REPUBLIC OF THE PHILIPPINES
POLYTECHNIC UNIVERSITY OF THE PHILIPPINES
STA. MESA, MANILA
COLLEGE OF ENGINEERING
COMPUTER ENGINEERING DEPARTMENT


**CMPE 30174**

# COMPUTER NETWORKS AND SECURITY

## INSTRUCTIONAL MATERIAL

***ENGR. ORLANDO PAJABERA***

## Module 1: Basic Device Configuration

### *Objectives:*

At the end of this module, the student should be able to:

- Configure initial settings on a Cisco switch;
- Configure switch ports to meet network requirements;
- Configure secure management access on a switch;
- Configure basic settings on a router to route between two directly-connected networks, using CLI;
- Verify connectivity between two networks that are directly connected to a router.

### Configure a Switch with Initial Settings

#### *Switch Boot Sequence*

After a Cisco switch is powered on, it goes through the following five-step boot sequence:

**Step 1:** First, the switch loads a power-on self-test (POST) program stored in ROM. POST checks the CPU subsystem. It tests the CPU, DRAM and the portion of the flash device that makes up the flash file system.

**Step 2:** Next, the switch loads the boot loader software. The boot loader is a small program stored in ROM that is run immediately after POST successfully completes.

**Step 3:** The boot loader performs low-level CPU initialization. It initializes the CPU registers, which control where physical memory is mapped, the quantity of memory, and its speed.

**Step 4:** The boot loader initializes the flash file system on the system board.

**Step 5:** Finally, the boot loader locates and loads a default IOS operating system software image into memory and gives control of the switch over to the IOS.

#### *The Boot System Command*

- The switch attempts to automatically boot by using information in the BOOT environment variable. If this variable is not set, the switch attempts to load and execute the first executable file it can find.
- The IOS operating system then initializes the interfaces using the Cisco IOS commands found in the startup-config file. The startup-config file is called **config.text** and is located in flash.
- In the example, the BOOT environment variable is set using the **boot system** global configuration mode command. Notice that the IOS is located in a distinct folder and the folder path is specified. Use the command **show boot** to see what the current IOS boot file is set to.

```
S1(config)# boot system flash:/c2960-lanbasek9-mz.150-2.SE/c2960-lanbasek9-mz.150-2.SE.bin
```

| Command | Definition |
|---|---|
| **boot system** | The main command |
| **flash:** | The storage device |
| **c2960-lanbasek9-mz.150-2.SE/** | The path to the file system |
| **c2960-lanbasek9-mz.150-2.SE.bin** | The IOS file name |

***Switch LED Indicators***


**System LED (SYST):** Shows whether the system is receiving power and functioning properly.
**Redundant Power Supply LED (RPS):** Shows the RPS status.
**Port Status LED (STAT):** When green, indicates port status mode is selected, which is the default. Port status can then be understood by the light associated with each port.
**Port Duplex LED (DUPLX):** When green, indicates port duplex mode is selected. Port duplex can then be understood by the light associated with each port.
**Port Speed LED (SPEED):** When green, indicates port speed mode is selected. Port speed can then be understood by the light associated with each port.
**Power over Ethernet LED (PoE):** Present if the switch supports PoE. Indicates the PoE status of ports on the switch.
The Mode button is used to move between the different modes – STAT, DUPLX, SPEED, and PoE

| | Off | Green | Blinking Green | Amber | Blinking Amber | Alternating Green/Amber |
|---|---|---|---|---|---|---|
| RPS | Off/No RPS | RPS ready | RPS up but not available | RPS standby or fault | Internal PS failed, RPS providing power | N/A |
| PoE | Not selected, no issues | Selected | N/A | N/A | Not selected, port issues present | N/A |
| When the named mode is selected, the light associated with each physical port indicates: | | | | | | |
| STAT | No link or shutdown | Link Up | Activity | Port blocked preventing loop | Port blocked preventing loop | Link fault |
| DUPLEX | Half-duplex | Full-duplex | N/A | N/A | N/A | N/A |
| SPEED | 10Mbps | 100Mbps | 1000Mbps | N/A | N/A | N/A |
| PoE | PoE off | PoE on | N/A | PoE disabled | PoE off due to fault | PoE denied (over budget) |

### *Recovering From a System Crash*

The boot loader provides access into the switch if the operating system cannot be used because of missing or damaged system files. The boot loader has a command line that provides access to the files stored in flash memory. The boot loader can be accessed through a console connection following these steps:

**Step 1**. Connect a PC by console cable to the switch console port. Configure terminal emulation software to connect to the switch.

**Step 2**. Unplug the switch power cord.

**Step 3**. Reconnect the power cord to the switch and, within 15 seconds, press and hold down the **Mode** button while the System LED is still flashing green.

**Step 4**. Continue pressing the **Mode** button until the System LED turns briefly amber and then solid green; then release the **Mode** button.

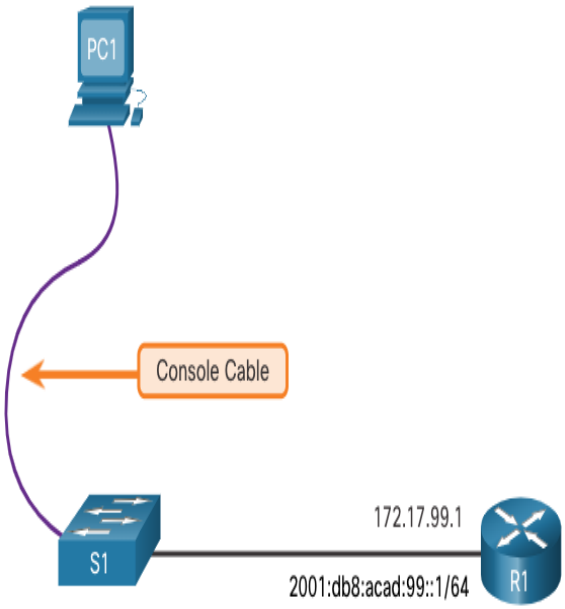
**Step 5**. The boot loader **switch:** prompt appears in the terminal emulation software on the PC.

The boot loader command line supports commands to format the flash file system, reinstall the operating system software, and recover a lost or forgotten password. For example, the **dir** command can be used to view a list of files within a specified directory.

### *Switch Management Access*

To prepare a switch for remote management access, the switch must be configured with an IP address and a subnet mask.

- To manage the switch from a remote network, the switch must be configured with a default gateway. This is very similar to configuring the IP address information on host devices.
- In the figure, the switch virtual interface (SVI) on S1 should be assigned an IP address. The SVI is a virtual interface, not a physical port on the switch. A console cable is used to connect to a PC so that the switch can be initially configured.



### *Switch SVI Configuration Example*

By default, the switch is configured to have its management controlled through VLAN 1. All ports are assigned to VLAN 1 by default. For security purposes, it is considered a best practice to use a VLAN other than VLAN 1 for the management VLAN,

**Step 1**: **Configure the Management Interface:** From VLAN interface configuration mode, an IPv4 address and subnet mask is applied to the management SVI of the switch.

**Note**: The SVI for VLAN 99 will not appear as “up/up” until VLAN 99 is created and there is a device connected to a switch port associated with VLAN 99.

**Note**: The switch may need to be configured for IPv6. For example, before you can configure IPv6 addressing on a Cisco Catalyst 2960 running IOS version 15.0, you will need to enter the global configuration command **sdm prefer dual-ipv4-and-ipv6 default** and then **reload** the switch.

| Task | IOS Commands |
|---|---|
| Enter global configuration mode. | S1# **configure terminal** |
| Enter interface configuration mode for the SVI. | S1(config)# **interface vlan 99** |
| Configure the management interface IPv4 address. | S1(config-if)# **ip address 172.17.99.11 255.255.255.0** |
| Configure the management interface IPv6 address | S1(config-if)# **ipv6 address 2001:db8:acad:99::1/64** |
| Enable the management interface. | S1(config-if)# **no shutdown** |
| Return to the privileged EXEC mode. | S1(config-if)# **end** |
| Save the running config to the startup config. | S1# **copy running-config startup-config** |

### Step 2: Configure the Default Gateway

- The switch should be configured with a default gateway if it will be managed remotely from networks that are not directly connected.
  - **Note**: Because, it will receive its default gateway information from a router advertisement (RA) message, the switch does not require an IPv6 default gateway.

| Task | IOS Commands |
|---|---|
| Enter global configuration mode. | S1# **configure terminal** |
| Configure the default gateway for the switch. | S1(config)# **ip default-gateway 172.17.99.1** |
| Return to the privileged EXEC mode. | S1(config-if)# **end** |
| Save the running config to the startup config. | S1# **copy running-config startup-config** |

**Step 3**: **Verify Configuration**

- The **show ip interface brief** and **show ipv6 interface brief** commands are useful for determining the status of both physical and virtual interfaces. The output shown confirms that interface VLAN 99 has been configured with an IPv4 and IPv6 address.

