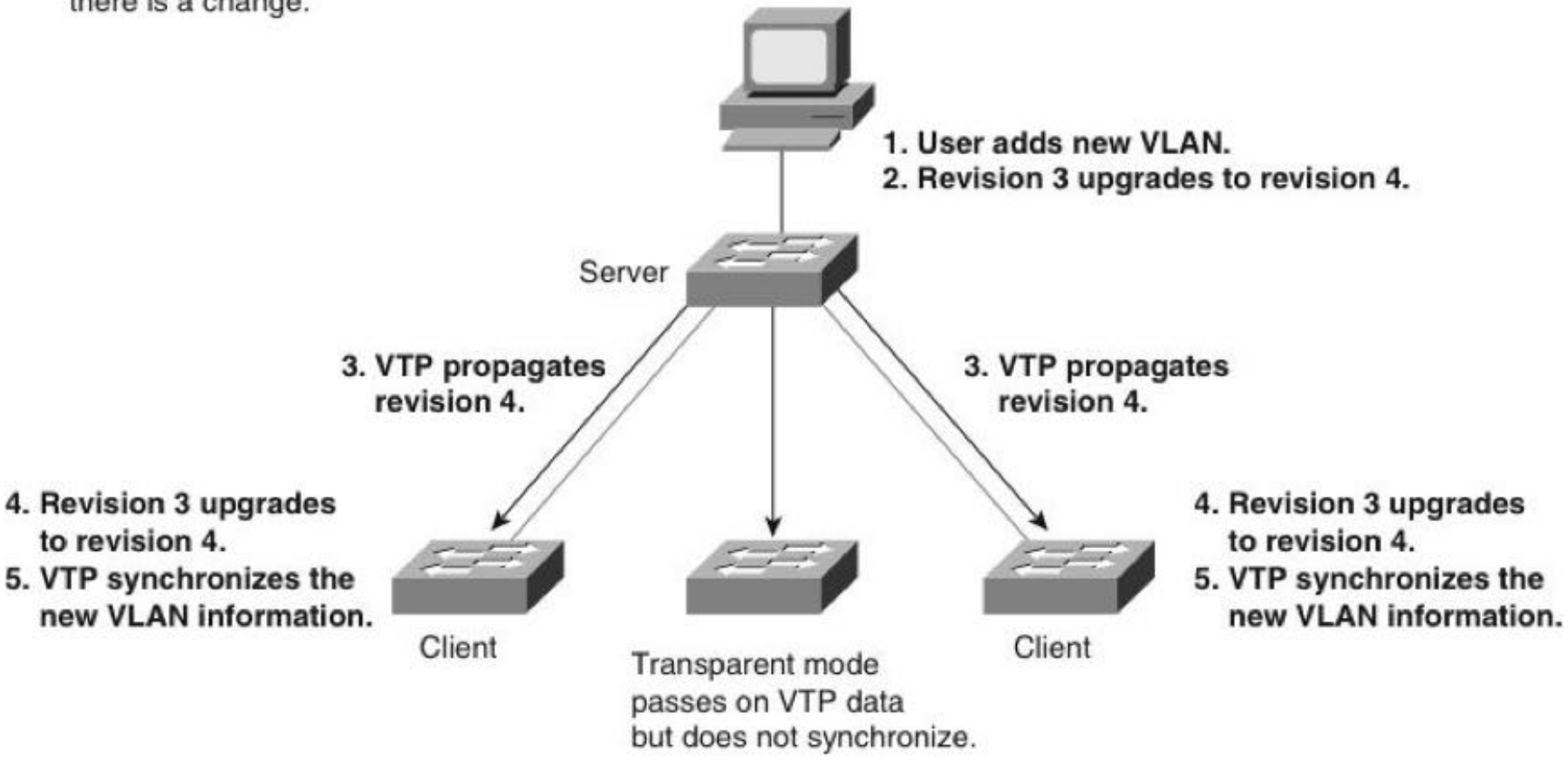
| Mode | Description |
|---|---|
| **Client** | • Cannot create, change, or delete VLANs on command-line interface (CLI).<br>• Forwards advertisements to other switches.<br>• Synchronizes VLAN configuration with latest information received from other switches in the management domain.<br>• Does not save VLAN configuration in nonvolatile RAM (NVRAM). |
| **Server** | • Can create, modify, and delete VLANs.<br>• Sends and forwards advertisements to other switches.<br>• Synchronizes VLAN configuration with latest information received from other switches in the management domain.<br>• Saves VLAN configuration in NVRAM. |
| **Transparent** | • Can create, modify, and delete VLANs only on the local switch.<br>• Forwards VTP advertisements received from other switches in the same management domain.<br>• Does not synchronize its VLAN configuration with information received from other switches in the management domain.<br>• Saves VLAN configuration in NVRAM. |

