VLANs

Objectives

After completing this course, students will be able to

- Define VLAN.
- Identify the benefits provided by implementing VLAN.
- Enumerate the different types of VLANs.
- Explain the use of a trunk.
- Explain the purpose of the native VLAN.
- Demonstrate how to configure VLANs and trunks.
- Define what DTP and explain when to use DTP.
- List the commands to be used to troubleshoot VLANs and trunks.
- Identify the types of security issues are related to VLANs and trunks and explain how to mitigate these issues.
- Identify what are the best practices to use when implementing VLANs and trunks.

Introduction

The role of providing access into a LAN is normally reserved for an access layer switch. A ***virtual local area network (VLAN)*** can be created on a Layer 2 switch to reduce the size of broadcast domains, similar to a Layer 3 device. VLANs are commonly incorporated into network design making it easier for a network to support the goals of an organization.

VLAN Segmentation

One way of breaking a larger network into smaller sections is by implementing VLANs. VLANs allow segmentation, or breaking a large network into smaller ones.

VLAN Definitions

- VLANs provide a way to group devices within a LAN. A group of devices within a VLAN communicate as if they were attached to the same wire.
- VLANs are based on logical connections, instead of physical connections.
- VLANs allow an administrator to segment networks based on factors such as function, project team, or application, without regard for the physical location of the user or device as shown in

Figure 1. Devices within a VLAN act as if they are in their own independent network, even if they share a common infrastructure with other VLANs. Any switch port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations within the VLAN where the packets are sourced.

- Each VLAN is considered a separate logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a device that supports routing.

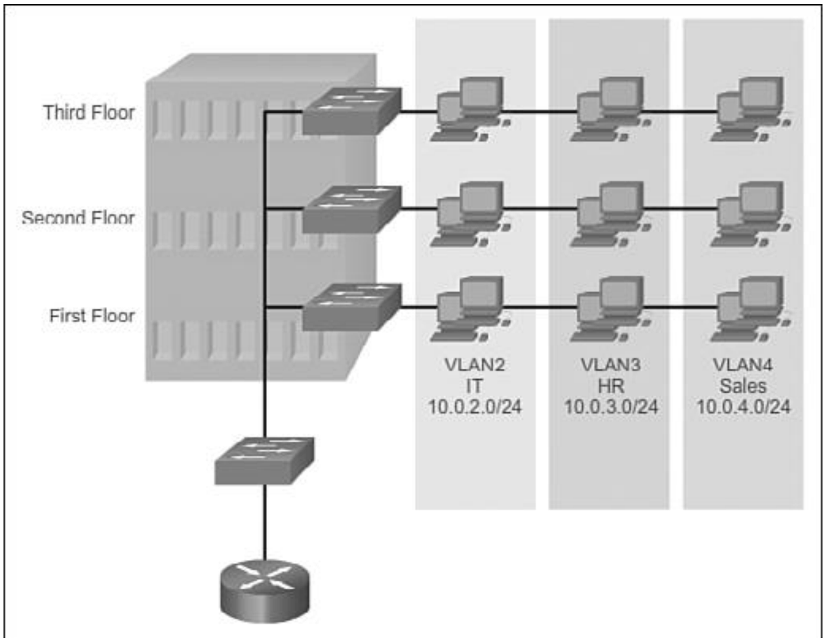


Figure 1 VLAN Groups

A VLAN creates a logical broadcast domain that can span multiple physical LAN segments. VLANs improve network performance by separating large broadcast domains into smaller ones. If a device in one VLAN sends a broadcast Ethernet frame, all devices in the VLAN receive the frame, but devices in other VLANs do not.

VLANs enable the implementation of access and security policies according to specific groupings of users. Each switch port can be assigned to only one VLAN (with the exception of a port connected to an IP phone or to another switch).

Benefits of VLANs

The primary benefits of using VLANs are as follows:

Security: Groups that have sensitive data are separated from the rest of the network, decreasing the chances of confidential information breaches. As shown in Figure 3-2, faculty computers are on VLAN 10 and completely separated from student and guest data traffic.

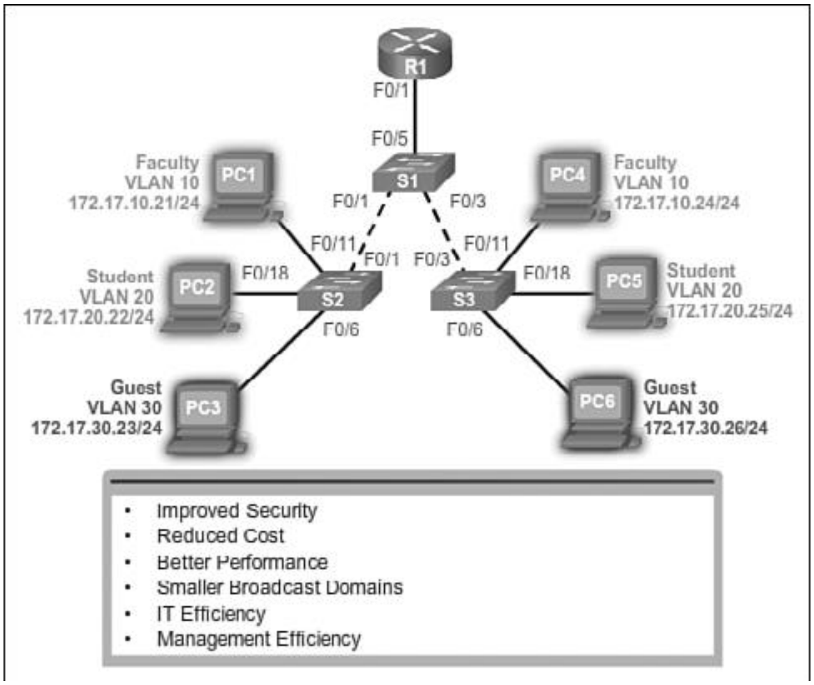
Cost reduction: Cost savings result from reduced need for expensive network upgrades and more efficient use of existing bandwidth and uplinks.

Better performance: Dividing flat Layer 2 networks into multiple logical workgroups (broadcast domains) reduces unnecessary traffic on the network and boosts performance.

Shrink broadcast domains: Dividing a network into VLANs reduces the number of devices in the broadcast domain. As shown in Figure 2, there are six computers on this network, but there are three broadcast domains: Faculty, Student, and Guest.

Improved IT staff efficiency: VLANs make it easier to manage the network because users with similar network requirements share the same VLAN. When a new switch is provisioned, all the policies and procedures already configured for the particular VLAN are implemented when the ports are assigned. It is also easy for the IT staff to identify the function of a VLAN by giving it an appropriate name. In Figure 2, for easy identification VLAN 10 has been named "Faculty," VLAN 20 is named "Student," and VLAN 30 "Guest."

Simpler project and application management: VLANs aggregate users and network devices to support business or geographic requirements. Having separate functions makes managing a project or working with a specialized application easier; an example of such an application is an e-learning development platform for faculty. Each VLAN in a switched network corresponds to an IP network; therefore, VLAN design must take into consideration the implementation of a hierarchical network addressing scheme. A hierarchical network addressing scheme means that IP network numbers are applied to network segments or VLANs in an orderly fashion that takes the network as a whole into consideration. Blocks of contiguous network addresses are reserved for and configured on devices in a specific area of the network, as shown in Figure 2 .



- Improved Security
- Reduced Cost
- Better Performance
- Smaller Broadcast Domains
- IT Efficiency
- Management Efficiency

Figure 2 Benefits of VLANs

Types of VLANs

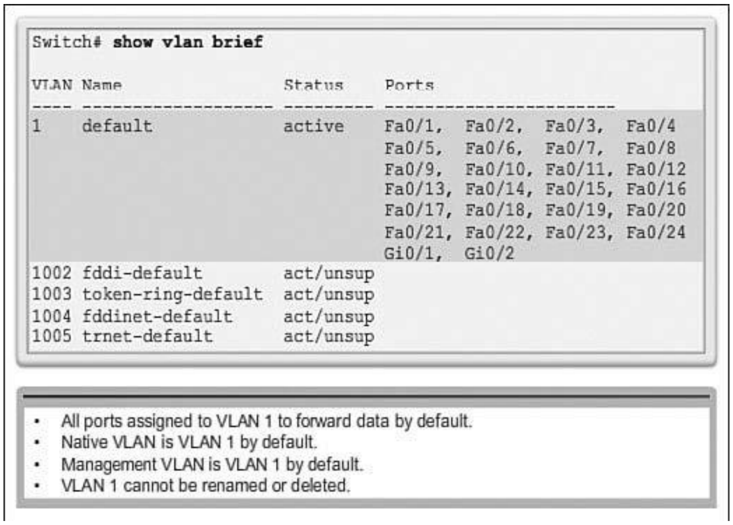
Data VLAN

A ***data VLAN*** is a VLAN that is configured to carry user-generated traffic. A VLAN carrying voice or management traffic would not be part of a data VLAN. It is common practice to separate voice and management traffic from data traffic. A data VLAN, is sometimes referred to as a user VLAN. Data VLANs are used to separate the network into groups of users or devices.

Default VLAN

All switch ports become a part of the ***default VLAN*** after the initial boot up of a switch loading the default configuration. Switch ports that participate in the default VLAN are part of the same broadcast domain. This allows any device connected to any switch port to communicate with other devices on other switch ports. The default VLAN for Cisco switches is VLAN 1. In Figure 3, the **show vlan brief** command was issued on a switch running the default configuration. Notice that all ports are assigned to VLAN 1 by default.

VLAN 1 has all the features of any VLAN, except it cannot be renamed or deleted. By default, all Layer 2 control traffic is associated with VLAN 1. In Figure 3, all ports are currently assigned to the default VLAN 1.



```
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
```

- All ports assigned to VLAN 1 to forward data by default.
- Native VLAN is VLAN 1 by default.
- Management VLAN is VLAN 1 by default.
- VLAN 1 cannot be renamed or deleted.

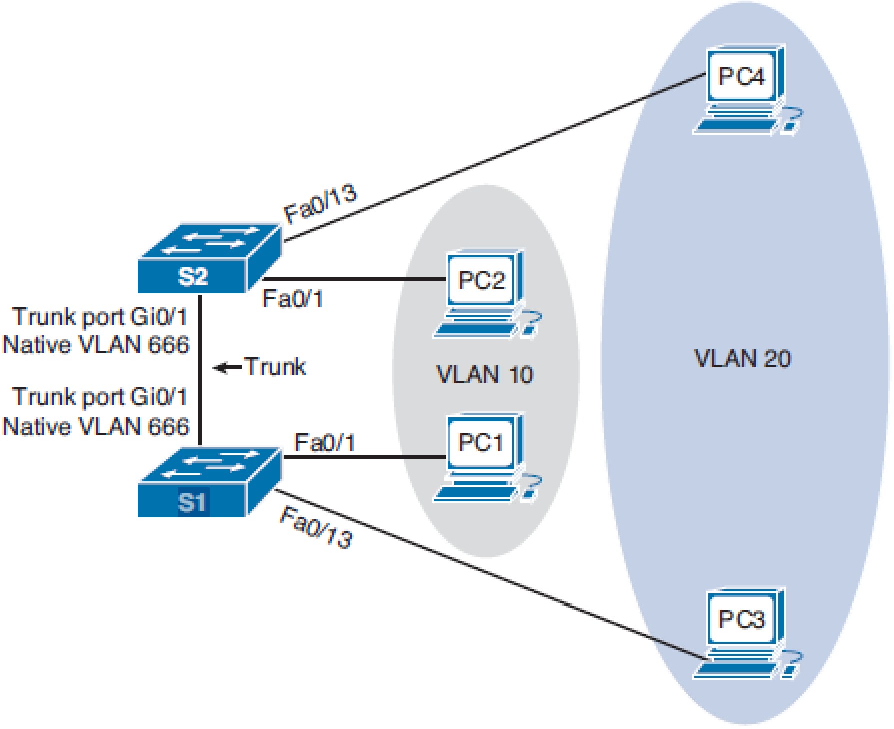
Figure 3 Default VLAN 1

Native VLAN

A ***native VLAN*** is assigned to an 802.1Q ***trunk*** port. Trunk ports are the links between switches that support the transmission of traffic associated with more than one VLAN. An 802.1Q trunk port supports traffic coming from many VLANs (tagged traffic), as well as traffic that does not come from a VLAN (untagged traffic). The 802.1Q trunk port places untagged traffic on the native VLAN, which by default is VLAN 1.