**Note**: An IP address applied to the SVI is only for remote management access to the switch; this does not allow the switch to route Layer 3 packets.

```
S1# show ip interface brief
Interface      IP-Address      OK? Method   Status     Protocol
Vlan99         172.17.99.11    YES manual   down       down
(output omitted)
S1# show ipv6 interface brief
Vlan99                  [down/down]
    FE80::C27B:BCFF:FEC4:A9C1
    2001:DB8:ACAD:99::1
(output omitted)
```

**Configure Switch Ports**

***Duplex Communication***

- Full-duplex communication increases bandwidth efficiency by allowing both ends of a connection to transmit and receive data simultaneously. This is also known as bidirectional communication and it requires microsegmentation.
- A microsegmented LAN is created when a switch port has only one device connected and is operating in full-duplex mode. There is no collision domain associated with a switch port operating in full-duplex mode.
- Unlike full-duplex communication, half-duplex communication is unidirectional. Half-duplex communication creates performance issues because data can flow in only one direction at a time, often resulting in collisions.
- Gigabit Ethernet and 10 Gb NICs require full-duplex connections to operate. In full-duplex mode, the collision detection circuit on the NIC is disabled. Full-duplex offers 100 percent efficiency in both directions (transmitting and receiving). This results in a doubling of the potential use of the stated bandwidth.

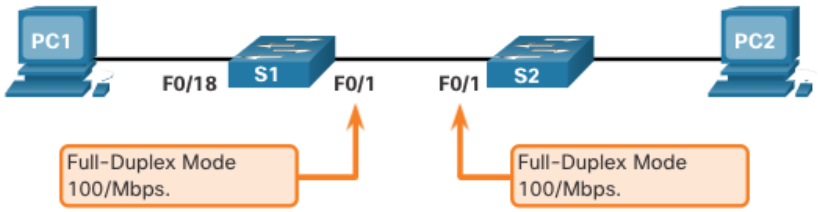
***Configure Switch Ports at the Physical Layer***

- Switch ports can be manually configured with specific duplex and speed settings. The respective interface configuration commands are **duplex** and **speed**.
- The default setting for both duplex and speed for switch ports on Cisco Catalyst 2960 and 3560 switches is auto. The 10/100/1000 ports operate in either half- or full-duplex mode when they are set to 10 or 100 Mbps and operate only in full-duplex mode when it is set to 1000 Mbps (1 Gbps).
- Autonegotiation is useful when the speed and duplex settings of the device connecting to the port are unknown or may change. When connecting to known devices such as servers, dedicated workstations, or network devices, a best practice is to manually set the speed and duplex settings.

- When troubleshooting switch port issues, it is important that the duplex and speed settings are checked.

**Note**: Mismatched settings for the duplex mode and speed of switch ports can cause connectivity issues. Autonegotiation failure creates mismatched settings.

All fiber-optic ports, such as 1000BASE-SX ports, operate only at one preset speed and are always full-duplex




| Task | IOS Commands |
|---|---|
| Enter global configuration mode. | S1# **configure terminal** |
| Enter interface configuration mode. | S1(config)# **interface FastEthernet 0/1** |
| Configure the interface duplex. | S1(config-if)# **duplex full** |
| Configure the interface speed. | S1(config-if)# **speed 100** |
| Return to the privileged EXEC mode. | S1(config-if)# **end** |
| Save the running config to the startup config. | S1# **copy running-config startup-config** |

***Auto-MDIX***

- When automatic medium-dependent interface crossover (auto-MDIX) is enabled, the switch interface automatically detects the required cable connection type (straight-through or crossover) and configures the connection appropriately.
- When connecting to switches without the auto-MDIX feature, straight-through cables must be used to connect to devices such as servers, workstations, or routers. Crossover cables must be used to connect to other switches or repeaters.
- With auto-MDIX enabled, either type of cable can be used to connect to other devices, and the interface automatically adjusts to communicate successfully.
- On newer Cisco switches, the **mdix auto** interface configuration mode command enables the feature. When using auto-MDIX on an interface, the interface speed and duplex must be set to **auto** so that the feature operates correctly.

**Note**: The auto-MDIX feature is enabled by default on Catalyst 2960 and Catalyst 3560 switches but is not available on the older Catalyst 2950 and Catalyst 3550 switches.

To examine the auto-MDIX setting for a specific interface, use the **show controllers ethernet-controller** command with the **phy** keyword. To limit the output to lines referencing auto-MDIX, use the **include Auto-MDIX** filter.

### *Switch Verification Commands*

| Task | IOS Commands |
|---|---|
| Display interface status and configuration. | S1# **show interfaces** [*interface-id*] |
| Display current startup configuration. | S1# **show startup-config** |
| Display current running configuration. | S1# **show running-config** |
| Display information about flash file system. | S1# **show flash** |
| Display system hardware and software status. | S1# **show version** |
| Display history of command entered. | S1# **show history** |
| Display IP information about an interface. | S1# **show ip interface** [*interface-id*]<br>OR<br>S1# **show ipv6 interface** [*interface-id*] |
| Display the MAC address table. | S1# **show mac-address-table**<br>OR<br>S1# **show mac address-table** |

### *Verify Switch Port Configuration*

The **show running-config** command can be used to verify that the switch has been correctly configured. From the sample abbreviated output on S1, some important information is shown in the figure:

- Fast Ethernet 0/18 interface configured with the management VLAN 99
- VLAN 99 configured with an IPv4 address of 172.17.99.11 255.255.255.0
- Default gateway set to 172.17.99.1

```
S1# show running-config
Building configuration...
Current configuration : 1466 bytes
!
(output omitted)
interface Vlan99
 ip address 172.17.99.11 255.255.255.0
 ipv6 address 2001:DB8:ACAD:99::1/64
!
ip default-gateway 172.17.99.1
```

The **show interfaces** command is another commonly used command, which displays status and statistics information on the network interfaces of the switch. The **show interfaces** command is frequently used when configuring and monitoring network devices.

The first line of the output for the **show interfaces fastEthernet 0/18** command indicates that the FastEthernet 0/18 interface is up/up, meaning that it is operational. Further down, the output shows that the duplex is full and the speed is 100 Mbps.

```
S1# show interfaces fastEthernet 0/18
FastEthernet0/18 is up, line protocol is up (connected)
  Hardware is Fast Ethernet, address is 0025.83e6.9092 (bia 0025.83e6.9092)
  MTU 1500 bytes, BW 100000 Kbit/sec, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, media type is 10/100BaseTX
```

### *Network Access Layer Issues*

The output from the **show interfaces** command is useful for detecting common media issues. One of the most important parts of this output is the display of the line and data link protocol status, as shown in the example.

The first parameter (FastEthernet0/18 is up) refers to the hardware layer and indicates whether the interface is receiving a carrier detect signal. The second parameter (line protocol is up) refers to the data link layer and indicates whether the data link layer protocol keepalives are being received. Based on the output of the **show interfaces** command, possible problems can be fixed as follows:

- If the interface is up and the line protocol is down, a problem exists. There could be an encapsulation type mismatch, the interface on the other end could be error-disabled, or there could be a hardware problem.
- If the line protocol and the interface are both down, a cable is not attached, or some other interface problem exists. For example, in a back-to-back connection, the other end of the connection may be administratively down.
- If the interface is administratively down, it has been manually disabled (the **shutdown** command has been issued) in the active configuration.

```
S1# show interfaces fastEthernet 0/18
FastEthernet0/18 is up, line protocol is up (connected)
Hardware is Fast Ethernet, address is 0025.83e6.9092 (bia 0025.83e6.9092)MTU 1500 bytes, BW
100000 Kbit/sec, DLY 100 usec,
```

The **show interfaces** command output displays counters and statistics for the FastEthernet0/18 interface, as shown here:

```
S1# show interfaces fastEthernet 0/18
FastEthernet0/18 is up, line protocol is up (connected)
  Hardware is Fast Ethernet, address is 0025.83e6.9092 (bia 0025.83e6.9092)
  MTU 1500 bytes, BW 100000 Kbit/sec, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, media type is 10/100BaseTX
  input flow-control is off, output flow-control is unsupported
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output 00:00:01, 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
     2295197 packets input, 305539992 bytes, 0 no buffer
     Received 1925500 broadcasts (74 multicasts)
     0 runts, 0 giants, 0 throttles
     3 input errors, 3 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 74 multicast, 0 pause input
     0 input packets with dribble condition detected
     3594664 packets output, 436549843 bytes, 0 underruns
     8 output errors, 1790 collisions, 10 interface resets
     0 unknown protocol drops
     0 babbles, 235 late collision, 0 deferred
```

Some media errors are not severe enough to cause the circuit to fail but do cause network performance issues. The table explains some of these common errors which can be detected using the **show interfaces** command.

| Error Type | Description |
|---|---|
| **Input Errors** | Total number of errors. It includes runts, giants, no buffer, CRC, frame, overrun, and ignored counts. |
| **Runts** | Packets that are discarded because they are smaller than the minimum packet size for the medium. For instance, any Ethernet packet that is less than 64 bytes is considered a runt. |
| **Giants** | Packets that are discarded because they exceed the maximum packet size for the medium. For example, any Ethernet packet that is greater than 1,518 bytes is considered a giant. |
| **CRC** | CRC errors are generated when the calculated checksum is not the same as the checksum received. |
| **Output Errors** | Sum of all errors that prevented the final transmission of datagrams out of the interface that is being examined. |
| **Collisions** | Number of messages retransmitted because of an Ethernet collision. |
| **Late Collisions** | A collision that occurs after 512 bits of the frame have been transmitted |