# VTP Modes

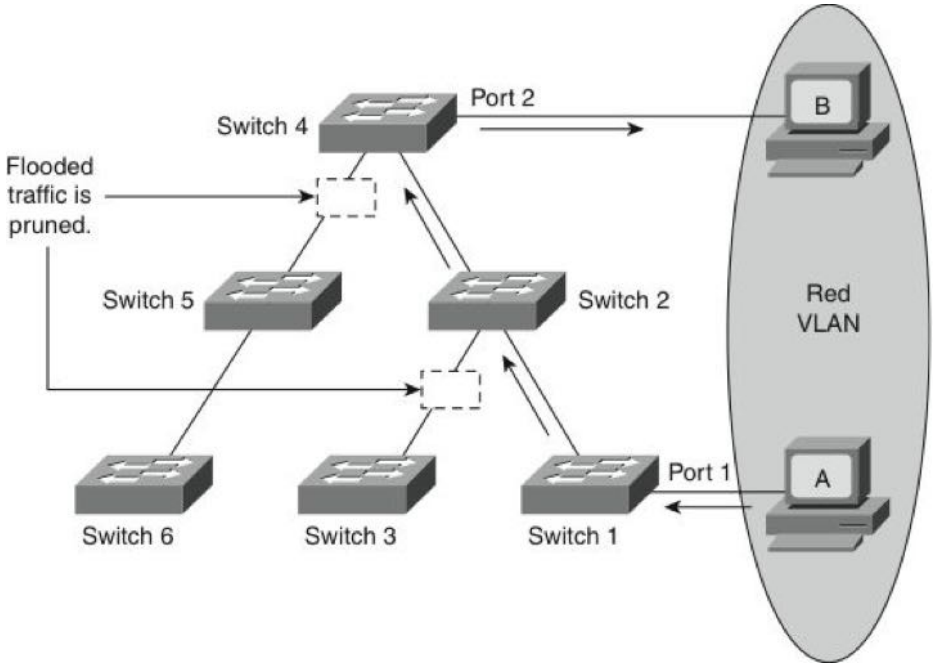
| VTP Question | VTP Server | VTP Client | VTP Transparent |
|---|---|---|---|
| What are the differences? | • Manages domain and VLAN configuration.<br>• Multiple VTP servers can be configured. | • Updates local VTP configurations.<br>• VTP client switches cannot change VLAN configurations. | • Manages local VLAN configurations.<br>• VLAN configurations are not shared with VTP network. |
| Does it respond to VTP advertisements? | Participates fully | Participates fully | Only forwards VTP advertisements |
| Is the global VLAN configuration preserved on restart? | Yes, global configurations are stored in NVRAM | No, global configurations are stored in RAM only | No, local VLAN configuration is only stored in NVRAM |
| Does it update other VTP-enabled switches? | Yes | Yes | No |

# VTP Operation

VTP advertisements are sent as multicast frames.
VTP servers and clients are synchronized to the latest revision number.
VTP advertisements are sent every 5 minutes or when there is a change.



# VTP Pruning



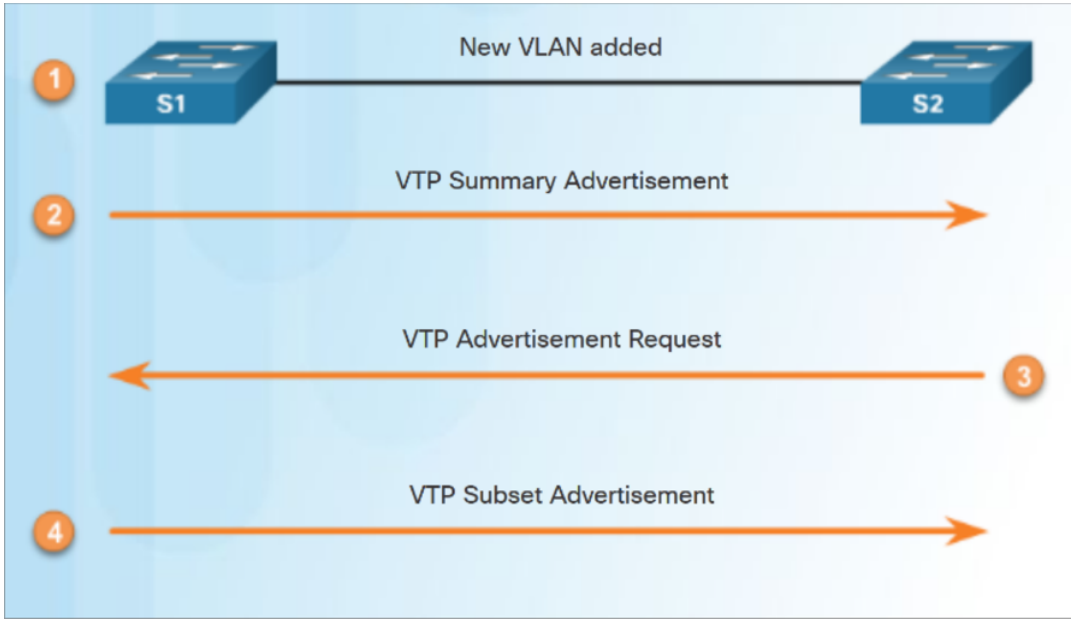

- VTP pruning prevents flooded traffic from propagating to switches that do not have members in specific VLANs.
- VTP pruning uses VLAN advertisements to determine when a trunk connection is flooding traffic needlessly. Switches 1 and 4 in the figure support ports statically configured in the Red VLAN.
- The broadcast traffic from Station A is not forwarded to Switches 3, 5, and 6 because traffic for the Red VLAN has been pruned on the links indicated on Switches 2 and 4.