Native VLANs are defined in the IEEE 802.1Q specification to maintain backward compatibility with untagged traffic common to legacy LAN scenarios. A native VLAN serves as a common identifier on opposite ends of a trunk link.

It is a best practice to configure the native VLAN as an unused VLAN, distinct from VLAN 1 and other VLANs. In fact, it is not unusual to dedicate a fixed VLAN to serve the role of the native VLAN for all trunk ports in the switched domain. Look at Figure 4.Traffic from VLANs 10 and 20 cross the trunk. A tag is added with the VLAN number before the data leaves the switch port. An unused VLAN number is configured as the native VLAN.



Traffic from VLANs 10 and 20 cross the trunk. A tag is added with the VLAN number before the data leaves the switch port. An unused VLAN number is configured as the native VLAN.

Figure 4 Native VLAN

PC1 and PC2 are in VLAN 10. PC3 and PC4 are in VLAN 20. Traffic from both VLANs crosses the trunk link that is configured between the two switches. If PC1was sending traffic to PC2, as the data leaves the S1 Gi0/1 port, the S1 switch would “tag” the traffic with VLAN 10. When S2 receives the tag, the switch removes it and sends the data on to PC2. The native VLAN should be an unused VLAN, as shown in Figure 4. If any devices were configured in the native VLAN, the switches would not tag the traffic before it is placed on the trunk link.

Management VLAN

A ***management VLAN*** is any VLAN configured to access the management capabilities of a switch. VLAN 1 is the management VLAN by default. To create the management VLAN, the switch virtual interface (SVI) of that VLAN is assigned an IP address and subnet mask, allowing the switch to be managed via HTTP, Telnet, SSH, or SNMP.

In the past, the management VLAN for a 2960 switch was the only active SVI. On 15.x versions of the Cisco IOS for Catalyst 2960 Series switches, it is possible to have more than one active SVI. With Cisco IOS 15.x, the particular active SVI assigned for remote management must be documented. Although theoretically a switch can have more than one management VLAN, having more than one increases exposure to network attacks.

Note

If the native VLAN is the same as the management VLAN, a security risk exists. The native VLAN, when used, and the management VLAN should always be a VLAN number distinct from any other VLANs.

Voice VLANs

A separate VLAN known as a ***voice VLAN*** is needed to support Voice over IP

(VoIP). VoIP traffic requires:

- Assured bandwidth to ensure voice quality
- Transmission priority over other types of network traffic
- Capability to be routed around congested areas on the network
- Delay of less than 150 ms across the network

VLANs in a Multi switched Environment

Even a small business might have more than one switch. Multiple switch configuration and design influences network performance. Trunks are commonly used to connect a switch to a switch or to another network device such as a router.

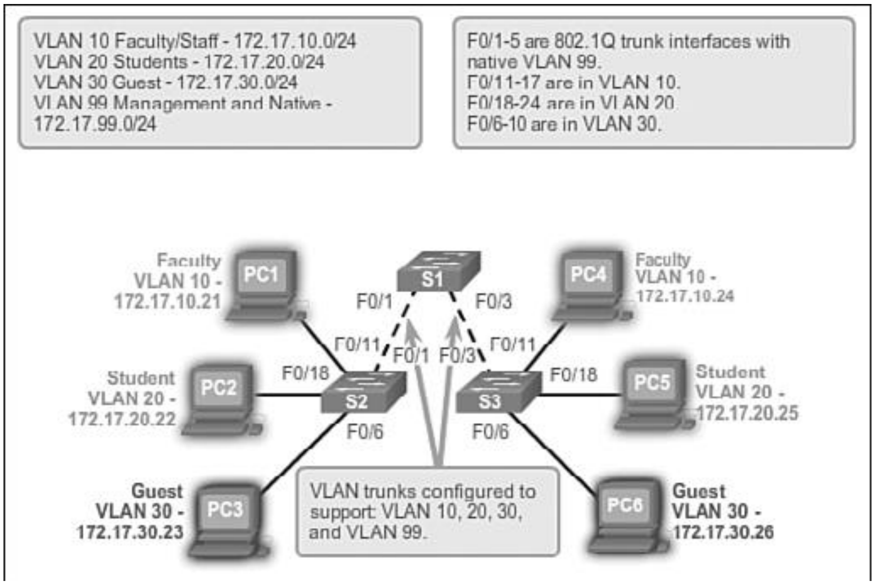
VLAN Trunks

A VLAN trunk, or trunk, is a point-to-point link between two network devices that carries more than one VLAN. A VLAN trunk extends VLANs across two or more network devices. Cisco supports IEEE 802.1Q for coordinating trunks on Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet interfaces.

VLANs would not be very useful without VLAN trunks. VLAN trunks allow all VLAN traffic to propagate between switches, so that devices which are in the same VLAN, but connected to different switches, can communicate without the intervention of a router.

A VLAN trunk does not belong to a specific VLAN; rather, it is a conduit for multiple VLANs between switches and routers. A trunk could also be used between a network device and server or other device that is equipped with an appropriate 802.1Q-capable NIC. By default, on a Cisco Catalyst switch, all VLANs are supported on a trunk port.

In Figure 5, the links between switches S1 and S2, and S1 and S3 are configured to transmit traffic coming from VLANs 10, 20, 30, and 99 across the network. This network could not function without VLAN trunks.



VLAN 10 Faculty/Staff - 172.17.10.0/24
VLAN 20 Students - 172.17.20.0/24
VLAN 30 Guest - 172.17.30.0/24
VLAN 99 Management and Native - 172.17.99.0/24

F0/1-5 are 802.1Q trunk interfaces with native VLAN 99.
F0/11-17 are in VLAN 10.
F0/18-24 are in VLAN 20.
F0/6-10 are in VLAN 30.

VLAN trunks configured to support VLAN 10, 20, 30, and VLAN 99.

Figure 5 Trunks

Tagging Ethernet Frames for VLAN Identification

Layer 2 devices use the Ethernet frame header information to forward packets. The standard Ethernet frame header does not contain information about the VLAN to which the frame belongs; thus, when Ethernet frames are placed on a trunk, information about the VLANs to which they belong must be added. This process, called ***tagging***, is accomplished by using the IEEE 802.1Q header specified in the IEEE 802.1Q standard. The 802.1Q header includes a 4-byte tag inserted within the original Ethernet frame header, specifying the VLAN to which the frame belongs, as shown in Figure 6.

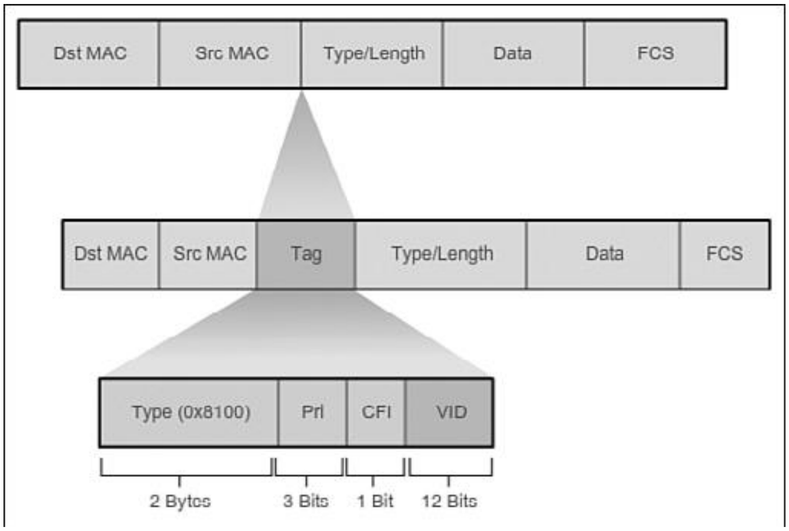


Figure 6 Fields in an Ethernet 802.1Q Frame

When the switch receives a frame on a port configured in access mode and assigned a VLAN, the switch inserts a VLAN tag in the frame header, recalculates the FCS, and sends the tagged frame out of a trunk port.

VLAN Tag Field Details

The VLAN tag field consists of a Type field, a tag control information field, and the FCS field:

- **Type:** A 2-byte value called the tag protocol ID (TPID) value. For Ethernet, it is set to hexadecimal 0x8100.
- **User priority:** A 3-bit value that supports level or service implementation.
- **Canonical Format Identifier (CFI):** A 1-bit identifier that enables Token Ring frames to be carried across Ethernet links.
- **VLAN ID (VID):** A 12-bit VLAN identification number that supports up to 4096 VLAN IDs.

After the switch inserts the Type and tag control information fields, it recalculates the FCS values and inserts the new FCS into the frame.

Native VLANs and 802.1Q Tagging

Native VLANs frequently baffle students. Keep in mind that all trunks have a native VLAN whether you configure it or not. It is best if you control the VLAN ID used as the native VLAN on a trunk.

Tagged Frames on the Native VLAN

Some devices that support trunking add a VLAN tag to native VLAN traffic. Control traffic sent on the native VLAN should not be tagged. If an 802.1Q trunk port receives a tagged frame with the VLAN ID the same as the native VLAN, it drops the frame. Consequently, when configuring a switch port on a Cisco switch, configure devices so that they do not send tagged frames on the native VLAN. Devices from other vendors that support tagged frames on the native VLAN include IP phones, servers, routers, and non-Cisco switches.

Untagged Frames on the Native VLAN

When a Cisco switch trunk port receives untagged frames (which are unusual in a well-designed network), the switch forwards those frames to the native VLAN. If there are no devices associated with the native VLAN (which is not unusual) and there are no other trunk ports, then the frame is dropped. The default native VLAN is VLAN 1 on a Cisco switch. When configuring an 802.1Q trunk port, a default Port VLAN ID (PVID) is assigned the value of the native VLAN ID. All untagged traffic coming in or out of the 802.1Q port is forwarded based on the PVID value. For example, if VLAN 99 is configured as the native VLAN, the PVID is 99 and all untagged traffic is forwarded to VLAN 99. If the native VLAN has not been reconfigured, the PVID value is set to VLAN 1.

In Figure 7, PC1 is connected by a hub to an 802.1Q trunk link. PC1 sends untagged traffic which the switches associate with the native VLAN configured on the trunk ports, and forward accordingly. Tagged traffic on the trunk received by PC1 is dropped. This scenario reflects poor network design for several reasons: it uses a hub, it has a host connected to a trunk link, and it implies that the switches have access ports assigned to the native VLAN. But it illustrates the motivation for the IEEE 802.1Q specification for native VLANs as a means of handling legacy scenarios.A better designed network without a hub is shown in Figure 8.

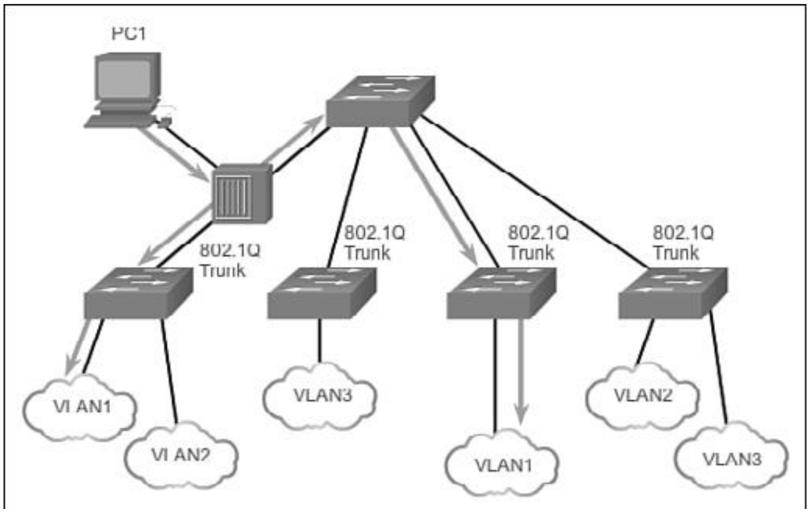


Figure 7 Native VLAN on 802.1Q Trunk

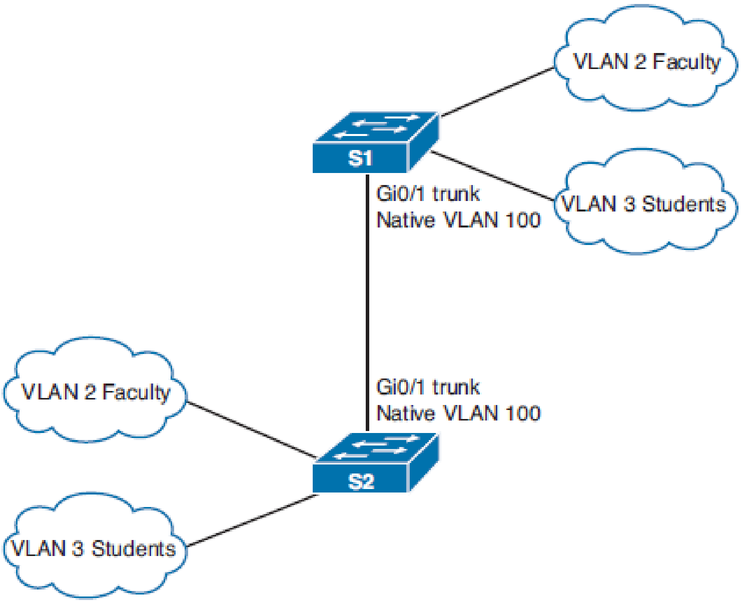


Figure 8 Better Native VLAN Design

Voice VLAN Tagging

As shown in Figure 9, the F0/18 port on S3 is configured to be in voice mode so that voice frames will be tagged with VLAN 150. Data frames coming through the Cisco IP phone from PC5 are left untagged. Data frames destined for PC5 coming from port F0/18 are tagged with VLAN 20 on the way to the phone. The phone strips the VLAN tag before the data is forwarded to PC5.