### *Interface Input and Output Errors*

"Input errors" is the sum of all errors in datagrams that were received on the interface being examined. This includes runts, giants, CRC, no buffer, frame, overrun, and ignored counts. The reported input errors from the **show interfaces** command include the following:

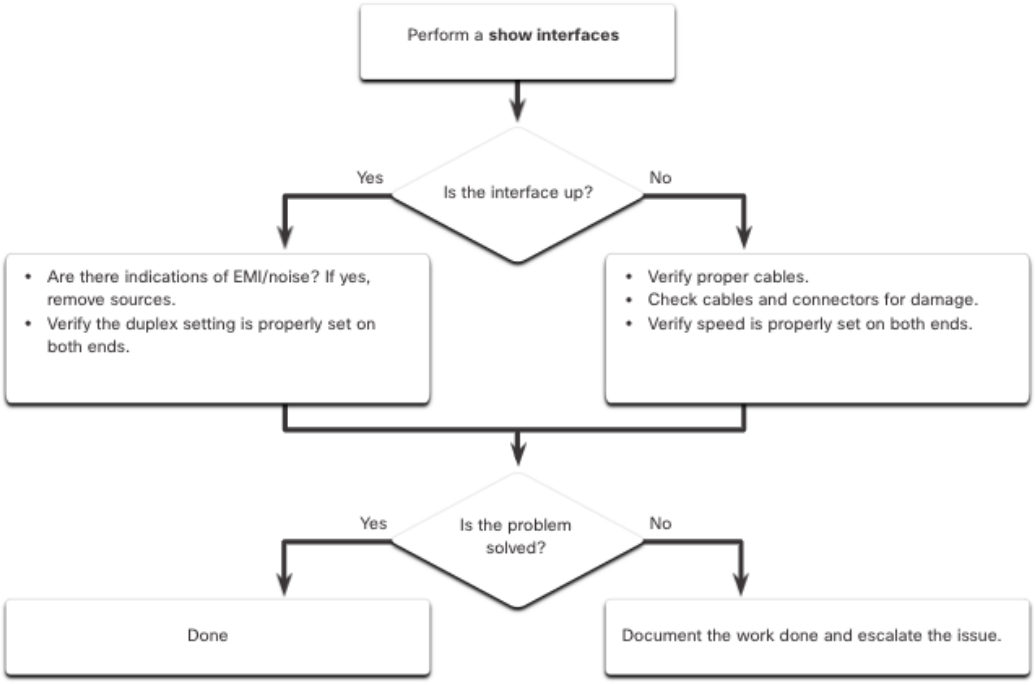
- **Runt Frames** - Ethernet frames that are shorter than the 64-byte minimum allowed length are called runts. Malfunctioning NICs are the usual cause of excessive runt frames, but they can also be caused by collisions.
- **Giants** - Ethernet frames that are larger than the maximum allowed size are called giants.
- **CRC errors** - On Ethernet and serial interfaces, CRC errors usually indicate a media or cable error. Common causes include electrical interference, loose or damaged connections, or incorrect cabling. If you see many CRC errors, there is too much noise on the link and you should inspect the cable. You should also search for and eliminate noise sources.

"Output errors" is the sum of all errors that prevented the final transmission of datagrams out the interface that is being examined. The reported output errors from the **show interfaces** command include the following:

- **Collisions** - Collisions in half-duplex operations are normal. However, you should never see collisions on an interface configured for full-duplex communication.
- **Late collisions** - A late collision refers to a collision that occurs after 512 bits of the frame have been transmitted. Excessive cable lengths are the most common cause of late collisions. Another common cause is duplex misconfiguration.

### *Troubleshooting Network Access Layer Issues*

To troubleshoot scenarios involving no connection, or a bad connection, between a switch and another device, follow the general process shown in the figure.



## Secure Remote Access

### *Telnet Operation*

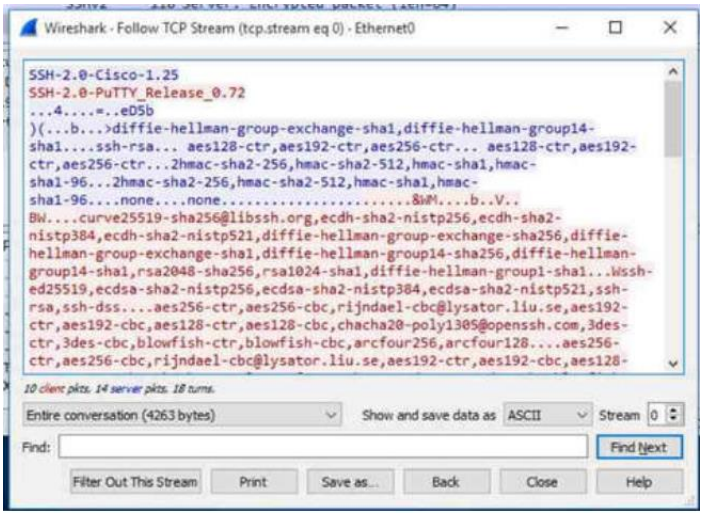
Telnet uses TCP port 23. It is an older protocol that uses unsecure plaintext transmission of both the login authentication (username and password) and the data transmitted between the communicating devices.

A threat actor can monitor packets using Wireshark. For example, in the figure the threat actor captured the username **admin** and password **ccna** from a Telnet session.


### *SSH Operation*

Secure Shell (SSH) is a secure protocol that uses TCP port 22. It provides a secure (encrypted) management connection to a remote device. SSH should replace Telnet for management connections. SSH provides security for remote connections by providing strong encryption when a device is authenticated (username and password) and also for the transmitted data between the communicating devices.

The figure shows a Wireshark capture of an SSH session. The threat actor can track the session using the IP address of the administrator device. However, unlike Telnet, with SSH the username and password are encrypted.



### *Verify the Switch Supports SSH*

To enable SSH on a Catalyst 2960 switch, the switch must be using a version of the IOS software including cryptographic (encrypted) features and capabilities. Use the **show version** command on the switch to see which IOS the switch is currently running. An IOS filename that includes the combination "k9" supports cryptographic (encrypted) features and capabilities.

The example shows the output of the **show version** command.

```
S1# show version
Cisco IOS Software, C2960 Software (C2960-LANBASEK9-M), Version 15.0(2)SE7, RELEASE SOFTWARE
(fc1)
```

### *Configure SSH*

Before configuring SSH, the switch must be minimally configured with a unique hostname and the correct network connectivity settings.

**Step 1**: Verify SSH support - Use the show ip ssh command to verify that the switch supports SSH. If the switch is not running an IOS that supports cryptographic features, this command is unrecognized.

**Step 2**: Configure the IP domain - Configure the IP domain name of the network using the ip domain-name domain-name global configuration mode command.

**Step 3**: Generate RSA key pairs - Generating an RSA key pair automatically enables SSH. Use the crypto key generate rsa global configuration mode command to enable the SSH server on the switch and generate an RSA key pair.

**Note:** To delete the RSA key pair, use the crypto key zeroize rsa global configuration mode command. After the RSA key pair is deleted, the SSH server is automatically disabled.

**Step 4**: Configure user authentication - The SSH server can authenticate users locally or using an authentication server. To use the local authentication method, create a username and password pair using the username username secret password global configuration mode command.

**Step 5**: Configure the vty lines - Enable the SSH protocol on the vty lines by using the transport input ssh line configuration mode command. Use the line vty global configuration mode command and then the login local line configuration mode command to require local authentication for SSH connections from the local username database.

**Step 6:** Enable SSH version 2 - By default, SSH supports both versions 1 and 2. When supporting both versions, this is shown in the show ip ssh output as supporting version 2. Enable SSH version using the ip ssh version 2 global configuration command.

### *Verify SSH is Operational*

On a PC, an SSH client such as PuTTY, is used to connect to an SSH server. For example, assume the following is configured:

- SSH is enabled on switch S1
- Interface VLAN 99 (SVI) with IPv4 address 172.17.99.11 on switch S1
- PC1 with IPv4 address 172.17.99.21

Using a terminal emulator, initiate an SSH connection to the SVI VLAN IPv4 address of S1 from PC1.

When connected, the user is prompted for a username and password as shown in the example. Using the configuration from the previous example, the username **admin** and password **ccna** are entered. After entering the correct combination, the user is connected via SSH to the command line interface (CLI) on the Catalyst 2960 switch.

```
Login as: admin
Using keyboard-interactive
Authentication.
Password:
S1> enable
Password:
S1#
```

To display the version and configuration data for SSH on the device that you configured as an SSH server, use the **show ip ssh** command. In the example, SSH version 2 is enabled.

```
S1# show ip ssh
SSH Enabled - version 2.0
Authentication timeout: 120 secs; Authentication retries: 3
To check the SSH connections to the device, use the show ssh command as shown.
S1# show ssh
%No SSHv1 server connections running.
Connection Version Mode Encryption  Hmac         State                Username
0          2.0  IN   aes256-cbc  hmac-sha1  Session started   admin
0          2.0  OUT  aes256-cbc  hmac-sha1  Session started   admin
S1#
```

## Basic Router Configuration

### *Configure Basic Router Settings*

Cisco routers and Cisco switches have many similarities. They support a similar modal operating system, similar command structures, and many of the same commands. In addition, both devices have similar initial configuration steps. For example, the following configuration tasks should always be performed. Name the device to distinguish it from other routers and configure passwords, as shown in the example.