# VTP Versions

| VTP Versions | Definition |
|---|---|
| Version 1 | • Default VTP mode on all switches<br>• Supports normal VLAN range only |
| Version 2 | • Supports normal VLAN range only<br>• Supports legacy Token Ring networks<br>• Not interoperable with Version 1<br>• Supports advance features including version dependent transparent mode, VLAN consistency check and unrecognized TLV |
| Version 3 | • Support for extended VLANs (1025 to 4094)<br>• Support for the creation and advertising of Private VLANs<br>• Enhancements to a mechanism for protection from the “wrong” database accidentally being inserted into a VTP domain<br>• Interaction with VTP versions 1 and 2 |

# VTP Advertisements

- Three types of VTP Advertisements:
- **Summary advertisements** – contain VTP domain name and configuration revision number.
- **Advertisement request** - response to a summary advertisement message when the summary advertisement contains a higher configuration revision number than the current value.
- **Subset advertisements** - contain VLAN information including any changes.



# Configuring VTP

- **Step 1.** Enter global configuration mode:

```
Switch# configure terminal
```

- **Step 2.** Configure the VTP mode as server:

```
Switch(config)# vtp mode server
```

- **Step 3.** Configure the domain name:

```
Switch(config)# vtp domain domain_name
```

- **Step 4.** (Optional.) Enable VTP version 2:

```
Switch(config)# vtp version 2
```

- **Step 5.** (Optional.) Specify a VTP password:

```
Switch(config)# vtp password password_string
```

- **Step 6.** (Optional.) Enable VTP pruning in the management domain:

```
Switch(config)# vtp pruning
```

# VTP Configuration Example

```
Switch# configure terminal
Switch(config)# vtp mode server
Setting device to VTP SERVER mode.
Switch(config)# vtp domain VTPLAB
Switch(config)# vtp version 2
Switch(config)# vtp password lab123
Switch(config)# vtp pruning
Switch(config)# end
```

```
Switch# show vtp status
VTP Version : 2
Configuration Revision : 247
Maximum VLANs supported locally : 1005
Number of existing VLANs : 10
VTP Operating Mode : Server
VTP Domain Name : VTPLAB
VTP Pruning Mode : Enabled
VTP V2 Mode : Enabled
VTP Traps Generation : Disabled
MD5 digest : 0x45 0x52 0xB6 0xFD 0x63 0xC8 0x49 0x80
Configuration last modified by 0.0.0.0 at 8-12-99 15:04:4
```