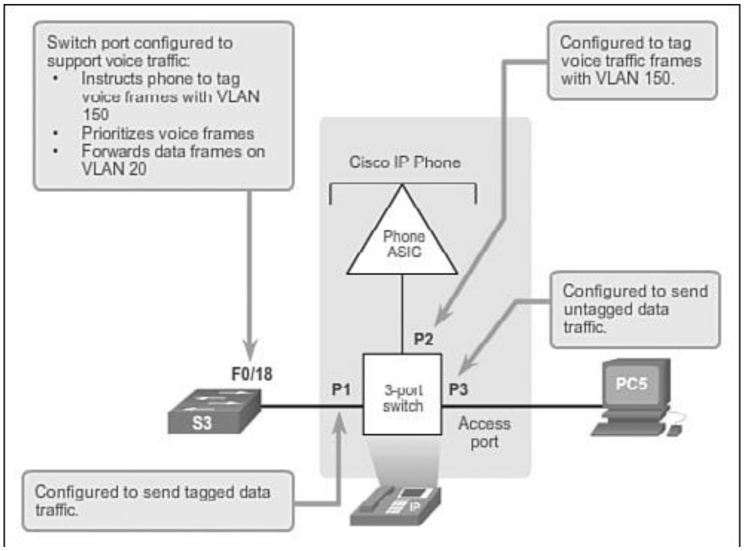

Switch port configured to support voice traffic:
• Instructs phone to tag voice frames with VLAN 150
• Prioritizes voice frames
• Forwards data frames on VLAN 20
Configured to tag voice traffic frames with VLAN 150.
Cisco IP Phone
Phone ASIC
Configured to send untagged data traffic.
P2
F0/18
P1
3-port switch
P3
PC5
S3
Access port
Configured to send tagged data traffic.
IP

Figure 9 Voice VLAN Tagging

The Cisco IP phone contains an integrated three-port 10/100 switch. The ports provide dedicated connections to these devices:

- Port 1 connects to the switch or other VoIP device.
- Port 2 is an internal 10/100 interface that carries the IP phone traffic.
- Port 3 (access port) connects to a PC or other device.

When the switch port has been configured with a voice VLAN, the link between the switch and the IP phone acts as a trunk to carry both the tagged voice traffic and untagged data traffic. Communication between the switch and IP phone is facilitated by the Cisco Discovery Protocol (CDP).

Sample Configuration

Look at the sample output.

```
S1# show interfaces fa0/18 switchport
Name: Fa0/18
Switchport: Enabled
Administrative Mode: static access
Operational Mode: down
Administrative Trunking Encapsulation: dot1q
Negotiation of Trunking: Off
Access Mode VLAN: 20 (student)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: 150 (voice)
<output omitted>
```

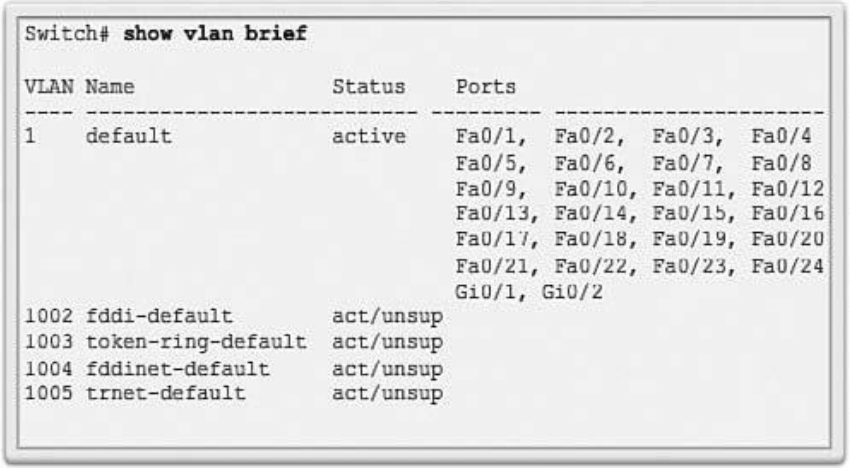
A discussion of voice Cisco IOS commands are beyond the scope of this course, but the highlighted areas in the sample output show the F0/18 interface configured with a VLAN configured for data (VLAN 20) and a VLAN configured for voice (VLAN 150).

VLAN Implementations

VLANs allow multiple networks to exist on one or more switches. Companies commonly use VLANs to separate a user network from other networks such as a voice network, printer/copier network, and guest network.

VLAN Ranges on Catalyst Switches

Different Cisco Catalyst switches support various numbers of VLANs. The number of supported VLANs is large enough to accommodate the needs of most organizations. For example, the Catalyst 2960 and 3560 Series switches support more than 4000 VLANs. Normal range VLANs on these switches are numbered 1 to 1005 and extended range VLANs are numbered 1006 to 4094. Figure 10 illustrates the available VLAN IDs on a Catalyst 2960 switch running Cisco IOS Release 15.x.



```
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
```

Figure 10 Normal VLAN ID Range

Normal Range VLANs

Used in small- and medium-sized business and enterprise networks.

- Identified by a VLAN ID between 1 and 1005.
- IDs 1002 through 1005 are reserved for Token Ring and FDDI VLANs.
- IDs 1 and 1002 to 1005 are automatically created and cannot be removed.
- Configurations are stored within a VLAN database file, called ***vlan.dat***. The vlan.dat file is located in the flash memory of the switch.
- The ***VLAN Trunking Protocol (VTP)*** is a Cisco-proprietary Layer 2 protocol used to manage VLAN configurations between switches; VTP can learn and store only normal range VLANs.

Extended Range VLANs

- Enable service providers to extend their infrastructure to a greater number of customers. Some global enterprises could be large enough to need extended range VLAN IDs.
- Are identified by a VLAN ID between 1006 and 4094.
- Configurations are not written to the vlan.dat file.
- Support fewer VLAN features than normal range VLANs.
- Are, by default, saved in the running configuration file.
- VTP does not learn extended range VLANs.

Note

Because there are 12 bits in the VLAN ID field of the IEEE 802.1Q header, 4096 is the upper boundary for the number of VLANs available on Catalyst switches.

Creating a VLAN

When configuring normal range VLANs, the configuration details are stored in flash memory on the switch in a file called vlan.dat. Flash memory is persistent and does not require the **copy running-config startup-config** command. However, because other details are often configured on a Cisco switch at the same time that VLANs are created, it is good practice to save running configuration changes to the startup configuration.

Table 3-1 displays the Cisco IOS command syntax used to add a VLAN to a switch and give it a name.

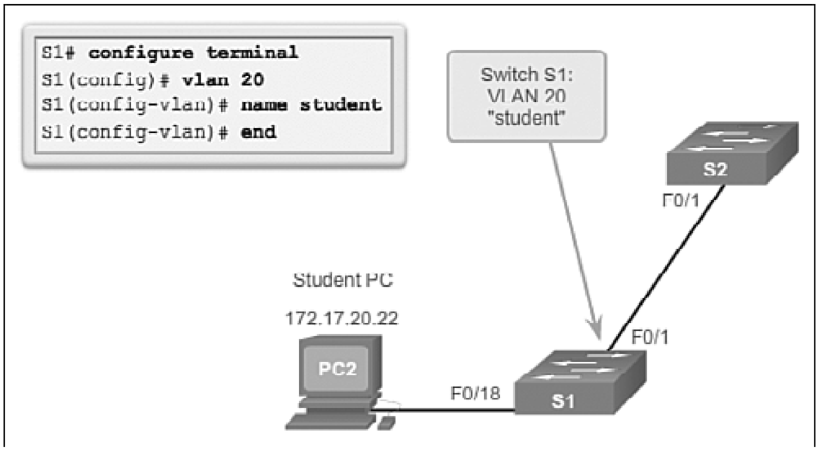
Note

Naming each VLAN is considered a best practice in switch configuration.

Table 3-1 Commands Used to Create a VLAN

Enter global configuration mode.	S1# **configure terminal**
Create a VLAN with a valid VLAN 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**

Figure 11 shows how the student VLAN (VLAN 20) is configured on switch S1. In the topology example, the student computer (PC1) has not been associated with a VLAN yet, but it does have an IP address of 172.17.20.22.



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

Figure 11 Sample VLAN Configuration

Assigning Ports to VLANs

After creating a VLAN, the next step is to assign ports to the VLAN. An access port can belong to only one VLAN at a time; one exception to this rule is that of a port connected to an IP phone, in which case, there are two VLANs associated with the port: one for voice and one for data.

Table 3-2 displays the syntax for defining a port to be an access port and assigning it to a VLAN. The **switchport mode access** command is optional, but strongly recommended as a security best practice. With this command, the interface changes to permanent access mode.

Note

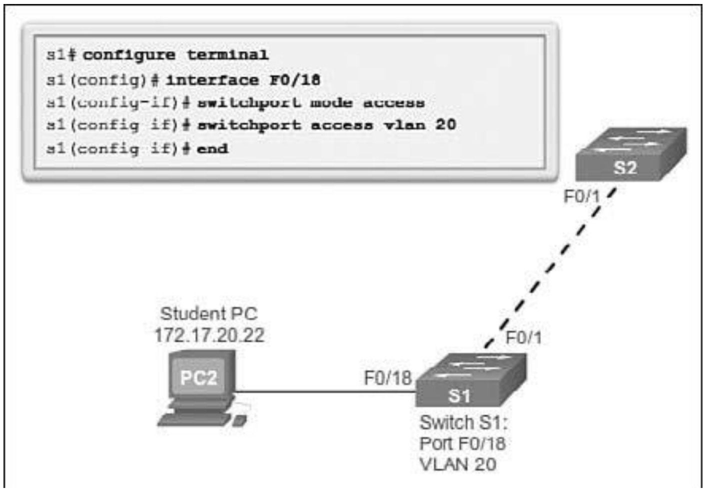
Use the **interface range** command to simultaneously configure multiple interfaces.

Table 3-2 Commands Used to Assign Ports to VLANs

Enter global configuration mode.	S1# **configure terminal**
Enter interface configuration mode for a particular port number	S1(config)# **interface** *interface_id*
Set the port to access mode.	S1(config-if)# **switchport mode access**
Assign the port to a particular VLAN.	S1(config-if)# **switchport access vlan** *vlan-id*

Return to the privileged EXEC mode.	S1(config-if)# **end**

In Figure 12, VLAN 20 is assigned to port F0/18 on switch S1; therefore, the student computer (PC2) is in VLAN 20. When VLAN 20 is configured on others witches, the network administrator knows to configure the other student computers to be in the same subnet as PC2 (172.17.20.0/24).



```
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
```

Figure 12 Sample VLAN Interface Configuration

The **switchport access vlan** command forces the creation of a VLAN if it does not already exist on the switch. For example, VLAN 30 is not present in the **show vlan brief** output of the switch. If the **switchport access vlan 30** command is entered on any interface with no previous configuration, then the switch displays the following:

% Access VLAN does not exist. Creating vlan 30

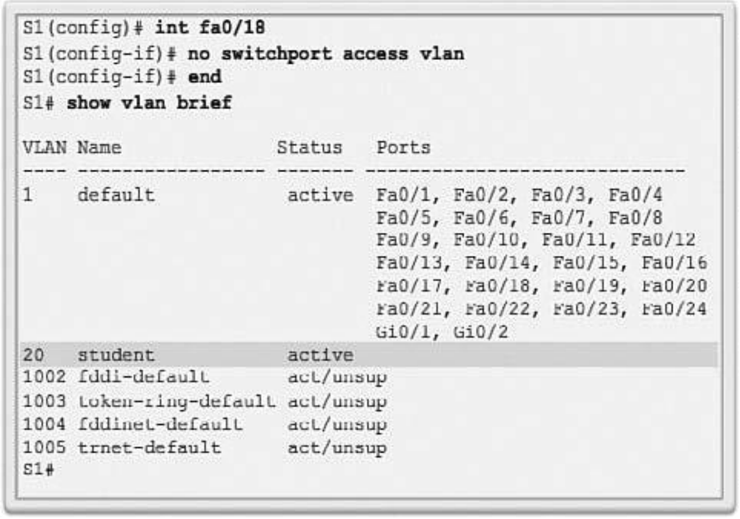
Changing VLAN Port Membership

There are a number of ways to change VLAN port membership. Table 3-3 shows the syntax for changing a switch port to VLAN 1 membership with the **no switchport access vlan** interface configuration mode command.