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# hostname R1
R1(config)# enable secret class
R1(config)# line console 0
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)# service password-encryption
R1(config)#
```

Configure a banner to provide legal notification of unauthorized access, as shown in the example.

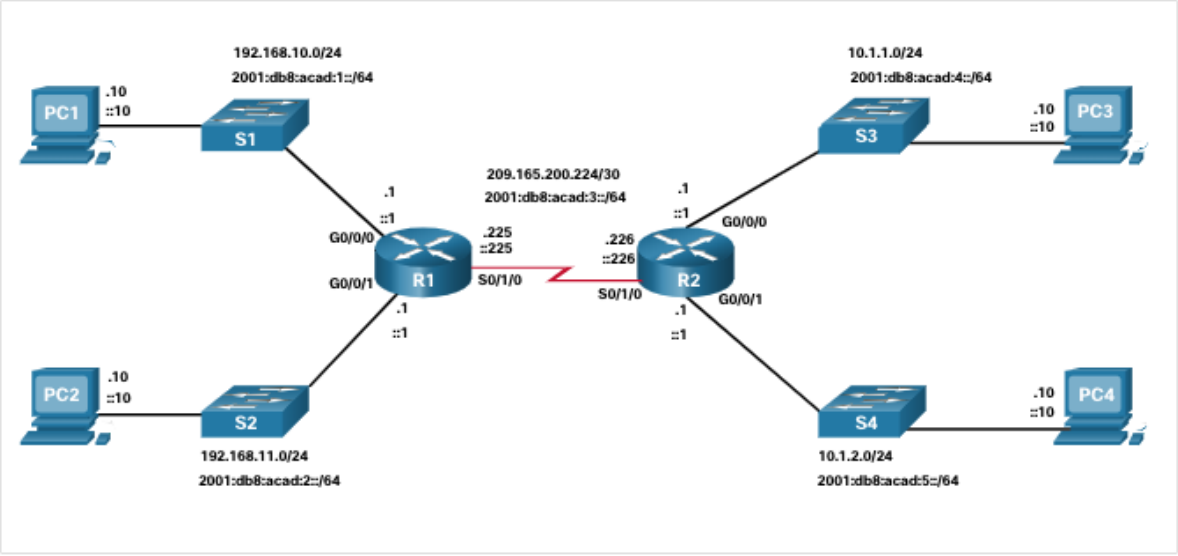
```
R1(config)# banner motd $ Authorized Access Only! $
R1(config)#
```

Save the changes on a router, as shown in the example.

```
R1# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

### *Dual Stack Topology*

One distinguishing feature between switches and routers is the type of interfaces supported by each. For example, Layer 2 switches support LANs; therefore, they have multiple FastEthernet or Gigabit Ethernet ports. The dual stack topology in the figure is used to demonstrate the configuration of router IPv4 and IPv6 interfaces.



### *Configure Router Interfaces*

Routers support LANs and WANs and can interconnect different types of networks; therefore, they support many types of interfaces. For example, G2 ISRs have one or two integrated Gigabit Ethernet interfaces and High-Speed WAN Interface Card (HWIC) slots to accommodate other types of network interfaces, including serial, DSL, and cable interfaces.

To be available, an interface must be:

- **Configured with at least one IP address** - Use the **ip address** *ip-address subnet-mask* and the **ipv6 address** *ipv6-address/prefix* interface configuration commands.
- **Activated** - By default, LAN and WAN interfaces are not activated (**shutdown**). To enable an interface, it must be activated using the **no shutdown** command. (This is similar to powering on the interface.) The interface must also be connected to another device (a hub, a switch, or another router) for the physical layer to be active.
- **Description** - Optionally, the interface could also be configured with a short description of up to 240 characters. It is good practice to configure a description on each interface. On production networks, the benefits of interface descriptions are quickly realized as they are helpful in troubleshooting and in identifying a third-party connection and contact information.
    - The example shows the configure for the interfaces on R1:

```
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# ip address 192.168.10.1 255.255.255.0
R1(config-if)# ipv6 address 2001:db8:acad:1::1/64
R1(config-if)# description Link to LAN 1
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# ip address 192.168.11.1 255.255.255.0
R1(config-if)# ipv6 address 2001:db8:acad:2::1/64
R1(config-if)# description Link to LAN 2
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface serial 0/0/0
R1(config-if)# ip address 209.165.200.225 255.255.255.252
R1(config-if)# ipv6 address 2001:db8:acad:3::225/64
R1(config-if)# description Link to R2
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)#
```

### *IPv4 Loopback Interfaces*

Another common configuration of Cisco IOS routers is enabling a loopback interface.

- The loopback interface is a logical interface that is internal to the router. It is not assigned to a physical port and can never be connected to any other device. It is considered a software interface that is automatically placed in an “up” state, as long as the router is functioning.
- The loopback interface is useful in testing and managing a Cisco IOS device because it ensures that at least one interface will always be available. For example, it can be used for testing purposes, such as testing internal routing processes, by emulating networks behind the router.
- Loopback interfaces are also commonly used in lab environments to create additional interfaces. For example, you can create multiple loopback interfaces on a router to simulate more networks for configuration practice and testing purposes. The IPv4 address for each loopback interface must be unique and unused by any other interface. In this curriculum, we often use a loopback interface to simulate a link to the internet.

- Enabling and assigning a loopback address is simple:

```
Router(config)# interface loopback number
Router(config-if)# ip address ip-address subnet-mask
```

### Verify Directly Connected Networks

#### *Interface Verification Commands*

There are several **show** commands that can be used to verify the operation and configuration of an interface.

The following commands are especially useful to quickly identify the status of an interface:

- **show ip interface brief** and **show ipv6 interface brief** - These display a summary for all interfaces including the IPv4 or IPv6 address of the interface and current operational status.
- **show running-config interface** *interface-id* - This displays the commands applied to the specified interface.
- **show ip route** and **show ipv6 route** - These display the contents of the IPv4 or IPv6 routing table stored in RAM. In Cisco IOS 15, active interfaces should appear in the routing table with two related entries identified by the code '**C**' (Connected) or '**L**' (Local). In previous IOS versions, only a single entry with the code '**C**' will appear.

#### *Verify Interface Status*

The output of the **show ip interface brief** and **show ipv6 interface brief** commands can be used to quickly reveal the status of all interfaces on the router. You can verify that the interfaces are active and operational as indicated by the Status of "up" and Protocol of "up", as shown in the example. A different output would indicate a problem with either the configuration

```
R1# show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0/0   192.168.10.1    YES manual up                    up
GigabitEthernet0/0/1   192.168.11.1    YES manual up                    up
Serial0/1/0            209.165.200.225 YES manual up                    up
Serial0/1/1            unassigned      YES unset  administratively down down
R1# show ipv6 interface brief
GigabitEthernet0/0/0   [up/up]
    FE80::7279:B3FF:FE92:3130
    2001:DB8:ACAD:1::1
GigabitEthernet0/0/1   [up/up]
    FE80::7279:B3FF:FE92:3131
    2001:DB8:ACAD:2::1
Serial0/1/0            [up/up]
    FE80::7279:B3FF:FE92:3130
    2001:DB8:ACAD:3::1
Serial0/1/1            [down/down]     Unassigned
```

#### *Verify IPv6 Link Local and Multicast Addresses*

The output of the **show ipv6 interface brief** command displays two configured IPv6 addresses per interface. One address is the IPv6 global unicast address that was manually entered. The other address, which begins with FE80, is the link-local unicast address for the interface. A link-local address is automatically added to an interface whenever a global unicast address is assigned. An IPv6 network interface is required to have a link-local address, but not necessarily a global unicast address.

The **show ipv6 interface gigabitethernet 0/0/0 c**ommand displays the interface status and all of the IPv6 addresses belonging to the interface. Along with the link local address and global unicast address, the output includes the multicast addresses assigned to the interface, beginning with prefix FF02, as shown in the example.

```
R1# show ipv6 interface gigabitethernet 0/0/0
GigabitEthernet0/0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::7279:B3FF:FE92:3130
  No Virtual link-local address(es):
  Global unicast address(es):
    2001:DB8:ACAD:1::1, subnet is 2001:DB8:ACAD:1::/64
  Joined group address(es):
    FF02::1
    FF02::1:FF00:1
    FF02::1:FF92:3130
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
```

### *Verify Interface Configuration*

The output of the **show running-config interface** command displays the current commands applied to the specified interface, as shown.

The following two commands are used to gather more detailed interface information:

- **show interfaces**- Displays interface information and packet flow count for all interfaces on the device.
- **show ip interface** and **show ipv6 interface** - Displays the IPv4 and IPv6 related information for all interfaces on a router.

```
R1 show running-config interface gigabitethernet 0/0/0
Building configuration...
Current configuration : 158 bytes
!
interface GigabitEthernet0/0/0
 description Link to LAN 1
 ip address 192.168.10.1 255.255.255.0
 negotiation auto
 ipv6 address 2001:DB8:ACAD:1::1/64
end
R1#
```

### *Verify Routes*

The output of the **show ip route** and **show ipv6 route** commands reveal the three directly connected network entries and the three local host route interface entries, as shown in the example.