# Dynamic Trunking Protocol (DTP)

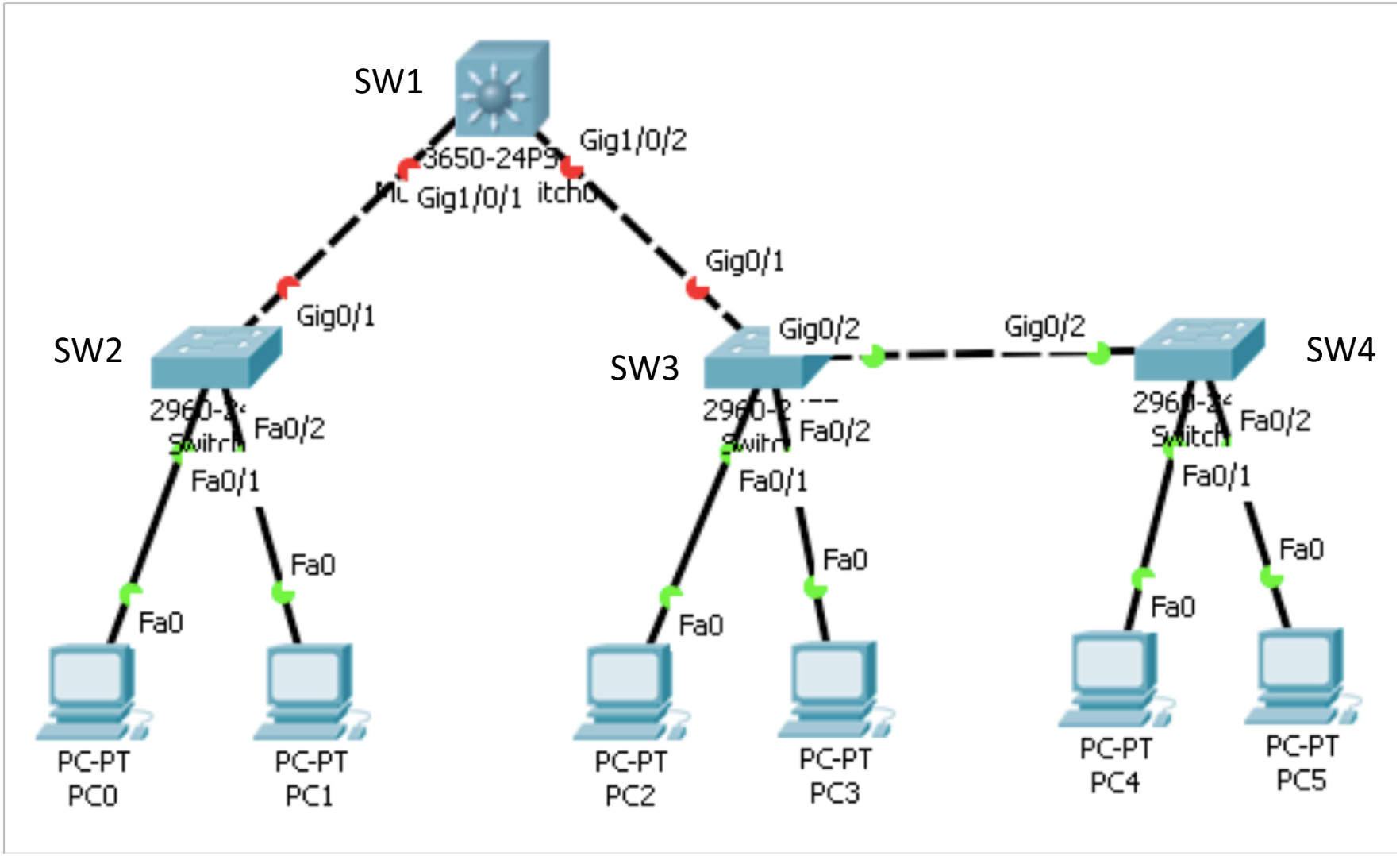
# DTP Overview

| | Dynamic Auto | Dynamic Desirable | Trunk | Access |
|---|---|---|---|---|
| Dynamic Auto | Access | Trunk | Trunk | Access |
| Dynamic Desirable | Trunk | Trunk | Trunk | Access |
| Trunk | Trunk | Trunk | Trunk | Limited connectivity |
| Access | Access | Access | Limited connectivity | Access |

- **Switchport mode access** - interface becomes a nontrunk interface.
- **Switchport mode dynamic auto** - interface becomes a trunk if the neighboring interface is set to trunk or desirable mode.
- **Switchport mode dynamic desirable** - interface becomes a trunk if the neighboring interface is set to trunk, desirable, or dynamic auto mode.
- **Switchport mode trunk** - interface becomes a trunk even if the neighboring interface is not a trunk interface.
- **Switchport nonegotiate** - prevents the interface from generating DTP frames.

# VLAN, VTP and DTP Lab

# Lab Topology



# IP Plan

- VLAN:
  - VLAN 10: PC0, PC2 and PC4
  - VLAN 20: PC1, PC3 and PC5
- Prefix:
  - VLAN 10: 10.0.10.0/24
  - VLAN 20: 10.0.20.0/24
- IP Address:
  - PC0: 10.0.10.1, PC2: 10.0.10.2, PC4: 10.0.10.3
  - PC1: 10.0.20.1, PC3: 10.0.20.2, PC5: 10.0.20.3
- GW:
  - VLAN 10: 10.0.10.254
  - VLAN 20: 10.0.20.254

# Task 1: DTP

## Configure Trunk Ports

# Task 1: Configure Trunk Ports

- Using DTP: Configure SW1 (Trunk) and SW2 (Dynamic Auto)

  Example:

  ```
  SW1(config-if)# switchport trunk encapsulation dot1q
  SW1(config-if)# switchport mode trunk
  ```

- Without using DTP: Configure SW1 (Trunk) and SW3 (Trunk)

  Example:

  ```
  SW3(config-if)# switchport nonegotiate
  SW3(config-if)# switchport mode trunk
  ```

- Using DTP: Configure SW3 (Dynamic Auto) and SW4 (Dynamic Desirable)

  Example:

  ```
  SW4(config-if)# switchport mode dynamic desirable
  ```

**Verify:**

```
Switch# show interface trunk
Switch# show interface <ID> switchport
```

# Task 2: VTP

Configure VTP

# Task 2: Configure VTP

- Configure SW1 (Server), SW2 (Client), SW3 (Transparent) and SW4 (Client)

  Example:

```
SW1(config)# vtp mode server
SW1(config)# vtp domain VTPLAB
SW1(config)# vtp password lab123
SW1(config)# vtp version 2
```

- Configure VLAN in VTP Server and Transparent Switch

  Example:

```
SW1(config)# vlan 10
SW1(config-vlan)# name STAFF
SW1(config)# vlan 20
SW1(config-vlan)# name GUEST
```

**Verify:**

```
Switch# show vlan brief
Switch# show vtp status
```

# Task 3: VLAN

## Assign Ports to VLANs

# Task 3: Configure Ports to VLANs

- Configure Access Switches to assign ports with specific VLANs

  Example:

```
SW2(config)# int fa0/1
SW2(config-if)# switchport mode access
SW2(config-if)# switchport access vlan 10
SW2(config)# int fa0/2
SW2(config-if)# switchport mode access
SW2(config-if)# switchport access vlan 20
```