Table 3-3 Remove **VLAN** Configuration Commands

Enter global configuration mode.	S1# **configure terminal**
Enter interface configuration mode for a particular port number.	S1(config)# **interface** *interface_id*
Assign the port to a particular VLAN.	S1(config-if)# **no switchport access vlan** *vlan-id*
Return to the privileged EXEC mode.	S1(config-if)# **end**

Interface F0/18 was previously assigned to VLAN 20. The **no switchport access vlan** command is entered for interface F0/18. Examine the output in the **show vlan brief** command that immediately follows as shown in Figure 13. The **show vlan brief** command displays the VLAN assignment and membership type for all switch ports. The **show vlan brief** command displays one line for each VLAN. The output for each VLAN includes the VLAN name, status, and switch ports.



```
S1(config)# int fa0/18
S1(config-if)# no switchport access vlan
S1(config-if)# end
S1# 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
20   student              active
1002 fddi-default         act/unsup
1003 token-ring-default act/unsup
1004 fddinet-default      act/unsup
1005 trnet-default        act/unsup
S1#
```

Figure 13. Sample Interface Removal of a VLAN

VLAN 20 is still active, even though no ports are assigned to it. The **show interfaces fa0/18 switchport** output verifies that the access VLAN for interface F0/18 has been reset to VLAN 1.

```
S1# show interfaces fa0/18 switchport
Name: Fa0/18
Switchport: Enabled
Administrative Mode: static access
Operational Mode: down
Administrative Trunking Encapsulation: dot1q
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
<output omitted>
```

A port can easily have its VLAN membership changed. It is not necessary to first remove a port from a VLAN to change its VLAN membership. When an access port has its VLAN membership reassigned to another existing VLAN, the new VLAN membership simply replaces the previous VLAN membership. In the following output, port F0/11 is assigned to VLAN 20.

```
S1# config t
S1(config)# interface fastethernet0/11
S1(config-if)# switchport mode access
S1(config-if)# switchport access vlan 20
% Access VLAN does not exist. Creating vlan 20
S1(config-if)# end
S1# 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, Fa0/12, Fa0/13
Fa0/14, Fa0/15, Fa0/16, Fa0/17
Fa0/18, Fa0/19, Fa0/20, Fa0/21
```

Fa0/22, Fa0/23, Fa0/24, Gig1/1

Gig1/2

20 VLAN0020 active Fa0/11

1002 fddi-default act/unsup

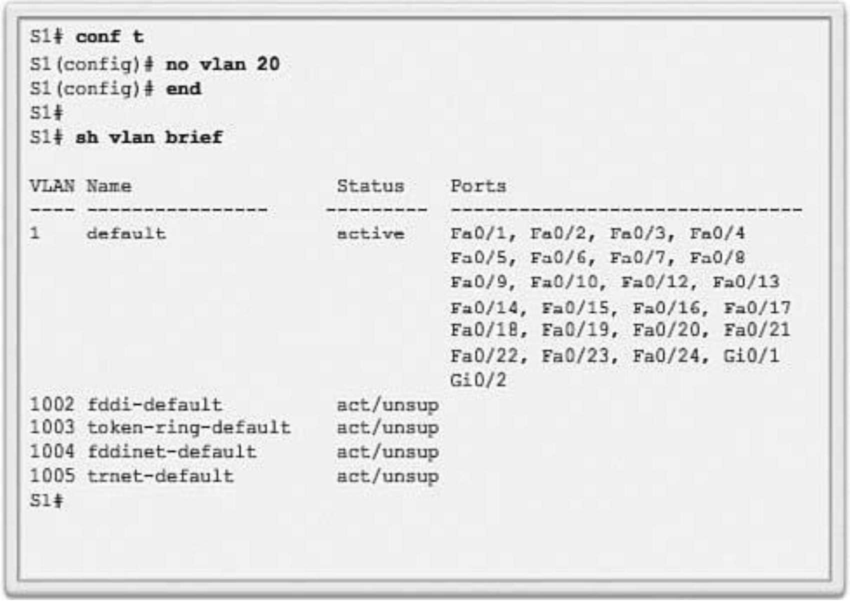
1003 token-ring-default act/unsup

1004 fddinet-default act/unsup

1005 trnet-default act/unsup

Deleting VLANs

In Figure 14, the **no vlan** *vlan-id* global configuration mode command is used to remove VLAN 20 from the switch. Switch S1 had a minimal configuration with all ports in VLAN 1 and an unused VLAN 20 in the VLAN database. The **show vlan brief** command verifies that VLAN 20 is no longer present in the vlan.dat file after using the **no vlan 20** command.



```
S1# conf t
S1(config)# no vlan 20
S1(config)# end
S1#
S1# sh 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, Fa0/12, 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
S1#
```

Figure 14 Deleting a VLAN

Caution

Before deleting a VLAN, be sure to first reassign all member ports to a different VLAN. Any ports that are not moved to an active VLAN are unable to communicate with other hosts after the VLAN is deleted and until they are assigned to an active VLAN.

Alternatively, the entire vlan.dat file can be deleted using the **delete flash:vlan.dat** privileged EXEC mode command. The abbreviated command version (**delete vlan.dat**) can be used if the vlan.dat file has not been moved from its default location.After issuing this command and reloading the switch, the previously configured VLANs are no longer present. This effectively places the switch into its factory default condition concerning VLAN configurations.

Note

For a Cisco Catalyst switch, the **erase startup-config** command must accompany the **delete vlan.dat** command prior to using the **reload** command to restore the switch to its factory default condition.

Verifying VLAN Information

After a VLAN is configured, VLAN configurations can be validated using Cisco IOS **show** commands. Table 3-4 shows common **show vlan** command options.

Table 3-4 The show vlan Command Options

show vlan [**brief** | **id** *vlan-id* | **name** *vlan-name* | **summary**]

Display one line for each VLAN with the VLAN name, status, and associated ports.	**Brief**
Display information about a single VLAN identified by the VLAN ID number, which can be a number between 1 and 4094	**id** *vlan-id*
Display information about a single VLAN identified by a VLAN name. The VLAN name is an ASCII string from 1 to 32 characters.	**name** *vlan-name*
Display VLAN summary information.	**Summary**

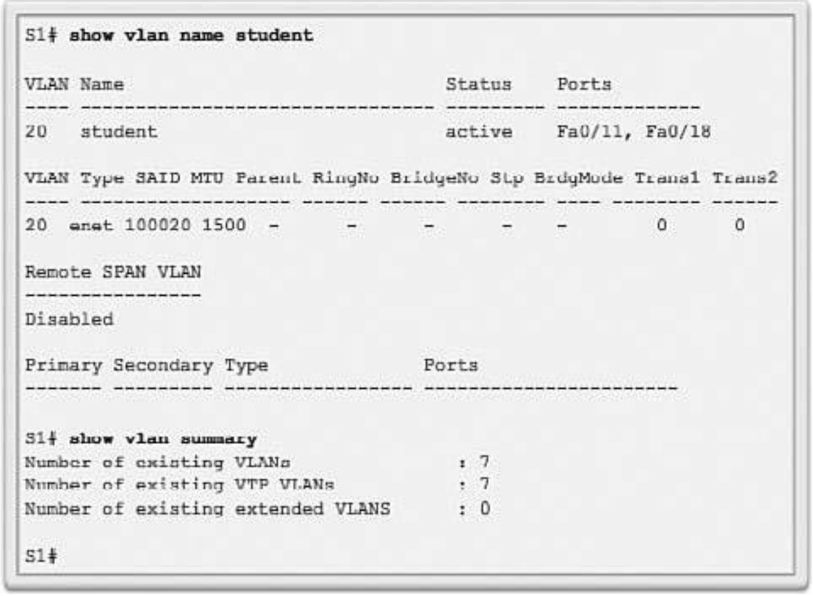
Table 3-5 shows common **show interfaces** command options.

Table 3-5 The show interfaces Command Options

show interfaces [*interface-id* | **vlan** *vlan-id*] | **switchport**

Valid interfaces include physical ports (including type, module, and port number) and port channels. The port-channel range is 1 to 6.	*interface-id*
VLAN identification, which is a number from 1 to 4094.	**vlan** *vlan-id*
Display the administrative and operational status of a switch port, including port blocking and port protection settings.	**Switchport**

In Figure 15, the **show vlan name student** command produces output that is not easily interpreted. The preferable option is to use the **show vlan brief** command. The **show vlan summary** command displays the count of all configured VLANs. The output in Figure 3-18 shows seven VLANs.



```
S1# show vlan name student

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------
20   student                          active    Fa0/11, Fa0/18

VLAN Type SAID MTU Parent RingNo BridgeNo Stp BrdgMode Trans1 Trans2
---- ------------------ ------ ------ -------- ---- -------- ------
20   enet 100020 1500  -      -      -      -    -         0      0

Remote SPAN VLAN
----------------
Disabled

Primary Secondary Type              Ports
------- --------- ----------------- -----------------------

S1# show vlan summary
Number of existing VLANs              : 7
Number of existing VTP VLANs          : 7
Number of existing extended VLANS     : 0

S1#
```

Figure 15 Using the **show vlan** Command

The **show interfaces vlan** *vlan-id* command displays details that are beyond the scope of this course. The important information appears on the second line in the output, indicating that VLAN 20 is up.

```
S1# show interfaces vlan 20
Vlan 20 is up, line protocol is down
Hardware is EtherSVI, address is 001c.57ec.0641 (bia 001c.57ec.0641)
MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes);

Total output drops: 0
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts (0 IP multicast)
0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 interface resets
0 output buffer failures, 0 output buffers swapped out
<output omitted>
```

VLAN Trunks

Trunks are commonly used between switches and other network devices such as a router, another switch, or a server. A network technician must be very familiar with configuring a trunk and ensuring it works properly.

Configuring IEEE 802.1Q Trunk Links

A VLAN trunk is an OSI Layer 2 link between two switches that carries traffic for all VLANs (unless the allowed VLAN list is restricted manually or dynamically). To enable trunk links, configure the ports on either end of the physical link with parallel sets of commands.

To configure a switch port on one end of a trunk link, use the **switchport mode trunk** command. With this command, the interface changes to permanent trunking mode. The port enters into a Dynamic Trunking Protocol (DTP) negotiation to convert the link into a trunk link even if the interface connecting to it does not agree to the change.

The Cisco IOS command syntax to specify a native VLAN (other than VLAN 1) is shown in Table 3-6. In the example, VLAN 99 is configured as the native VLAN using the **switchport trunk native vlan 99** command.

Use the Cisco IOS **switchport trunk allowed vlan** *vlan-list* command to specify the list of VLANs to be allowed on the trunk link.

Table 3-6 Switch Port Trunk Commands

Enter global configuration mode.	S1# **configure terminal**
Enter interface configuration mode for a particular port number.	S1(config)# **interface** *interface_id*
Optionally, put the trunk in the appropriate trunking mode if the switch supports more than one mode.	S1(config-if)# **switchport trunk** **encapsulation** [**dot1q** | **isl**]
Force the link to be a trunk link. S1(config-if)# **switchport mode trunk**	S1(config-if)# **switchport trunk native** **vlan** *vlan_id*

Specify a native VLAN for untagged 802.1Q frames.	
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**

.In Figure 16, VLANs 10, 20, and 30 support the Faculty, Student, and Guest computers (PC1, PC2, and PC3). The F0/1 port on switch S1 is configured as a trunk port and forwards traffic for VLANs 10, 20, and 30. VLAN 99 is configured as the native VLAN.