The local host route has an administrative distance of 0. It also has a /32 mask for IPv4, and a /128 mask for IPv6. The local host route is for routes on the router that owns the IP address. It is used to allow the router to process packets destined to that IP.

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

Gateway of last resort is not set
      192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.10.0/24 is directly connected, GigabitEthernet0/0/0
L        192.168.10.1/32 is directly connected, GigabitEthernet0/0/0
      192.168.11.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.11.0/24 is directly connected, GigabitEthernet0/0/1
L        192.168.11.1/32 is directly connected, GigabitEthernet0/0/1
      209.165.200.0/24 is variably subnetted, 2 subnets, 2 masks
C        209.165.200.224/30 is directly connected, Serial0/1/0
L        209.165.200.225/32 is directly connected, Serial0/1/0A
R1# show ipv6 route
IPv6 Routing Table - default - 7 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route

C   2001:DB8:ACAD:1::/64 [0/0]
     via GigabitEthernet0/0/0, directly connected
L   2001:DB8:ACAD:1::1/128 [0/0]
     via GigabitEthernet0/0/0, receive
C   2001:DB8:ACAD:2::/64 [0/0]
     via GigabitEthernet0/0/1, directly connected
L   2001:DB8:ACAD:2::1/128 [0/0]
     via GigabitEthernet0/0/1, receive
C   2001:DB8:ACAD:3::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:DB8:ACAD:3::1/128 [0/0]
     via Serial0/1/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
R1#
```

A '**C**' next to a route within the routing table indicates that this is a directly connected network. When the router interface is configured with a global unicast address and is in the "up/up" state, the IPv6 prefix and prefix length are added to the IPv6 routing table as a connected route.
The IPv6 global unicast address applied to the interface is also installed in the routing table as a local route. The local route has a /128 prefix. Local routes are used by the routing table to efficiently process packets with the interface address of the router as the destination.

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

Gateway of last resort is not set
      192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.10.0/24 is directly connected, GigabitEthernet0/0/0
L        192.168.10.1/32 is directly connected, GigabitEthernet0/0/0
      192.168.11.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.11.0/24 is directly connected, GigabitEthernet0/0/1
L        192.168.11.1/32 is directly connected, GigabitEthernet0/0/1
      209.165.200.0/24 is variably subnetted, 2 subnets, 2 masks
C        209.165.200.224/30 is directly connected, Serial0/1/0
L        209.165.200.225/32 is directly connected, Serial0/1/0A
R1# show ipv6 route
IPv6 Routing Table - default - 7 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route

C   2001:DB8:ACAD:1::/64 [0/0]
     via GigabitEthernet0/0/0, directly connected
L   2001:DB8:ACAD:1::1/128 [0/0]
     via GigabitEthernet0/0/0, receive
C   2001:DB8:ACAD:2::/64 [0/0]
     via GigabitEthernet0/0/1, directly connected
L   2001:DB8:ACAD:2::1/128 [0/0]
     via GigabitEthernet0/0/1, receive
C   2001:DB8:ACAD:3::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:DB8:ACAD:3::1/128 [0/0]
     via Serial0/1/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
R1#
```

### *Filter Show Command Output*

Commands that generate multiple screens of output are, by default, paused after 24 lines. At the end of the paused output, the --More-- text displays. Pressing **Enter** displays the next line and pressing the spacebar displays the next set of lines. Use the **terminal length** command to specify the number of lines to be displayed. A value of 0 (zero) prevents the router from pausing between screens of output.

Another very useful feature that improves the user experience in the CLI is the filtering of **show** output. Filtering commands can be used to display specific sections of output. To enable the filtering command, enter a pipe (**|**) character after the **show** command and then enter a filtering parameter and a filtering expression.

There are four filtering parameters that can be configured after the pipe:

- section - Shows the entire section that starts with the filtering expression.
- include - Includes all output lines that match the filtering expression.
- exclude - Excludes all output lines that match the filtering expression.
- begin - Shows all the output lines from a certain point, starting with the line that matches the filtering expression

### *Command History Feature*

The command history feature is useful because it temporarily stores the list of executed commands to be recalled.

- To recall commands in the history buffer, press **Ctrl**+**P** or the **Up Arrow** key. The command output begins with the most recent command. Repeat the key sequence to recall successively older commands. To return to more recent commands in the history buffer, press **Ctrl**+**N** or the **Down Arrow** key. Repeat the key sequence to recall successively more recent commands.
- By default, command history is enabled and the system captures the last 10 command lines in its history buffer. Use the **show history** privileged EXEC command to display the contents of the buffer.
- It is also practical to increase the number of command lines that the history buffer records during the current terminal session only. Use the **terminal history size** user EXEC command to increase or decrease the size of the buffer.

**Laboratory Exercise: Basic Switch and Router Configuration**

## Module 2: Switching Concepts

### *Objectives:*

At the end of this module, the student should be able to:

- Explain how frames are forwarded in a switched network;
- Compare a collision domain to a broadcast domain.

## Frame Forwarding

### *Switching in Networking*

Two terms are associated with frames entering or leaving an interface:

- **Ingress** – entering the interface
- **Egress** – exiting the interface

A switch forwards based on the ingress interface and the destination MAC address.

A switch uses its MAC address table to make forwarding decisions.

**Note**: A switch will never allow traffic to be forwarded out the interface it received the traffic.


Port Table

| Destination Addresses | Port |
|---|---|
| EE | 1 |
| AA | 2 |
| BA | 3 |
| EA | 4 |
| AC | 5 |
| AB | 6 |

### *The Switch MAC Address Table*

A switch will use the destination MAC address to determine the egress interface.
Before a switch can make this decision it must learn what interface the destination is located.
A switch builds a MAC address table, also known as a Content Addressable Memory (CAM) table, by recording the source MAC address into the table along with the port it was received.

### *The Switch Learn and Forward Method*

The switch uses a two step process:

**Step 1.** Learn – Examines Source Address

- Adds the source MAC if not in table
- Resets the time out setting back to 5 minutes if source is in the table

**Step 2.** Forward – Examines Destination Address

- If the destination MAC is in the MAC address table it is forwarded out the specified port.
- If a destination MAC is not in the table, it is flooded out all interfaces except the one it was received.

### *Switch Forwarding Methods*

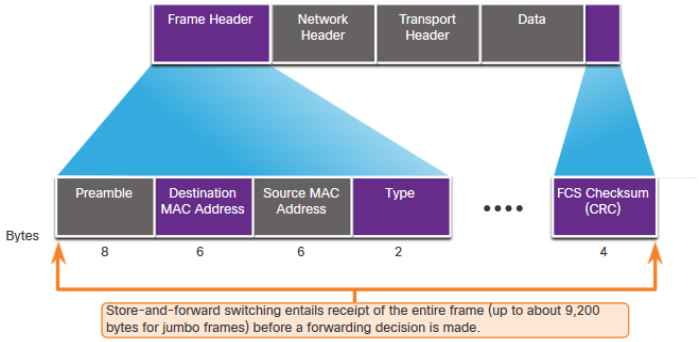
Switches use software on application-specific-integrated circuits (ASICs) to make very quick decisions.
A switch will use one of two methods to make forwarding decisions after it receives a frame:

- **Store-and-forward switching** - Receives the entire frame and ensures the frame is valid. Store-and-forward switching is Cisco's preferred switching method.
- **Cut-through switching** – Forwards the frame immediately after determining the destination MAC address of an incoming frame and the egress port.

### *Store-and-Forward Switching*

Store-and-forward has two primary characteristics:

- Error Checking – The switch will check the Frame Check Sequence (FCS) for CRC errors. Bad frames will be discarded.
- Buffering – The ingress interface will buffer the frame while it checks the FCS. This also allows the switch to adjust to a potential difference in speeds between the ingress and egress ports.

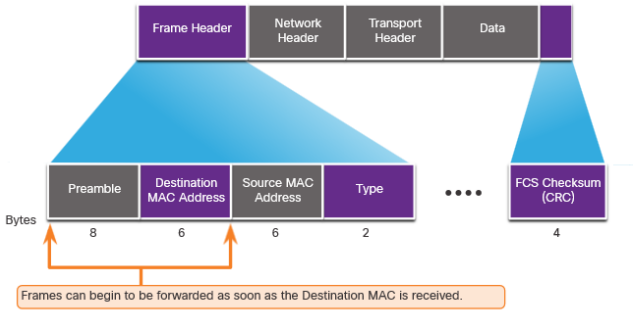



### *Cut-Through Switching*

- Cut-through forwards the frame immediately after determining the destination MAC.
- Fragment (Frag) Free method will check the destination and ensure that the frame is at least 64 Bytes. This will eliminate runts.