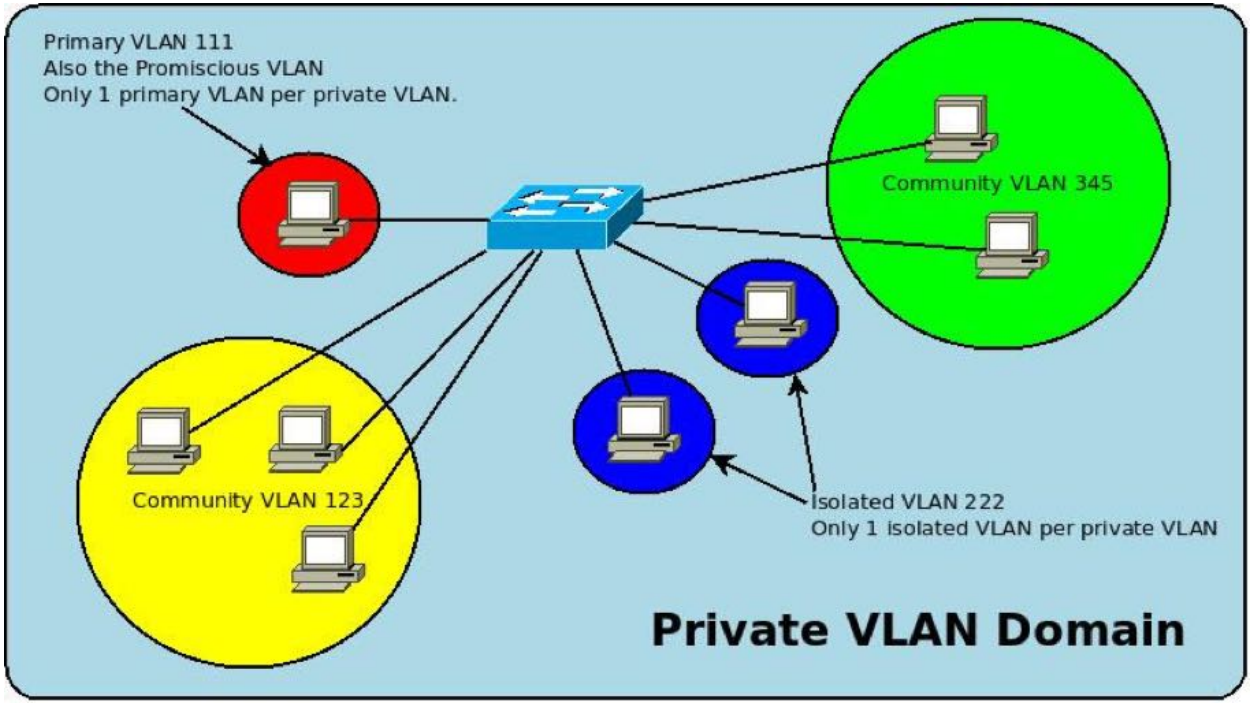
**Verify:**

```
Switch# show vlan brief
```