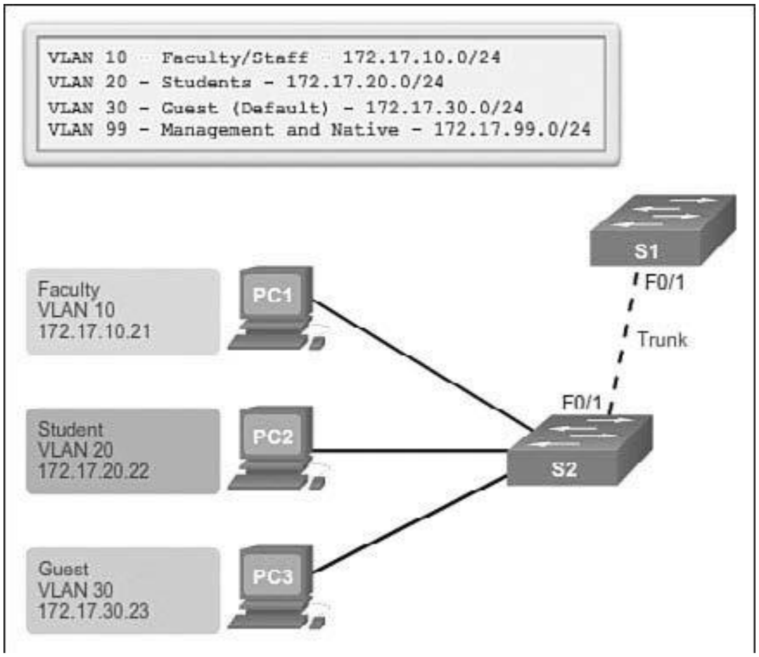

VLAN 10 · Faculty/Staff · 172.17.10.0/24
VLAN 20 - Students - 172.17.20.0/24
VLAN 30 - Guest (Default) - 172.17.30.0/24
VLAN 99 - Management and Native - 172.17.99.0/24
S1
F0/1
Trunk
F0/1
S2
Faculty
VLAN 10
172.17.10.21
PC1
Student
VLAN 20
172.17.20.22
PC2
Guest
VLAN 30
172.17.30.23
PC3

Figure 16 Sample VLAN Design

Look at the configuration of port F0/1 on switch S1 as a trunk port. The native VLAN is changed to VLAN 99 and the allowed VLAN list is restricted to 10, 20, and 30. If the native VLAN is not allowed on the trunk link, the trunk will not allow any data traffic for the native VLAN.

```
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
S1(config-if)# end
```

Note

This configuration assumes the use of Cisco Catalyst 2960 switches, which automatically use 802.1Q encapsulation on trunk links. Other switches may require manual configuration of the encapsulation. Always configure both ends of a trunk link with the same native VLAN. If 802.1Q trunk configuration is not the same on both ends, Cisco IOS Software reports errors.

Resetting the Trunk to Default State

Table 3-7 shows the commands to remove the allowed VLANs and reset the native VLAN of the trunk. When reset to the default state, the trunk allows all VLANs and uses VLAN 1 as the native VLAN.

Table 3-7 Resetting Configures Values on Trunk Lines

Enter global configuration mode.	S1# **configure terminal**
Enter interface onfiguration mode for a particular port number.	S1(config)# **interface** *interface_id*
Set trunk to allow all VLANs.	S1(config-if)# **no switchport trunk allowed vlan**
Reset the native VLAN to the default.	S1(config-if)# **no switchport trunk native vlan**
Configure the port in access mode.	S1(config-if)# **switchport mode access**
Optionally, remove the trunk mode if it was entered.	S1(config-if)# **no switchport trunk encapsulation** [**dot1q** | **isl**]

Return to the privileged EXEC mode.	S1(config-if)# **end**

The command to reset the switch port to an access port and, in effect, delete the trunk configuration is also shown.

The following output shows the commands used to reset all trunking characteristics of a trunking interface to the default settings. The **show interfaces f0/1 switchport** command reveals that the trunk has been reconfigured to a default state.

```
S1(config)# interface f0/1
S1(config-if)# no switchport trunk allowed vlan
S1(config-if)# no switchport trunk native vlan
S1(config-if)# end
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: 1 (default)
Administrative Native VLAN tagging: enabled
<output omitted>
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: All
Pruning VLANs Enabled: 2-1001
<output omitted>
```

The following sample output shows the commands used to remove the trunk feature from the F0/1 switch port on switch S1. The **show interfaces f0/1 switchport** command reveals that the F0/1 interface is now in static access mode.

```
S1(config)# interface f0/1
S1(config-if)# switchport mode access
S1(config-if)# end
S1# show interfaces f0/1 switchport
Name: Fa0/1
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
<output omitted>
```

Verifying Trunk Configuration

The following output displays the configuration of switch port F0/1 on switch S1. The configuration is verified with the **show interfaces** *interface-ID* **switchport** command.

```
S1(config)# interface f0/1
S1(config-if)# switchport mode trunk
S1(config-if)# switchport trunk native vlan 99
S1(config-if)# end
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 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>
```

The top highlighted area shows that port F0/1 has its administrative mode set to **trunk**. The port is in trunking mode. The next highlighted area verifies that the native VLAN is VLAN 99. Further down in the output, the bottom highlighted area shows that all VLANs are enabled on the trunk.

Dynamic Trunking Protocol

The ***Dynamic Trunking Protocol (DTP)*** is used to negotiate forming a trunk between two Cisco devices. DTP causes increased traffic, and is enabled by default, but may be disabled.

Introduction to DTP

Ethernet trunk interfaces support different trunking modes. An interface can be set to trunking or nontrunking, or to negotiate trunking with the neighbor interface. Trunk negotiation is managed by the Dynamic Trunking Protocol (DTP), which operates on a point-to-point basis only, between network devices.

DTP is a Cisco proprietary protocol that is automatically enabled on Catalyst 2960 and Catalyst 3560 Series switches. Switches from other vendors do not support DTP. DTP manages trunk negotiation only if the port on the neighbor switch is configured in a trunk mode that supports DTP.

Caution

Some internetworking devices might forward DTP frames improperly, which can cause misconfigurations. To avoid this, turn off DTP on interfaces on a Cisco switch connected to devices that do not support DTP.

The default DTP configuration for Cisco Catalyst 2960 and 3560 switches is dynamic auto as shown in Figure 17 on interface F0/3 of switches S1 and S3.

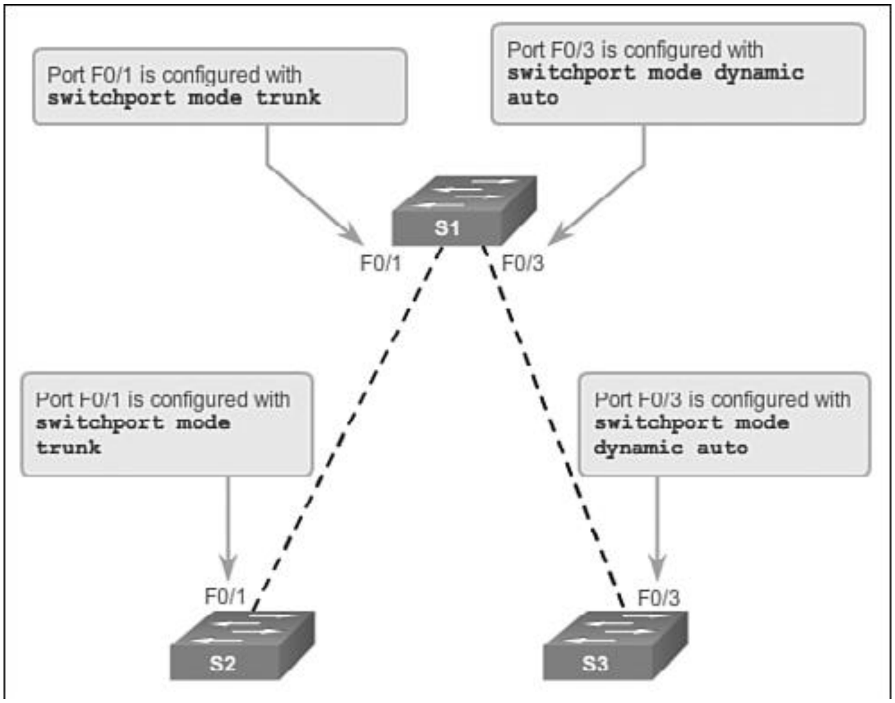

Port F0/1 is configured with
switchport mode trunk
Port F0/3 is configured with
switchport mode dynamic auto
S1
F0/1
F0/3
Port F0/1 is configured with
switchport mode trunk
Port F0/3 is configured with
switchport mode dynamic auto
F0/1
F0/3
S2
S3

Figure 17 Initial DTP Configuration

To enable trunking from a Cisco switch to a device that does not support DTP, use the **switchport mode trunk** and **switchport nonegotiate** interface configuration mode commands. This causes the interface to become a trunk but not generate DTP frames.In Figure 18, the link between switches S1 and S2 becomes a trunk because the F0/1 ports on switches S1 and S2 are configured to ignore all DTP advertisements, and to come up in and stay in trunk port mode. The F0/3 ports on switches S1 and S3 are set to dynamic auto, so the negotiation results in the access mode state. This creates an inactive trunk link. When configuring a port to be in trunk mode, there is no ambiguity about which state the trunk is in; it is always on. With this configuration, it is easy to remember which state the trunk ports are in; if the port is supposed to be a trunk, the mode is set to trunk.

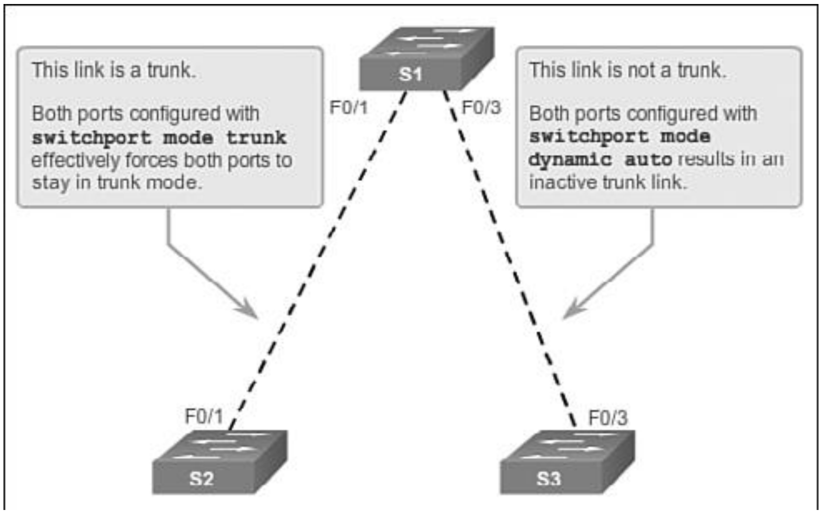

This link is a trunk.
Both ports configured with switchport mode trunk effectively forces both ports to stay in trunk mode.
This link is not a trunk.
Both ports configured with switchport mode dynamic auto results in an inactive trunk link.
S1
F0/1
F0/3
F0/1
F0/3
S2
S3

Figure 18 DTP Interaction Results

Negotiated Interface Modes

Ethernet interfaces on Catalyst 2960 and Catalyst 3560 Series switches support different trunking modes with the help of DTP:

- **switchport mode access:** Puts the interface (access port) into permanent nontrunking mode and negotiates to convert the link into a nontrunk link. The interface becomes a nontrunk interface, regardless of whether the neighboring interface is a trunk interface.\

- **switchport mode dynamic auto:** Makes the interface able to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to trunk or desirable mode. The default switchport mode for newer Cisco switch Ethernet interfaces is **dynamic auto**. Note that if two Cisco switches are left to the common default setting of **auto**, a trunk will never form.

- **switchport mode dynamic desirable:** Makes the interface actively attempt to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to trunk, desirable, or auto mode. This is the default switchport mode on older switches, such as the Catalyst 2950 and 3550 Series switches.

- **switchport mode trunk:** Puts the interface into permanent trunking mode and negotiates to convert the neighboring link into a trunk link. The interface becomes a trunk interface even if the neighboring interface is not a trunk interface.

- **switchport nonegotiate:** Prevents the interface from generating DTP frames. You can use this command only when the interface switchport mode is **access** or **trunk**. You must manually configure the neighboring interface as a trunk interface to establish a trunk link.

Table 3-8 illustrates the results of the DTP configuration options on opposite ends of a trunk link connected to Catalyst 2960 switch ports.