Concepts of Cut-Through switching:

- Is appropriate for switches needing latency to be under 10 microseconds
- Does not check the FCS, so it can propagate errors
- May lead to bandwidth issues if the switch propagates too many errors
- Cannot support ports with differing speeds going from ingress to egress

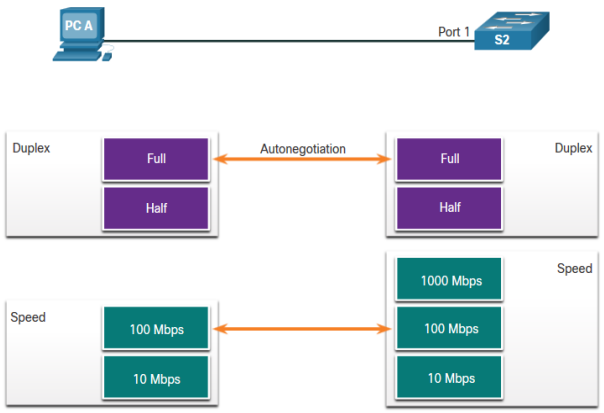


## Switching Domains

### *Collision Domains*

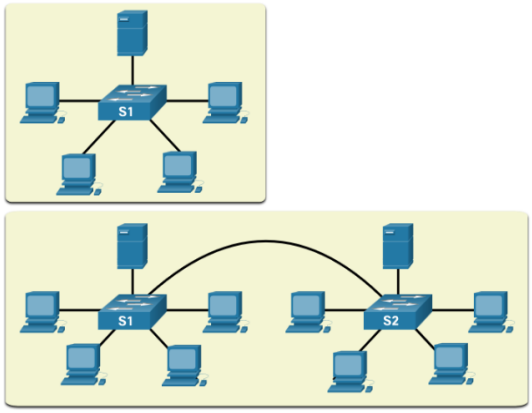
Switches eliminate collision domains and reduce congestion.

- When there is full duplex on the link the collision domains are eliminated.
- When there is one or more devices in half-duplex there will now be a collision domain.
  - There will now be contention for the bandwidth.
  - Collisions are now possible.
- Most devices, including Cisco and Microsoft use auto-negotiation as the default setting for duplex and speed.




### *Broadcast Domains*

- A broadcast domain extends across all Layer 1 or Layer 2 devices on a LAN.
  - Only a layer 3 device (router) will break the broadcast domain, also called a MAC broadcast domain.
  - The broadcast domain consists of all devices on the LAN that receive the broadcast traffic.
- When the layer 2 switch receives the broadcast it will flood it out all interfaces except for the ingress interface.
- Too many broadcasts may cause congestion and poor network performance.
- Increasing devices at Layer 1 or layer 2 will cause the broadcast domain to expand.




### *Alleviated Network Congestion*

Switches use the MAC address table and full-duplex to eliminate collisions and avoid congestion.
Features of the switch that alleviate congestion are as follows:

| Protocol | Function |
|---|---|
| **Fast Port Speeds** | Depending on the model, switches may have up to 100Gbps port speeds. |
| **Fast Internal Switching** | This uses fast internal bus or shared memory to improve performance. |
| **Large Frame Buffers** | This allows for temporary storage while processing large quantities of frames. |
| **High Port Density** | This provides many ports for devices to be connected to LAN with less cost. This also provides for more local traffic with less congestion. |

## Module 3: VLANs

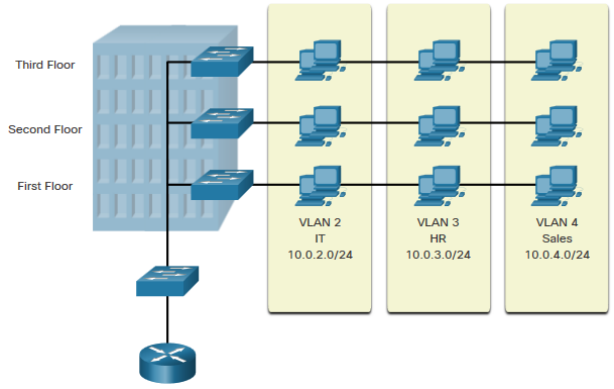
***Objectives:***

At the end of this module, the student should be able to:

- Explain the purpose of VLANs in a switched network;
- Explain how a switch forwards frames based on VLAN configuration in a multi-switched environment;
- Configure a switch port to be assigned to a VLAN based on requirements;
- Configure a trunk port on a LAN switch;
- Configure Dynamic Trunking Protocol (DTP).

### Overview of VLANs

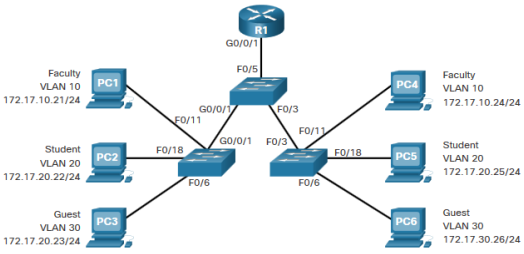
***VLAN Definitions***




VLANs are logical connections with other similar devices.
Placing devices into various VLANs have the following characteristics:

- Provides segmentation of the various groups of devices on the same switches
- Provide organization that is more manageable
  - Broadcasts, multicasts and unicasts are isolated in the individual VLAN
  - Each VLAN will have its own unique range of IP addressing
  - Smaller broadcast domains

***Benefits of a VLAN Design***




Benefits of using VLANs are as follows:

| Benefits | Description |
|---|---|
| Smaller Broadcast Domains | Dividing the LAN reduces the number of broadcast domains |
| Improved Security | Only users in the same VLAN can communicate together |
| Improved IT Efficiency | VLANs can group devices with similar requirements, e.g. faculty vs. students |
| Reduced Cost | One switch can support multiple groups or VLANs |
| Better Performance | Small broadcast domains reduce traffic, improving bandwidth |
| Simpler Management | Similar groups will need similar applications and other network resources |

### *Types of VLANs*

Default VLAN

VLAN 1 is the following:

- The default VLAN
- The default Native VLAN
- The default Management VLAN
- Cannot be deleted or renamed