# Private VLANs

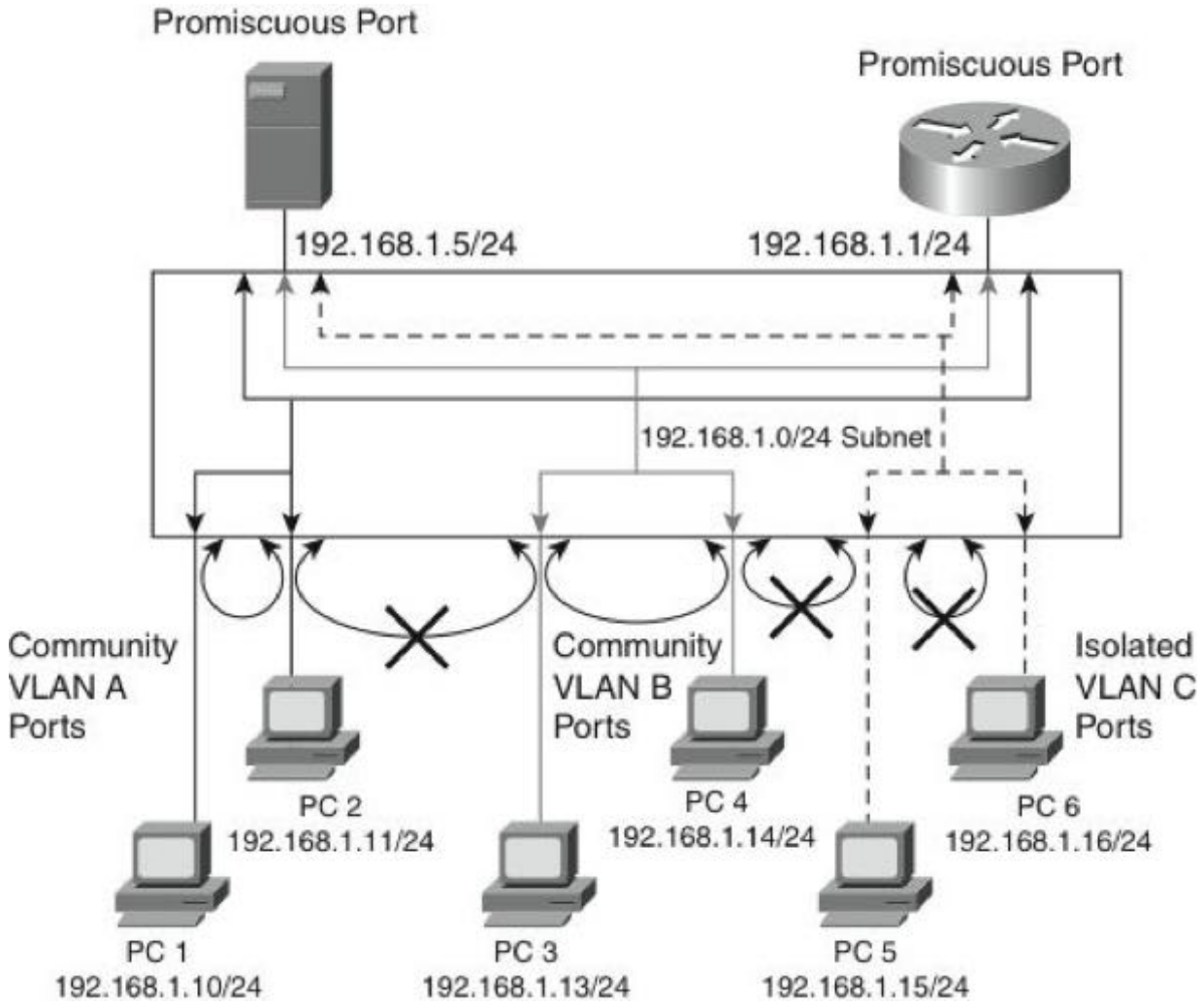
# Motivation for Private VLANs

- Service providers often have devices from multiple clients, in addition to their own servers, in a single Demilitarized Zone (DMZ) segment or VLAN. As security issues abound, it becomes more important to provide traffic isolation between devices, even though they might exist on the same Layer 3 segment and VLAN.
- Most Cisco IOS-based switches implement private VLANs to keep some switch ports shared and some switch ports isolated, even though all ports remain in the same VLAN.



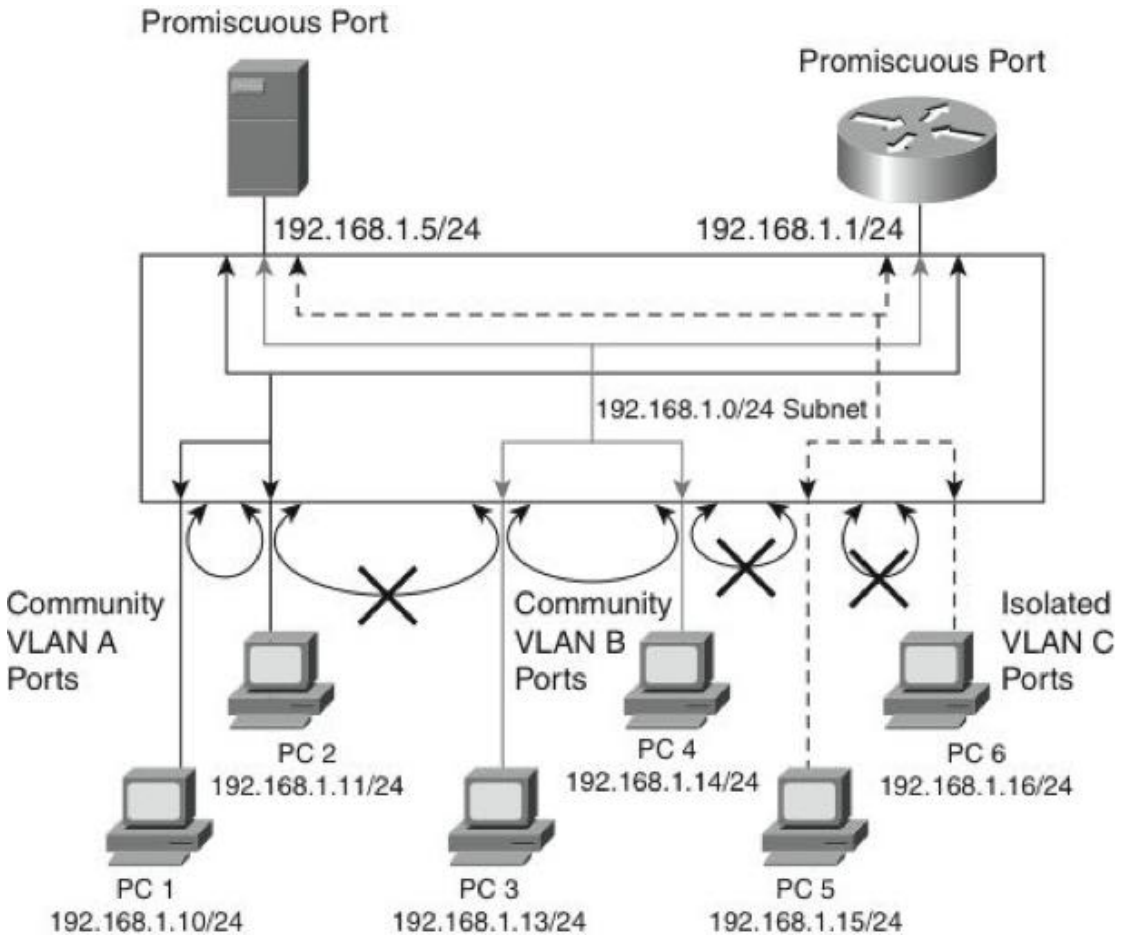
# pVLAN Port Types

- Isolated
- Promiscuous
- Community



# pVLAN Structure Supporting VLANs

- Primary Private VLAN
- Secondary Private VLAN
- Community Private VLAN
- Isolated Private VLAN