Table 3-8 DTP Negotiated Interface Modes

	Dynamic Auto	**Dynamic Desirable**	**Trunk**	**Access**
Dynamic Auto	Access	Trunk	Trunk	Access
Dynamic Desirable	Trunk	Trunk	Trunk	Access
Trunk	Trunk	Trunk	Trunk	Limited connectivit y
Access	Access	Access	Limited connectivity	Access

Note

Configure trunk links statically whenever possible.

The default DTP mode is dependent on the Cisco IOS Software version and on the platform. To determine the current DTP mode, issue the **show dtp interface** command, as shown in the following output.

```
S1# show dtp interface f0/1
```

```
DTP information for FastEthernet0/1:
TOS/TAS/TNS: TRUNK/ON/TRUNK
TOT/TAT/TNT: 802.1Q/802.1Q/802.1Q

Neighbor address 1:                              0CD996D23F81
Neighbor address 2:                               000000000000
Hello timer expiration (sec/state):              12/RUNNING
Access timer expiration (sec/state):             never/STOPPED
Negotiation timer expiration (sec/state):        never/STOPPED
Multidrop timer expiration (sec/state):          never/STOPPED
FSM state:                                       S6:TRUNK
# times multi & trunk :                          0
Enabled:                                         yes
In STP:                                          no
<output omitted>
```

Troubleshoot VLANs and Trunks

When first learning about switches, students have trouble knowing where to start troubleshooting. Pay particular attention to the **show** commands in this section to verify your configurations using the described techniques instead of simply using the **show running-configuration** command.

IP Addressing Issues with VLAN

Each VLAN must correspond to a unique IP subnet. If two devices in the same VLAN have different subnet addresses, they cannot communicate. This is a common problem, and it is easy to solve by identifying the incorrect configuration and changing the subnet address to the correct one.

In Figure 19, PC1 cannot connect to the web/TFTP server shown.

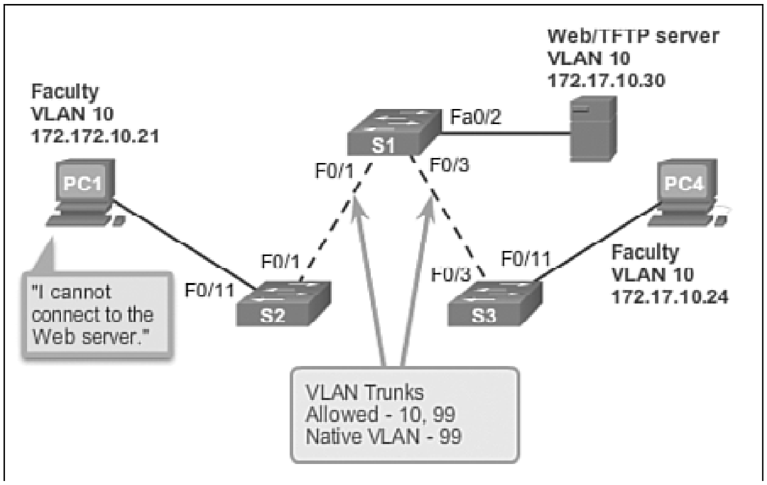
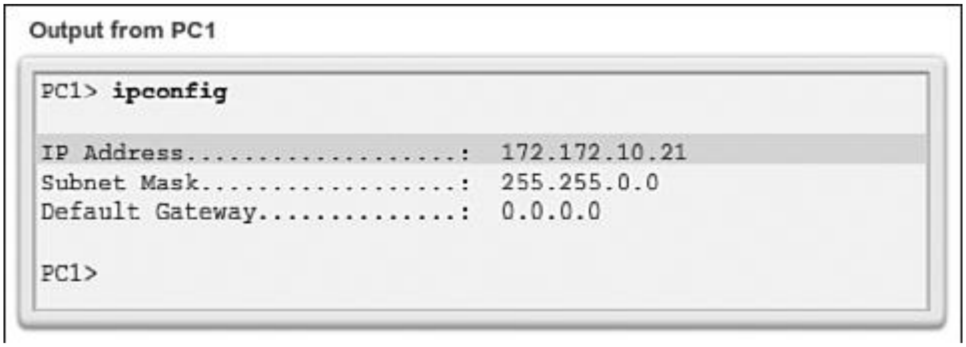


Figure 19 IP Issue Within a VLAN

A check of the IP configuration settings of PC1 shown in Figure 20 reveals the most common error in configuring VLANs: an incorrectly configured IP address. PC1 is configured with an IP address of 172.172.10.21, but it should have been configured with 172.17.10.21.


Output from PC1

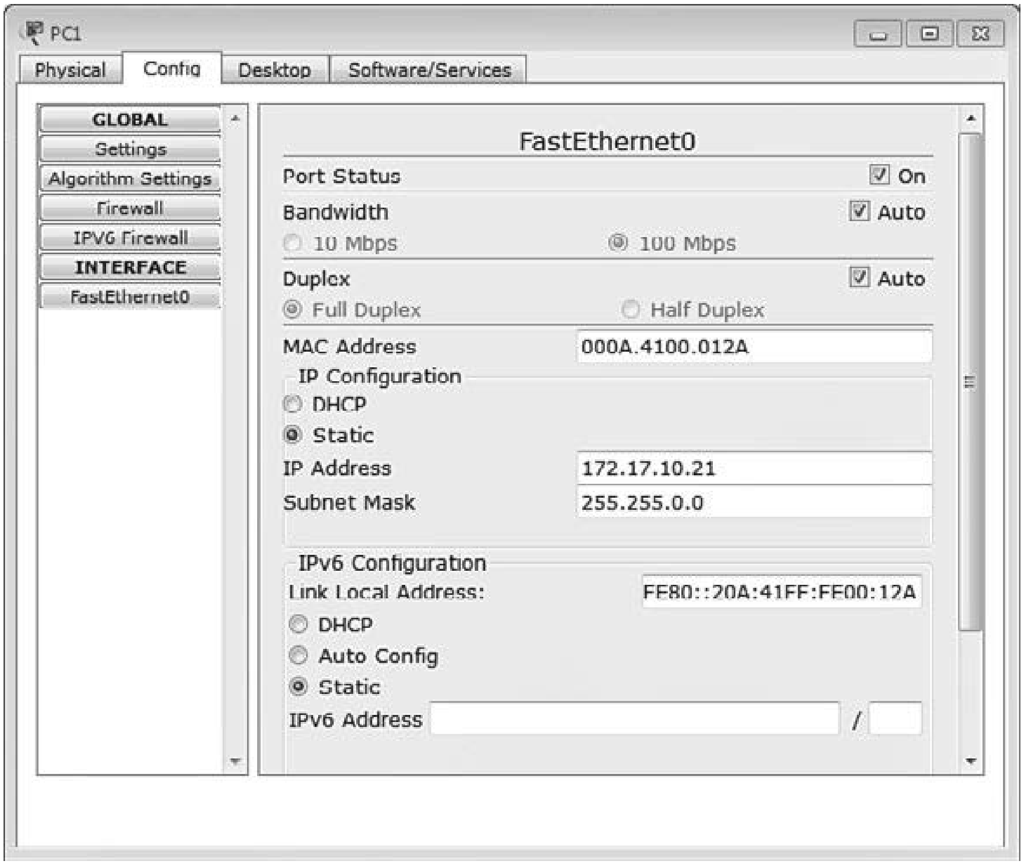
```
PC1> ipconfig

IP Address...................:  172.172.10.21
Subnet Mask..................:  255.255.0.0
Default Gateway..............:  0.0.0.0

PC1>
```

Figure 20 Incorrect IP Address Problem

The PC1 Fast Ethernet configuration dialog box shows the updated IP address of 172.17.10.21. In Figure 21, the output on the bottom reveals that PC1 has regained connectivity to the web/TFTP server found at IP address 172.17.10.30.



FastEthernet0

Port Status ☑ On

Bandwidth ☑ Auto

○ 10 Mbps ● 100 Mbps

Duplex ☑ Auto

● Full Duplex ○ Half Duplex

MAC Address 000A.4100.012A

IP Configuration

○ DHCP

● Static

IP Address 172.17.10.21

Subnet Mask 255.255.0.0

IPv6 Configuration

Link Local Address: FE80::20A:41FF:FE00:12A

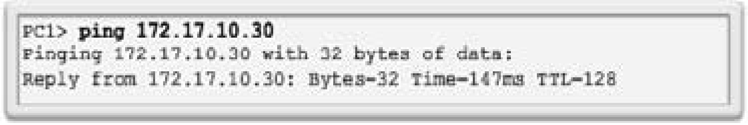
○ DHCP

○ Auto Config

● Static

IPv6 Address /

Output from Computer PC1



```
PC1> ping 172.17.10.30
Pinging 172.17.10.30 with 32 bytes of data:
Reply from 172.17.10.30: Bytes=32 Time=147ms TTL=128
```

Figure 21 Change PC IP Address

Missing VLANs

If there is still no connection between devices in a VLAN, but IP addressing issues have been ruled out, see the flowchart in Figure 3-25 to troubleshoot.

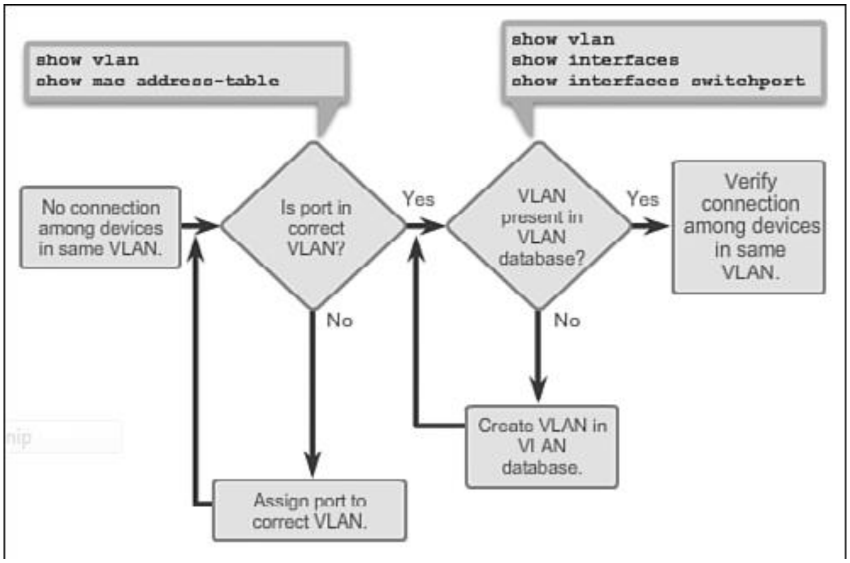

show vlan
show mac address-table
show vlan
show interfaces
show interfaces switchport
No connection among devices in same VLAN.
Is port in correct VLAN?
Yes
VLAN present in VLAN database?
Yes
Verify connection among devices in same VLAN.
No
No
Create VLAN in VLAN database.
Assign port to correct VLAN.

Figure 22 Missing VLAN Flowchart

- As shown in Figure 22, use the **show vlan** command to check whether the port belongs to the expected VLAN. If the port is assigned to the wrong VLAN, use the **switchport access vlan** command to correct the VLAN membership on a particular port. Use the **show mac address-table** command to check which addresses were learned on a particular port of the switch and to which VLAN that port is assigned, as shown in the following output.

```
S1# show mac address-table interface fastethernet 0/1
Mac Address Table
--------------------------------------
Vlan        Mac Address        Type        Ports
---- -------------- ------- -----
10          000c.296a.a21c     DYNAMIC     Fa0/1
10          000f.34f9.9181     DYNAMIC    Fa0/1
Total MAC addresses for this criterion: 2
```

- **Total Mac Addresses for this criterion:** If the VLAN to which the port is assigned is deleted, the port becomes inactive. Use the **show vlan** or **show interfaces switchport** command to verify whether a VLAN is active.

```
S1# show interfaces fastethernet0/1 switchport
Name: Fa0/1
```

```
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 10 (Inactive)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
<output omitted>
```

In the previous example of a MAC address table, the output shows the MAC addresses that were learned on the F0/1 interface. It can be seen that MAC address 000c.296a.a21c was learned on interface F0/1 in VLAN 10. If this number is not the expected VLAN number, change the port VLAN membership using the **switchport access vlan** command.

Note

Each port in a switch belongs to a VLAN. If the VLAN to which the port belongs is deleted, the port becomes inactive. All ports belonging to the VLAN that was deleted are unable to communicate with the rest of the network. Use the **show interface f0/1 switchport** command to check whether the port is inactive. If the port is inactive, it is not functional until the missing VLAN is created using the **vlan** *vlan_id* command.