**Note**: While we cannot delete VLAN1 Cisco will recommend that we assign these default features to other 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
```

### Data VLAN

- Dedicated to user-generated traffic (email and web traffic).
- VLAN 1 is the default data VLAN because all interfaces are assigned to this VLAN.

### Native VLAN

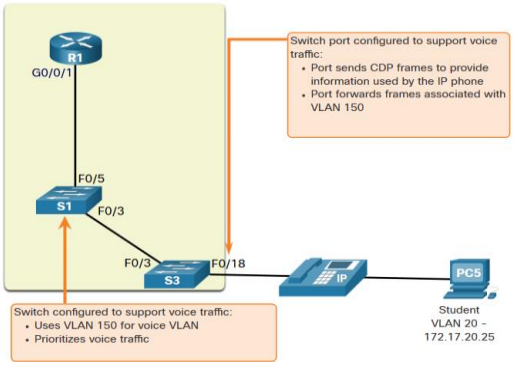
- This is used for trunk links only.
- All frames are tagged on an 802.1Q trunk link except for those on the native VLAN.

### Management VLAN

- This is used for SSH/Telnet VTY traffic and should not be carried with end user traffic.
- Typically, the VLAN that is the SVI for the Layer 2 switch.

### Voice VLAN

- A separate VLAN is required because Voice traffic requires:
  - Assured bandwidth
  - High QoS priority
  - Ability to avoid congestion
  - Delay less that 150 ms from source to destination
- The entire network must be designed to support voice.



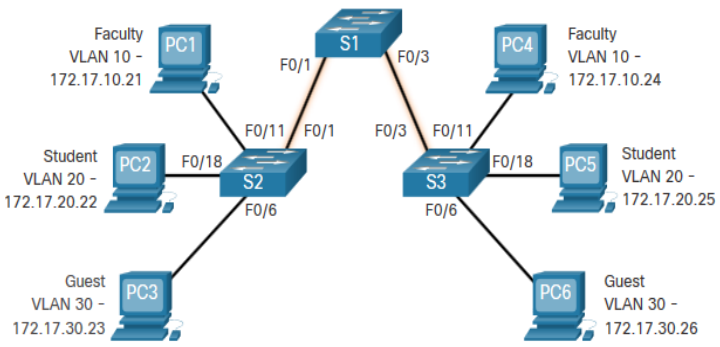

### VLANs in a Multi-Switched Environment

### *Defining VLAN Trunks*

A trunk is a point-to-point link between two network devices.

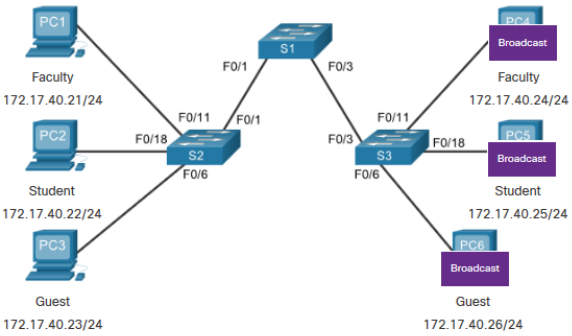
Cisco trunk functions:

- Allow more than one VLAN
- Extend the VLAN across the entire network
- By default, supports all VLANs
- Supports 802.1Q trunking



### *Networking Without VLANs*

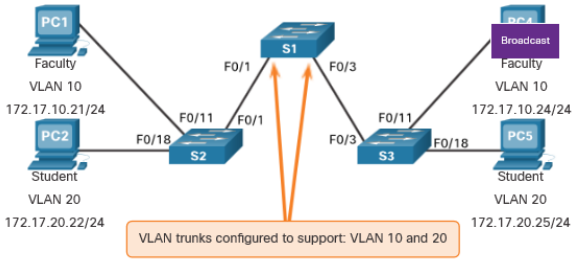
Without VLANs, all devices connected to the switches will receive all unicast, multicast, and broadcast traffic.




PC1 sends out a local Layer 2 broadcast. The switches forward the broadcast frame out all available ports.

### *Networks With VLANs*

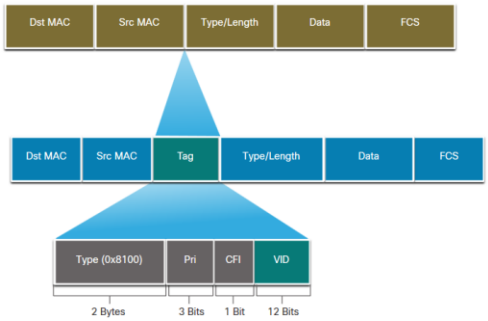
With VLANs, unicast, multicast, and broadcast traffic is confined to a VLAN. Without a Layer 3 device to connect the VLANs, devices in different VLANs cannot communicate.




PC1 sends out a local Layer 2 broadcast. The switches forward the broadcast frame only out ports configured for VLAN10.

### *VLAN Identification With a Tag*

- The IEEE 802.1Q header is 4 Bytes
- When the tag is created the FCS must be recalculated.
- When sent to end devices, this tag must be removed and the FCS recalculated back to its original number.

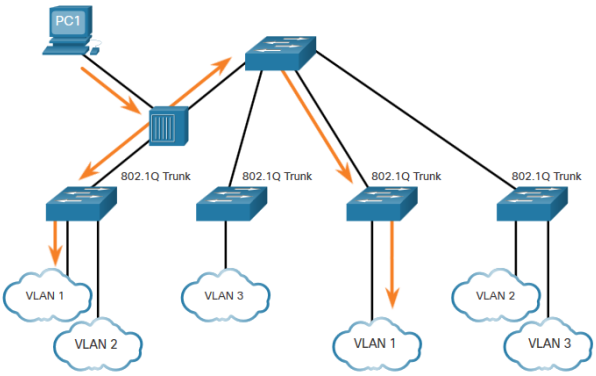



| 802.1Q VLAN Tag Field | Function |
|---|---|
| **Type** | • 2-Byte field with hexadecimal 0x8100<br>• This is referred to as Tag Protocol ID (TPID) |
| **User Priority** | • 3-bit value that supports |
| **Canonical Format Identifier (CFI)** | • 1-bit value that can support token ring frames on Ethernet |
| **VLAN ID (VID)** | • 12-bit VLAN identifier that can support up to 4096 VLANs |

### *Native VLANs and 802.1Q Tagging*

802.1Q trunk basics:

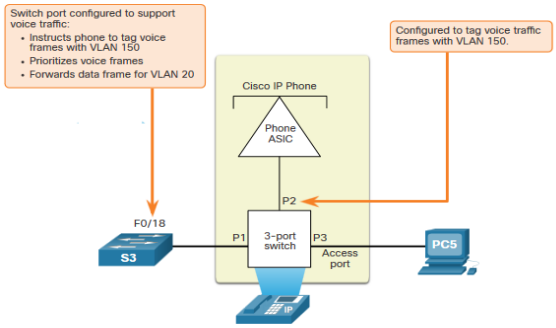
- Tagging is typically done on all VLANs.
- The use of a native VLAN was designed for legacy use, like the hub in the example.
- Unless changed, VLAN1 is the native VLAN.
- Both ends of a trunk link must be configured with the same native VLAN.
- Each trunk is configured separately, so it is possible to have a different native VLANs on separate trunks.




### *Voice VLAN Tagging*

The VoIP phone is a three port switch:

- The switch will use CDP to inform the phone of the Voice VLAN.
- The phone will tag its own traffic (Voice) and can set Cost of Service (CoS). CoS is QoS for layer 2.
- The phone may or may not tag frames from the PC.




| Traffic | Tagging Function |
|---|---|
| Voice VLAN | tagged with an appropriate Layer 2 class of service (CoS) priority value |
| Access VLAN | can also be tagged with a Layer 2 CoS priority value |
| Access VLAN | is not tagged (no Layer 2 CoS priority value) |

### *Voice VLAN Verification Example*

The **show interfaces fa0/18 switchport** command can show us both data and voice VLANs assigned to the interface.

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

## VLAN Configuration

### *VLAN Ranges on Catalyst Switches*

Catalyst switches 2960 and 3650 support over 4000 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
```

| Normal Range VLAN 1 – 1005 | Extended Range VLAN 1006 - 4095 |
|---|---|
| Used in Small to Medium sized businesses | Used by Service Providers |
| 1002 – 1005 are reserved for legacy VLANs | Are in Running-Config |
| 1, 1002 – 1005 are auto created and cannot be deleted | Supports fewer VLAN features |
| Stored in the vlan.dat file in flash | Requires VTP configurations |
| VTP can synchronize between switches | |

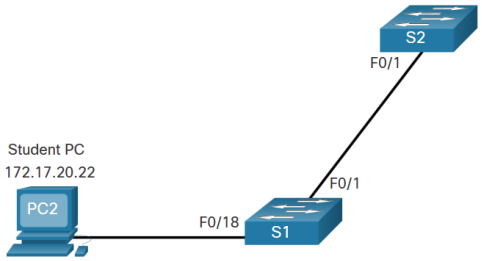
### *VLAN Creation Commands*

VLAN details are stored in the vlan.dat file. You create VLANs in the global configuration mode.

| Task | IOS Command |
|---|---|
| Enter global configuration mode. | Switch# **configure terminal** |
| Create a VLAN with a valid ID number. | Switch(config)# **vlan** *vlan-id* |
| Specify a unique name to identify the VLAN. | Switch(config-vlan)# **name** *vlan-name* |
| Return to the privileged EXEC mode. | Switch(config-vlan)# **end** |
| Enter global configuration mode. | Switch# **configure terminal** |

### *VLAN Creation Example*

- If the Student PC is going to be in VLAN 20, we will create the VLAN first and then name it.
- If you do not name it, the Cisco IOS will give it a default name of vlan and the four digit number of the VLAN. E.g. vlan0020 for VLAN 20.




| Prompt | Command |
|---|---|
| S1# | Configure terminal |
| S1(config)# | vlan 20 |
| S1(config-vlan)# | name student |
| S1(config-vlan)# | end |

### *VLAN Port Assignment Commands*

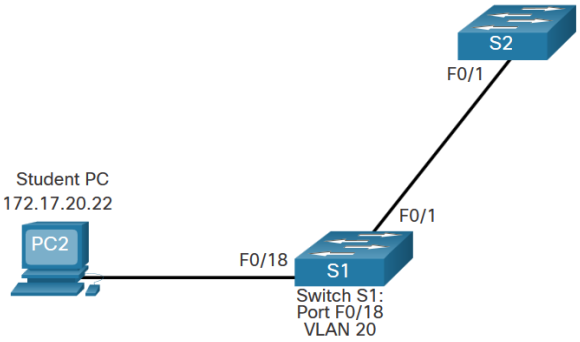
Once the VLAN is created, we can then assign it to the correct interfaces.

| Task | Command |
|---|---|
| Enter global configuration mode. | Switch# **configure terminal** |
| Enter interface configuration mode. | Switch(config)# **interface** *interface-id* |
| Set the port to access mode. | Switch(config-if)# **switchport mode access** |
| Assign the port to a VLAN. | Switch(config-if)# **switchport access vlan** *vlan-id* |
| Return to the privileged EXEC mode. | Switch(config-if)# **end** |

### *VLAN Port Assignment Example*

We can assign the VLAN to the port interface.

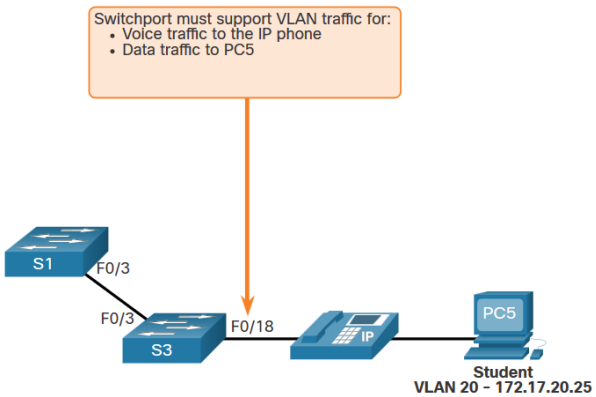
- Once the device is assigned the VLAN, then the end device will need the IP address information for that VLAN
- Here, Student PC receives 172.17.20.22




| Prompt | Command |
|---|---|
| S1# | Configure terminal |
| S1(config)# | Interface fa0/18 |
| S1(config-if)# | Switchport mode access |
| S1(config-if)# | Switchport access vlan 20 |
| S1(config-if)# | end |

### *Data and Voice VLANs*

An access port may only be assigned to one data VLAN. However it may also be assigned to one Voice VLAN for when a phone and an end device are off of the same switchport.



### *Data and Voice VLAN Example*

- We will want to create and name both Voice and Data VLANs.
- In addition to assigning the data VLAN, we will also assign the Voice VLAN and turn on QoS for the voice traffic to the interface.
- The newer catalyst switch will automatically create the VLAN, if it does not already exist, when it is assigned to an interface.