# Configuring pVLANs - Steps

- **Step 1.** Set VTP mode to transparent.
- **Step 2.** Create the secondary pVLANs.
- **Step 3.** Create the primary pVLAN.
- **Step 4.** Associate the secondary pVLAN with the primary pVLAN.
  - Only one isolated pVLAN can be mapped to a primary pVLAN, but more than one community pVLAN can be mapped to a primary pVLAN.
- **Step 5.** Configure an interface as an isolated or community port.
- **Step 6.** Associate the isolated port or community port with the primary-secondary pVLAN pair.
- **Step 7.** Configure an interface as a promiscuous port.
- **Step 8.** Map the promiscuous port to the primary-secondary pVLAN pair.

# Configuring pVLANs - Commands

```
Switch(config)# vlan pvlan-id

Switch(config-vlan)# private-vlan {community | isolated | primary}

Switch(config-vlan)# exit

Switch(config)# vlan primary-vlan-id

Switch(config-vlan)# private-vlan association {secondary-vlan-list | add
secondary-vlan-list | remove secondary-vlan-list}

Switch(config-vlan)# interface vlan primary-vlan-id

Switch(config-if)# private-vlan mapping {secondary-vlan-list | add secondary-
vlan-list | remove secondary-vlan-list}

Switch(config-if)# interface type slot/port

Switch(config-if)# switchport

Switch(config-if)# switchport mode private-vlan {host | promiscuous}

Switch(config-if)# switchport private-vlan host-association primary-vlan-id
secondary-vlan-id

Switch(config-if)# switchport private-vlan mapping primary-vlan-id {secondary-
vlan-list | add secondary-vlan-list | remove secondary-vlan-list}
```

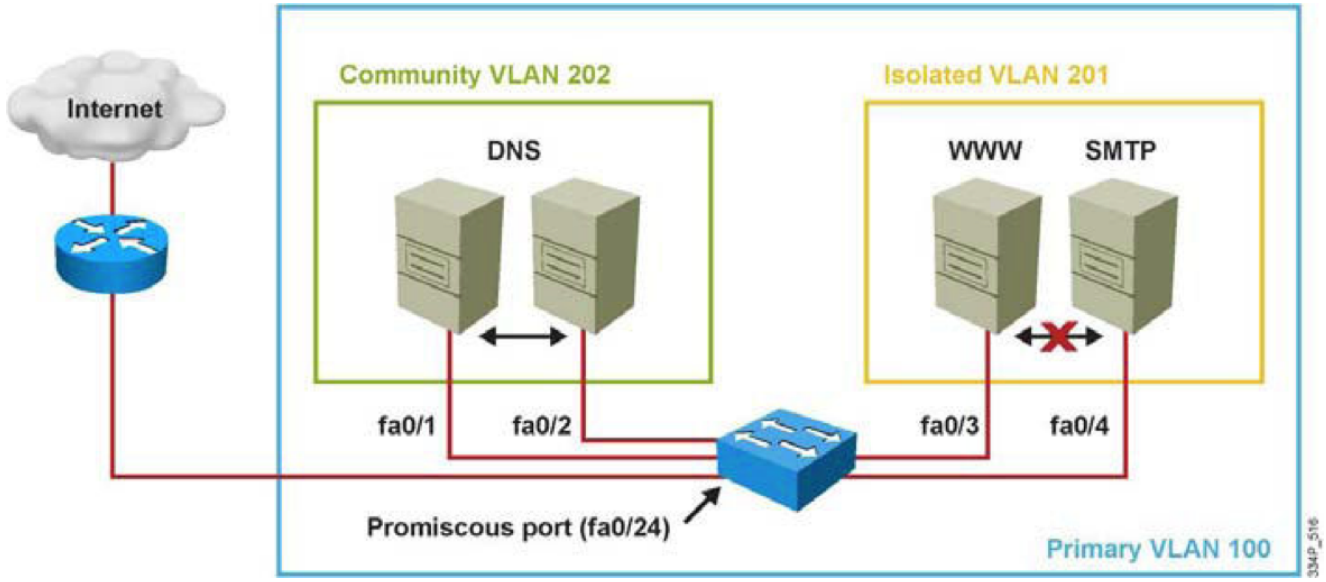
# Verifying pVLAN Configuration

- The two most useful commands for this purpose are **`show interface switchport`** and **`show vlan private-vlan`**.

```
Switch# show vlan private-vlan
Primary   Secondary           Type                    Interfaces
-------   ---------           --------------      -----------------
100       200       community
100       300       isolated

Switch# show interfaces FastEthernet 5/2 switchport
Name: Fa5/2
Switchport: Enabled
Administrative Mode: private-vlan host
Operational Mode: down
Administrative Trunking Encapsulation: negotiate
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative private-vlan host-association: 100 (VLAN0200) 300 (VLAN0300)
Administrative private-vlan mapping: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
```