Introduction to Troubleshooting Trunks

A common task of a network administrator is to troubleshoot trunk link formation or links incorrectly behaving as trunk links. Sometimes a switch port may behave like a trunk port even if it is not configured as a trunk port. For example, an access port might accept frames from VLANs different from the VLAN to which it is assigned. This is called ***VLAN leaking***, which is caused by a mismatched native VLAN or misconfigured trunk.

Figure 3-23 displays a flowchart of general trunk troubleshooting guidelines.

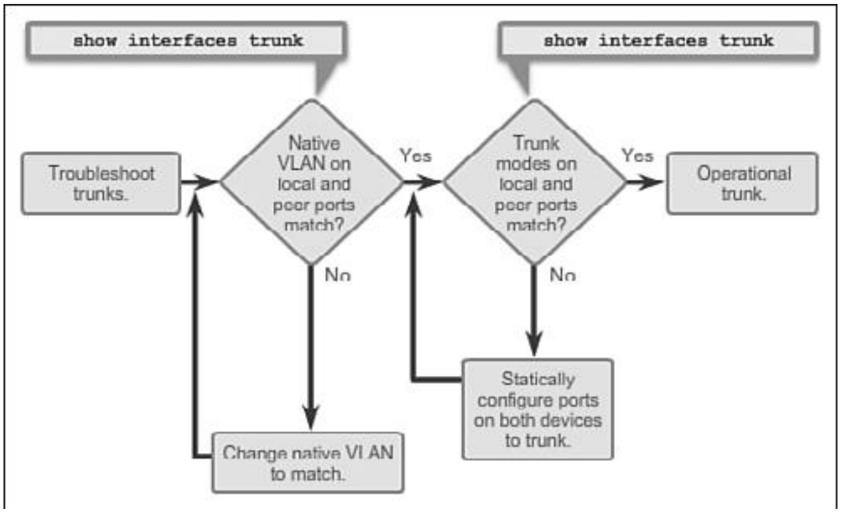

show interfaces trunk
show interfaces trunk
Troubleshoot trunks.
Native VLAN on local and peer ports match?
Yes
Trunk modes on local and peer ports match?
Yes
Operational trunk.
No
No
Statically configure ports on both devices to trunk.
Change native VLAN to match.

Figure 3-23 Trunk Troubleshooting Flowchart

To troubleshoot issues when a trunk is not forming or when VLAN leaking is occurring, proceed as follows:

- Use the **show interfaces trunk** command to check whether the local and peer native VLANs match. If the native VLAN does not match on both sides, VLAN leaking occurs.

- Use the **show interfaces trunk** command to check whether a trunk has been established between switches. Statically configure trunk links whenever possible. Cisco Catalyst switch ports use DTP by default and attempt to negotiate a trunk link.

To display the status of the trunk, determine the native VLAN used on that trunk link and verify trunk establishment using the **show interfaces trunk** command. The following output shows that the native VLAN on one side of the trunk link was changed to VLAN 2. If one end of the trunk is configured as native VLAN 99 and the other end is configured as native VLAN 2, a frame sent from VLAN 99 on one side is received on VLAN 2 on the other side. VLAN 99 leaks into the VLAN 2 segment .

```
SW1# show interfaces f0/1 trunk

Port        Mode      Encapsulation     Status      Native vlan
Fa0/1       auto      802.1q            trunking    2
<output omitted>
```

CDP displays a notification of a native VLAN mismatch on a trunk link with this message:

```
*Mar 1 06:45:26.232: %CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch
discovered on FastEthernet0/1 (2), with S2 FastEthernet0/1 (99).
```

Connectivity issues occur in the network if a native VLAN mismatch exists. Data traffic for VLANs, other than the two native VLANs configured, successfully propagates across the trunk link, but data associated with either of the native VLANs does not successfully propagate across the trunk link.

Note

The previous output indicates that there is an active trunk despite the native VLAN mismatch. Configure the native VLAN to be the same VLAN on both sides of the link to correct this behavior so that VLAN leaking does not occur.

Common Problems with Trunks

Trunking issues are usually associated with incorrect configurations, shown in Table 3-9.

Table 3-9 Common Problems with Trunks

Problem	Result	Example
Native VLAN mismatch	Poses a security risk and creates unintended results	One port is defined as native VLAN 99 and the opposite trunk end is defined as native VLAN 100.
Trunk mode mismatch	Causes loss of network connectivity	One end of the trunk is configured as trunk mode “off” and the other as trunk mode “on.”
Allowed VLANs on trunks	Causes unexpected traffic or no traffic to be sent over the trunk	The list of allowed VLANs does not support current VLAN trunking requirements.

When configuring VLANs and trunks on a switched infrastructure, the following types of configuration errors are the most common:

- **Native VLAN mismatches:** Trunk ports are configured with different native VLANs. This configuration error generates console notifications, and causes control and management traffic to be misdirected. This poses a security risk.

- **Trunk mode mismatches:** One trunk port is configured with trunk mode off and the other with trunk mode on. This configuration error causes the trunk link to stop working.

- **Allowed VLANs on trunks:** The list of allowed VLANs on a trunk has not been updated with the current VLAN trunking requirements. In this situation, unexpected traffic or no traffic is sent over the trunk. If an issue with a trunk is discovered and if the cause is unknown, start troubleshooting by examining the trunks for a native VLAN mismatch. If that is not the cause, check for trunk mode mismatches, and finally check for the allowed VLAN list on the trunk. The next two sections examine how to fix the common problems with trunks.

Trunk Mode Mismatches

Trunk links are normally configured statically with the **switchport mode trunk** command. Cisco Catalyst switch trunk ports use DTP to negotiate the state of the link. When a port on a trunk link is configured with a trunk mode that is incompatible with the neighboring trunk port, a trunk link fails to form between the two switches. In Figure 3-24, PC4 cannot connect to the internal web server. The topology indicates a valid configuration. Why is there a problem?

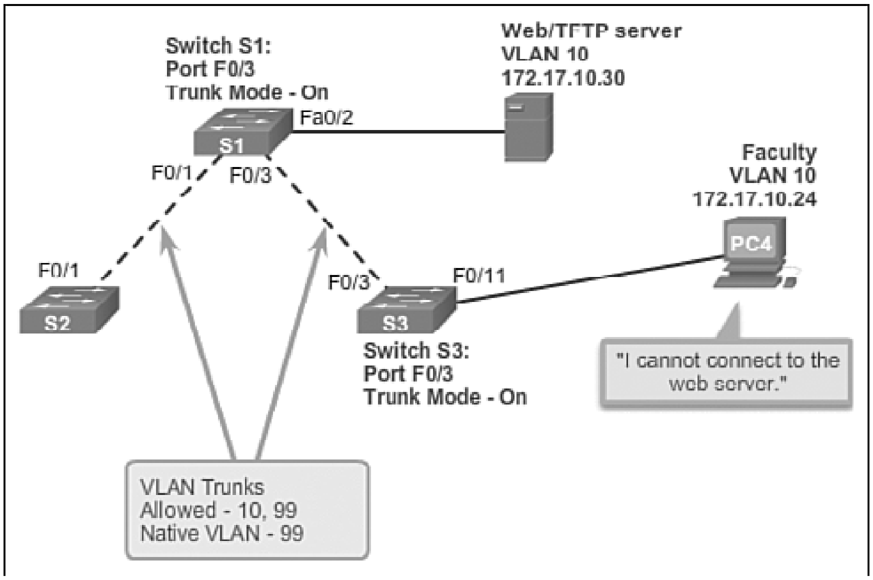


Figure 3-24 Trunk Scenario Topology

Check the status of the trunk ports on switch S1 using the **show interfaces trunk** command. The following output reveals that interface Fa0/3 on switch S1 is not currently a trunk link. Examining the F0/3 interface reveals that the switch port is actually in dynamic auto mode.

Output from Switch S1:

```
S1# show interfaces trunk
Port Mode Encapsulation Status Native vlan
Fa0/1 on 802.1q trunking 99
Port Vlans allowed on trunk
Fa0/1 10,99
Port Vlans allowed and active in management domain
Fa0/1 10,99
Port Vlans in spanning tree forwarding state and not pruned
Fa0/1 10,99

S1# show interfaces f0/3 switchport
Name: Fa0/3
Switchport: Enabled
Administrative Mode: dynamic auto
<output omitted>
```

An examination of the trunks on switch S3 reveals that there are no active trunk ports. Further checking reveals that the Fa0/3 interface is also in dynamic auto mode. This explains why the trunk is down as shown in the output.

Output from Switch S3:

```
S3# show interfaces trunk

S3#
S3# show interfaces f0/3 switchport
Name: Fa0/3
Switchport: Enabled
```

```
Administrative Mode: dynamic auto
<output omitted>
```

To resolve the issue, reconfigure the trunk mode of the F0/3 ports on switches S1 and S3, as shown in the following output. After the configuration change, the output of the **show interfaces** command indicates that the port on switch S1 is now in trunking mode. The output from PC4 indicates that it has regained connectivity to the Web/TFTP server found at IP address 172.17.10.30.

Output from Switch S1:

```
S1# config terminal
S1(config)# interface fastethernet0/3

S1(config-if)# switchport mode trunk
S1(config-if)# end
S1# show interfaces fa0/3 switchport
Name: Fa0/3
Switchport: Enabled
Administrative Mode: trunk
<output omitted>
```

Output from Switch S3:

```
S3# config terminal
S3(config)# interface fastethernet0/3
S3(config-if)# switchport mode trunk
S3(config-if)# end
S3# show interfaces fa0/3 switchport
Name: Fa0/3
Switchport: Enabled
Administrative Mode: trunk
<output omitted>
S3# show interfaces trunk
Port Mode Encapsulation Status Native vlan
```

```
Fa0/3 on 802.1q trunking 99
Port Vlans allowed on trunk
Fa0/3 10,99
Port Vlans allowed and active in management domain
Fa0/3 10,99
Port Vlans in spanning tree forwarding state and not pruned
Fa0/3 10,99
```

Output from Computer PC4:

```
Pc4> ping 172.17.10.30
Pinging 172.17.10.30 with 32 bytes of data:
Reply from 172.17.10.30: bytes=32 time=147ms TTL=128
<output omitted>
```

Incorrect VLAN List

For traffic from a VLAN to be transmitted across a trunk, it must be allowed on the trunk. To do so, use the **switchport trunk allowed vlan** *vlan-id* command. In Figure 3-25, VLAN 20 (Student) and PC5 have been added to the network. The documentation has been updated to show that the VLANs allowed on the trunk are 10, 20, and 99. In this scenario, PC5 cannot connect to the student email server.

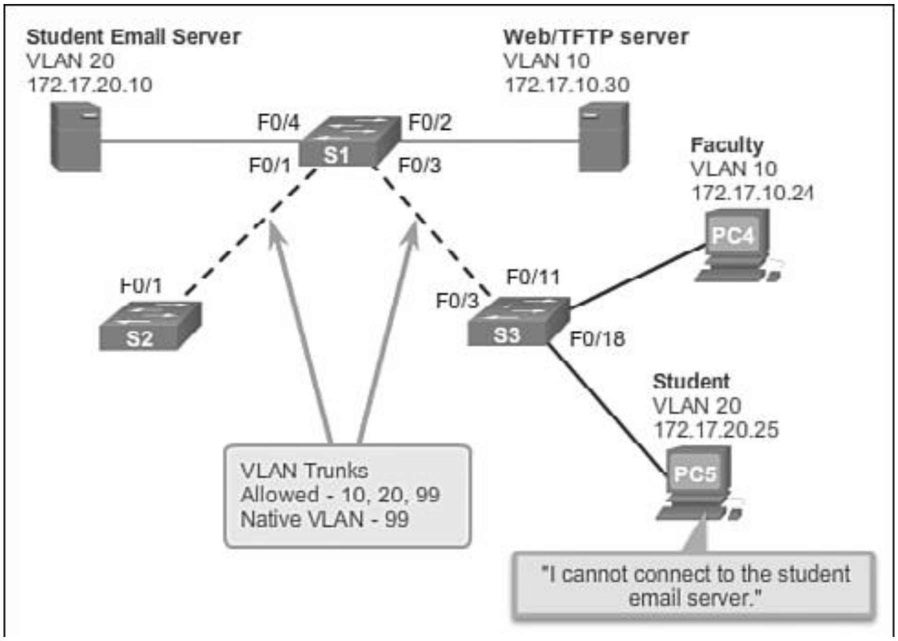


Figure 3-25 Incorrect VLAN List Scenario Topology

Check the trunk ports on switch S3 using the **show interfaces trunk** command as shown in the output that follows.