# pVLAN Scenario 1: Single Switch

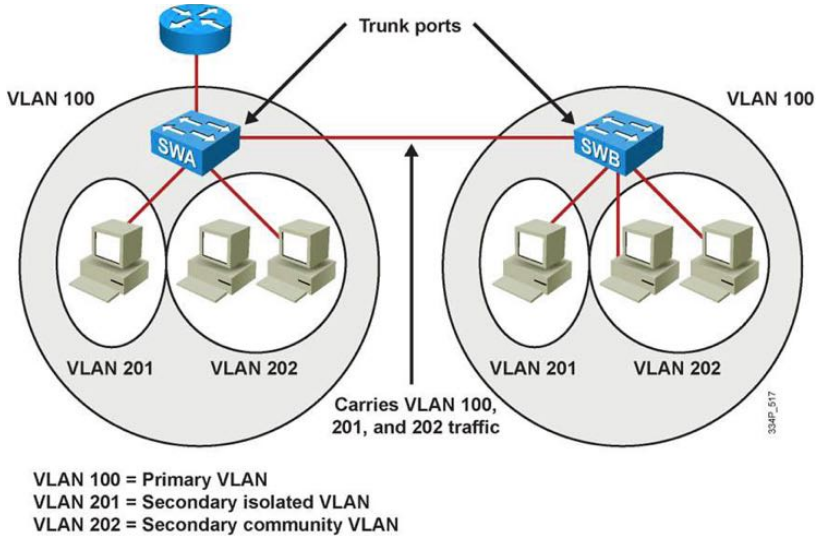



- A corporate DMZ contains two DNS servers, one web server and one SMTP server. All servers and their connecting router are in the same subnet.
- DNS servers are redundant copies, so they need to communicate with each other to update their entries and to the Internet. In addition to that, they also need to communicate with the Internet.
- The Web Server and the SMTP server need to communicate with the Internet, but for security purposes, the SMTP server should not be reachable from the Web or the DNS servers. The web server needs to be accessible from the Internet but not from the SMTP server.

# pVLAN Configuration for Scenario 1

```
Switch(config)# vtp transparent
Switch(config)# vlan 201
Switch(config-vlan)# private-vlan isolated
Switch(config)# vlan 202
Switch(config-vlan)# private-vlan community
Switch(config-vlan)# vlan 100
Switch(config-vlan)# private-vlan primary
Switch(config-vlan)# private-vlan association 201,202
Switch(config-vlan)# interface fastethernet 0/24
Switch(config-if)# switchport mode private-vlan promiscuous
Switch(config-if)# switchport private-vlan mapping 100 201,202
Switch(config-if)# interface range fastethernet 0/1 - 2
Switch(config-if)# switchport mode private-vlan host
Switch(config-if)# switchport private-vlan host-association 100 202
Switch(config-if)# interface range fastethernet 0/3 - 4
Switch(config-if)# switchport mode private-vlan host
Switch(config-if)# switchport private-vlan host-association 100 201
```

# pVLAN Scenario 2: Multiple Switches




- A trunk port carries the primary VLAN and secondary VLANs to a neighboring switch just like any other VLAN.
- A feature of pVLANs across multiple switches is that traffic from an isolated port in one switch does not reach an isolated port on another switch.
- Configure pVLANs on all switches on the path, which includes devices that have no pVLAN ports to maintain the security of your pVLAN configuration, and avoid using other VLANs configured as pVLANs.
- As shown in the figure, the switches SWA and SWB have the same pVLANs on two different switches and are connected through the trunk link.

# pVLAN Configuration for Scenario 2

- To configure a Layer 2 interface as a Private VLAN trunk port, use the interface command:

  `Switch(config-if)# **switchport private-vlan association trunk** *primary_vlan_ID secondary_vlan_ID*`

- If the port is set to promiscuous, use the **mapping** command:

  `Switch(config-if)# **switchport private-vlan mapping** *primary_vlan_ID secondary_vlan_list*`

- Once the trunk is configured, allow VLANs with the command

  `Switch(config-if)# **switchport private-vlan trunk allowed vlan** *vlan_list*`

- Configure the native VLAN with following command

  `Switch(config-if)# **switchport private-vlan trunk native vlan** *vlan_id*`

```
Switch(config)# interface fastethernet 5/2
Switch(config-if)# switchport mode private-vlan trunk secondary
Switch(config-if)# switchport private-vlan trunk native vlan 10
Switch(config-if)# switchport private-vlan trunk allowed vlan 10, 3,301-
302
Switch(config-if)# switchport private-vlan association trunk 3 301
Switch(config-if)# switchport private-vlan association trunk 3 302
```

# pVLAN Verification for Scenario 2

```
Switch# show interfaces fastethernet 5/2 switchport
Name: Fa5/2
Switchport: Enabled
Administrative Mode: private-vlan trunk secondary
Operational Mode: private-vlan trunk secondary
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: 10
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations:
3 (VLAN0003) 301 (VLAN0301)
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Operational Normal VLANs: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
```

# Questions?