Output from Switch S3:

```
S3# show interfaces trunk
Port Mode Encapsulation Status Native vlan
Fa0/3 on 802.1q trunking 99
Port Vlans allowed on trunk
Fa0/3 10,20,99
Port Vlans allowed and active in management domain
Fa0/3 10,20,99
Port Vlans in spanning tree forwarding state and not pruned
Fa0/3 10,20,99
```

The command reveals that the interface F0/3 on switch S3 is correctly configured to allow VLANs 10, 20, and 99 as shown in the output.

An examination of the F0/3 interface on switch S1 reveals that interfaces F0/1 and F0/3 allow only VLANs 10 and 99. Someone updated the documentation but forgot to reconfigure the ports on the S1 switch, as shown in the output.

Output from Switch S1:

```
S1# show interfaces trunk
Port Mode Encapsulation Status Native vlan
Fa0/1 on 802.1q
Fa0/3 on 802.1q trunking 99
Port Vlans allowed on trunk
Fa0/1 10,99
Fa0/3 10,99
<output omitted>
```

Reconfigure F0/1 and F0/3 on switch S1 using the **switchport trunk allowed vlan 10,20,99** command as shown in the following output. The output shows that VLANs 10, 20, and 99 are now added to the F0/1 and F0/3 ports on switch S1. The **show interfaces trunk** command is an excellent tool for revealing common trunking problems.

Output from Switch S1:

```
S1# config terminal
S1(config)# interface f0/1
S1(config-if)# switchport trunk allowed vlan 10,20,99
S1(config-if)# interface f0/3
S1(config-if)# switchport trunk allowed vlan 10,20,99
S1# show interface trunk
Port Mode Encapsulation Status Native vlan
Fa0/1 on 802.1q
Fa0/3 on 802.1q trunking 99
Port Vlans allowed on trunk
Fa0/1 10,20,99
Fa0/3 10,20,99
<output omitted>
```

PC5 has regained connectivity to the student email server found at IP address 172.17.20.10.

Output from Computer PC5:

```
PC5> ping 172.17.20.10
Pinging 172.17.20.10 with 32 bytes of data:
Reply from 172.17.20.10: bytes=32 time=147ms TTL=128
<output omitted>
```

VLAN Security and Design (3.3)

Learning what attacks can occur and how to design the switch network to mitigate these attacks is important to a network technician. Because VLANs are commonly configured in a business environment, VLANs are a common security target.

Switch Spoofing Attack

There are a number of different types of VLAN attacks in modern switched networks. The VLAN architecture simplifies network maintenance and improves performance, but it also opens the door to abuse. It is important to understand the general methodology behind these attacks and the primary approaches to mitigate them.

VLAN hopping enables traffic from one VLAN to be seen by another VLAN.

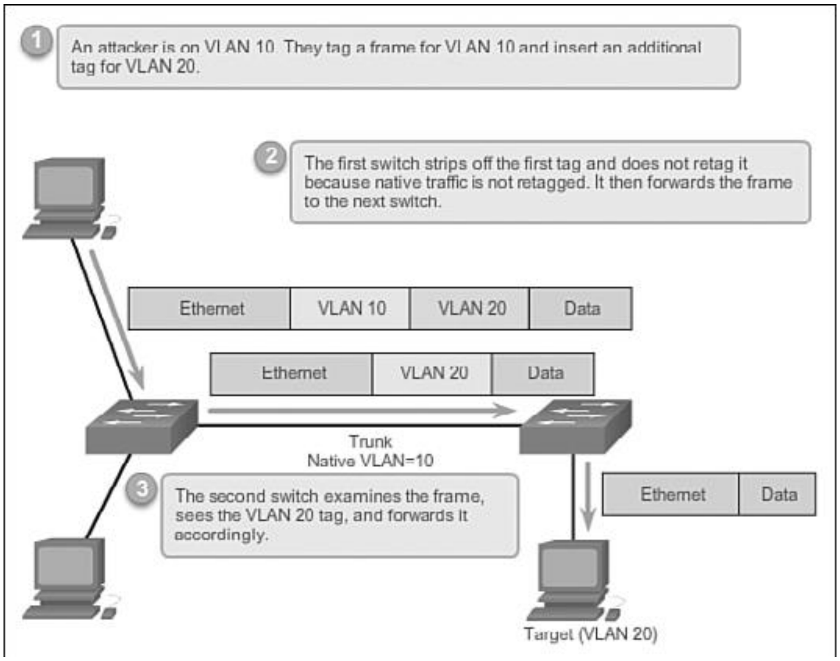
Switch spoofing is a type of VLAN hopping attack that works by taking advantage of an incorrectly configured trunk port. By default, trunk ports have access to all VLANs and pass traffic for multiple VLANs across the same physical link, generally between switches.

In a basic switch spoofing attack, the attacker takes advantage of the fact that the default configuration of the switch port is dynamic auto. The network attacker configures a system to spoof itself as a switch. This spoofing requires that the network attacker be capable of emulating 802.1Q and DTP messages. By tricking a switch into thinking that another switch is attempting to form a trunk, an attacker can gain access to all the VLANs allowed on the trunk port.

The best way to prevent a basic switch spoofing attack is to turn off trunking on all ports, except the ones that specifically require trunking. On the required trunking ports, disable DTP, and manually enable trunking.

Double-Tagging Attack

Another type of VLAN attack is a ***double-tagging*** (or double-encapsulated) VLAN hopping attack. This type of attack takes advantage of the way that hardware on most switches operates. Most switches perform only one level of 802.1Q de- encapsulation, which allows an attacker to embed a hidden 802.1Q tag inside the frame. This tag allows the frame to be forwarded to a VLAN that the original 802.1Q tag did not specify as shown in Figure 3-26. An important characteristic of the double-encapsulated VLAN hopping attack is that it works even if trunk ports are disabled, because a host typically sends a frame on a segment that is not a trunk link.



1 An attacker is on VLAN 10. They tag a frame for VLAN 10 and insert an additional tag for VLAN 20.

2 The first switch strips off the first tag and does not retag it because native traffic is not retagged. It then forwards the frame to the next switch.

3 The second switch examines the frame, sees the VLAN 20 tag, and forwards it accordingly.

Figure 3-26 Double-Tagging Attack

A double-tagging VLAN hopping attack follows three steps:

Step 1. The attacker sends a double-tagged 802.1Q frame to the switch. The outer header has the VLAN tag of the attacker, which is the same as the native VLAN of the trunk port. The assumption is that the switch processes the frame received from the attacker as if it were on a trunk port or a port with a voice VLAN. (A switch should not receive a tagged Ethernet frame on an access port.) For the purposes of this example, assume that the native VLAN is VLAN 10. The inner tag is the victim VLAN; in this case, it is VLAN 20.

Step 2. The frame arrives on the switch, which looks at the first 4-byte 802.1Q tag. The switch sees that the frame is destined for VLAN 10, which is the native VLAN. The switch forwards the packet out on all VLAN 10 ports after stripping the VLAN 10 tag. On the trunk port, the VLAN 10 tag is stripped, and the packet is not retagged because it is part of the native VLAN. At this point, the VLAN 20 tag is still intact and has not been inspected by the first switch.

Step 3. The second switch looks only at the inner 802.1Q tag that the attacker sent and sees that the frame is destined for VLAN 20, the target VLAN. The second switch sends the frame on to the victim port or floods it, depending on whether there is an existing MAC address table entry for the victim host.

This type of attack is unidirectional and works only when the attacker is connected to a port residing in the same VLAN as the native VLAN of the trunk port. Thwarting this type of attack is not as easy as stopping basic VLAN hopping attacks. The best approach to mitigating double-tagging attacks is to ensure that the native VLAN of the trunk ports is different from the VLAN of any user ports. In fact, it is considered a security best practice to use a fixed VLAN that is distinct from all user VLANs in the switched network as the native VLAN for all 802.1Q trunks. PVLAN Edge (3.3.1.3) Some applications require that no traffic be forwarded at Layer 2 between ports on

the same switch so that one neighbor does not see the traffic generated by anotherneighbor. In such an environment, the use of the ***Private VLAN (PVLAN) Edge*** feature, also known as protected ports, ensures that there is no exchange of unicast, broadcast, or multicast traffic between these ports on the switch, as shown in Figure 3-27.

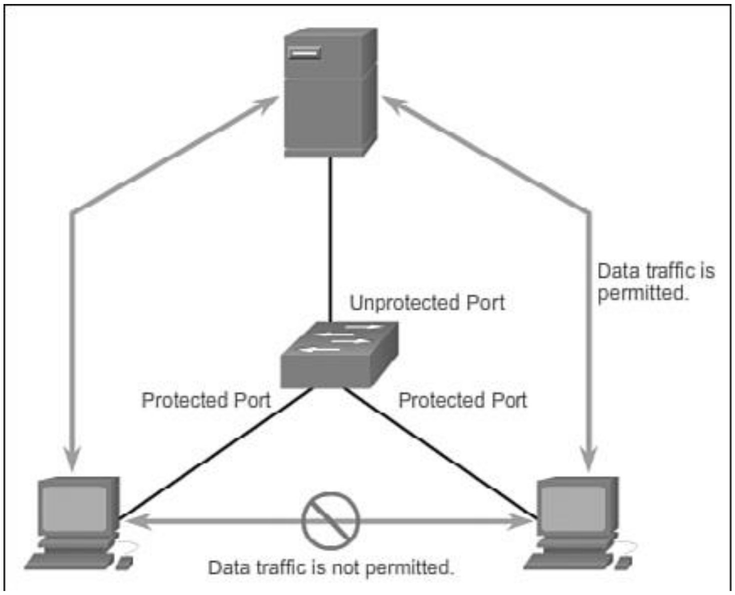


Figure 3-27 PVLAN Edge

The PVLAN Edge feature has the following characteristics:

- A protected port does not forward any traffic (unicast, multicast, or broadcast) to any other port that is also a protected port, except for control traffic. Data traffic cannot be forwarded between protected ports at Layer 2.
- Forwarding behavior between a protected port and a nonprotected port proceeds as usual.
- Protected ports must be manually configured.

To configure the PVLAN Edge feature, enter the **switchport protected** command in interface configuration mode as shown in the output that follows.

```
S1(config)# interface g0/1
S1(config-if)# switchport protected
S1(config-if)# end
S1# show interfaces g0/1 switchport
Name: G0/1
Switchport: Enabled
Administrative Mode: dynamic auto
Operational Mode: down
Administrative 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
<output omitted>

Protected: true
Unknown unicast blocked: disabled
Unknown multicast blocked: disabled
Appliance trust: none
```

To disable protected port, use the **no switchport protected** interface configuration mode command. To verify the configuration of the PVLAN Edge feature, use the **show interfaces** *interface-id* **switchport** global configuration mode command.

Design Best Practices for VLANs

Here are some best practices to use before you create the first VLAN on a switch.

Shut down unused switch ports to prevent unauthorized access

The default Ethernet VLAN is VLAN 1. It is a security best practice to configure all the ports on all switches to be associated with VLANs other than VLAN 1. This is usually done by configuring all unused ports to a ***black hole VLAN*** that is not used for anything on the network. All used ports are associated with VLANs distinct from VLAN 1 and distinct from the black hole VLAN.

Separate management and user data traffic.

The management VLAN, which is VLAN 1 by default, should be changed to a separate, distinct VLAN. To communicate remotely with a Cisco switch for management purposes, the switch must have an IP address configured on the management VLAN. Users in other VLANs would not be able to establish remote access sessions to the switch unless they were routed into the management VLAN, providing an additional layer of security. Also, the switch should be configured to accept only encrypted SSH sessions for remote management.

Change the native VLAN to a different VLAN than VLAN 1

All control traffic is sent on VLAN 1. Therefore, when the native VLAN is changed to something other than VLAN 1, all control traffic is tagged on IEEE 802.1Q VLAN trunks (tagged with VLAN ID 1). A recommended security practice is to change the native VLAN to a different VLAN than VLAN 1. The native VLAN should also be distinct from all user VLANs. Ensure that the native VLAN for an 802.1Q trunk is the same on both ends of the trunk link.

Do not use the dynamic auto or dynamic desirable switch port modes

DTP offers four switch port modes: access, trunk, dynamic auto, and dynamic desirable. A general guideline is to disable autonegotiation. As a port security best practice, do not use the dynamic auto or dynamic desirable switch port modes.

Separate VLANs for IP telephony and data traffic

Voice traffic has stringent QoS requirements. If user PCs and IP phones are on the same VLAN, each tries to use the available bandwidth without considering the other device. To avoid this conflict, it is good practice to use separate VLANs for IP telephony and data traffic.