**Note**: QoS is beyond the scope of this course. Here we do show the use of the **mls qos trust [cos | device cisco-phone | dscp | ip-precedence]** command.

```
S1(config)# vlan 20
S1(config-vlan)# name student
S1(config-vlan)# vlan 150
S1(config-vlan)# name VOICE
S1(config-vlan)# exit
S1(config)# interface fa0/18
S1(config-if)# switchport mode access
S1(config-if)# switchport access vlan 20
S1(config-if)# mls qos trust cos
S1(config-if)# switchport voice vlan 150
S1(config-if)# end
```

```
% Access VLAN does not exist. Creating vlan 30
```

### *Verify VLAN Information*

Use the **show vlan** command. The complete syntax is:

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

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

```
S1# show interface vlan 20
Vlan20 is up, line protocol is up
  Hardware is EtherSVI, address is 001f.6ddb.3ec1 (bia 001f.6ddb.3ec1)
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set

(Output omitted)
```

| Task | Command Option |
|---|---|
| Display VLAN name, status, and its ports one VLAN per line. | **brief** |
| Display information about the identified VLAN ID number. | **id** *vlan-id* |
| Display information about the identified VLAN name. The *vlan-name* is an ASCII string from 1 to 32 characters. | **name** *vlan-name* |
| Display VLAN summary information. | **summary** |

### *Change VLAN Port Membership*

There are a number of ways to change VLAN membership:

- re-enter **switchport access vlan** *vlan-id* command
- use the **no switchport access vlan** to place interface back in VLAN 1

Use the **show vlan brief** or the **show interface fa0/18 switchport** commands to verify the correct VLAN association.

```
S1(config)# interface fa0/18
S1(config-if)# no switchport access vlan
S1(config-if)# end
S1#
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# show interfaces fa0/18 switchport
Name: Fa0/18
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
```

### *Delete VLANs*

Delete VLANs with the **no vlan** *vlan-id* command.

**Caution**: Before deleting a VLAN, reassign all member ports to a different VLAN.

- Delete all VLANs with the **delete flash:vlan.dat** or **delete vlan.dat** commands.
- Reload the switch when deleting all VLANs.

**Note**: To restore to factory default – unplug all data cables, erase the startup-configuration and delete the vlan.dat file, then reload the device.

## VLAN Trunks

### *Trunk Configuration Commands*

Configure and verify VLAN trunks. Trunks are layer 2 and carry traffic for all VLANs.

| Task | IOS Command |
|---|---|
| Enter global configuration mode. | Switch# **configure terminal** |
| Enter interface configuration mode. | Switch(config)# **interface** *interface-id* |
| Set the port to permanent trunking mode. | Switch(config-if)# **switchport mode trunk** |
| Sets the native VLAN to something other than VLAN 1. | Switch(config-if)# **switchport trunk native vlan** *vlan-id* |
| Specify the list of VLANs to be allowed on the trunk link. | Switch(config-if)# **switchport trunk allowed vlan** *vlan-list* |
| Return to the privileged EXEC mode. | Switch(config-if)# **end** |

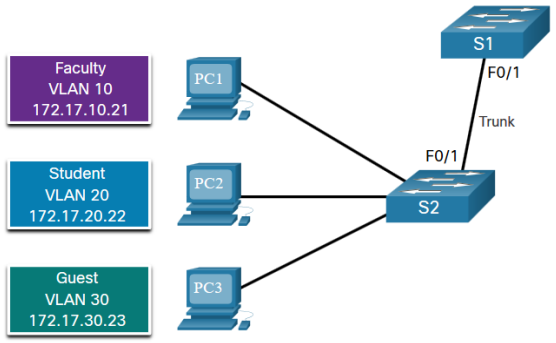
### *Trunk Configuration Example*

The subnets associated with each VLAN are:

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

F0/1 port on S1 is configured as a trunk port.

**Note**: This assumes a 2960 switch using 802.1q tagging. Layer 3 switches require the encapsulation to be configured before the trunk mode.




| Prompt | Command |
|---|---|
| S1(config)# | Interface fa0/1 |
| 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 |

### *Verify Trunk Configuration*

Set the trunk mode and native vlan. Notice **sh int fa0/1 switchport** command:

- Is set to trunk administratively
- Is set as trunk operationally (functioning)
- Encapsulation is dot1q
- Native VLAN set to VLAN 99
- All VLANs created on the switch will pass traffic on this trunk

```
S1(config)# interface fa0/1
S1(config-if)# switchport mode trunk
S1(config-if)# no switchport trunk native vlan 99
S1(config-if)# end
S1# show interfaces fa0/1 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)
```

### *Reset the Trunk to the Default State*

- Reset the default trunk settings with the no command.
  - All VLANs allowed to pass traffic
  - Native VLAN = VLAN 1
- Verify the default settings with a **sh int fa0/1 switchport** command.

```
S1(config)# interface fa0/1
S1(config-if)# no switchport trunk allowed vlan
S1(config-if)# no switchport trunk native vlan
S1(config-if)# end
```

```
S1# show interfaces fa0/1 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
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)
```

Reset the trunk to an access mode with the **switchport mode access** command:

- Is set to an access interface administratively
- Is set as an access interface operationally (functioning)

```
S1(config)# interface fa0/1
S1(config-if)# switchport mode access
S1(config-if)# end
S1# show interfaces fa0/1 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)
```

## Dynamic Trunking Protocol

### *Introduction to DTP*

Dynamic Trunking Protocol (DTP) is a proprietary Cisco protocol.
DTP characteristics are as follows:

- On by default on Catalyst 2960 and 2950 switches
- Dynamic-auto is default on the 2960 and 2950 switches
- May be turned off with the nonegotiate command
- May be turned back on by setting the interface to dynamic-auto

Setting a switch to a static trunk or static access will avoid negotiation issues with the **switchport mode trunk** or the **switchport mode access** commands.

### *Negotiated Interface Modes*

The **switchport mode** command has additional options.
Use the **switchport nonegotiate** interface configuration command to stop DTP negotiation.

| Option | Description |
|---|---|
| **access** | Permanent access mode and negotiates to convert the neighboring link into an access link |
| **dynamic auto** | Will becomes a trunk interface if the neighboring interface is set to trunk or desirable mode |
| **dynamic desirable** | Actively seeks to become a trunk by negotiating with other auto or desirable interfaces |
| **trunk** | Permanent trunking mode and negotiates to convert the neighboring link into a trunk link |

### *Results of a DTP Configuration*

DTP configuration options are as follows:

| | 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 |

### *Verify DTP Mode*

The default DTP configuration is dependent on the Cisco IOS version and platform.

- Use the **show dtp interface** command to determine the current DTP mode.
- Best practice recommends that the interfaces be set to access or trunk and to turnoff DTP

```
S1# show dtp interface fa0/1
DTP information for FastEthernet0/1:
TOS/TAS/TNS: ACCESS/AUTO/ACCESS
TOT/TAT/TNT: NATIVE/NEGOTIATE/NATIVE
Neighbor address 1: C80084AEF101
Neighbor address 2: 000000000000
Hello timer expiration (sec/state): 11/RUNNING
Access timer expiration (sec/state): never/STOPPED
Negotiation timer expiration (sec/state): never/STOPPED
Multidrop timer expiration (sec/state): never/STOPPED
FSM state: S2:ACCESS
# times multi & trunk 0
Enabled: yes
In STP: no
```

## Laboratory Exercise: VLAN Configuration

## Module 4: Inter-VLAN Routing

***Objectives:***

At the end of this module, the student should be able to:

- Describe options for configuring inter-VLAN routing;
- Configure router-on-a-stick inter-VLAN routing;
- Configure inter-VLAN routing using Layer 3 switching;
- Troubleshoot common inter-VLAN configuration issues.

### Inter-VLAN Routing Operation

***What is Inter-VLAN Routing?***

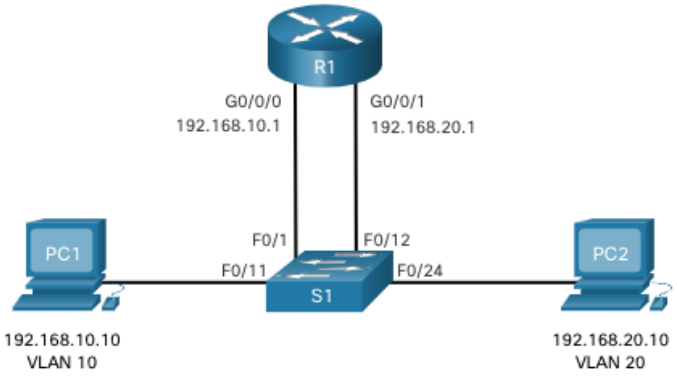
VLANs are used to segment switched Layer 2 networks for a variety of reasons. Regardless of the reason, hosts in one VLAN cannot communicate with hosts in another VLAN unless there is a router or a Layer 3 switch to provide routing services.
Inter-VLAN routing is the process of forwarding network traffic from one VLAN to another VLAN.

There are three inter-VLAN routing options:

- **Legacy Inter-VLAN routing** - This is a legacy solution. It does not scale well.
- **Router-on-a-Stick** - This is an acceptable solution for a small to medium-sized network.
- **Layer 3 switch using switched virtual interfaces (SVIs)** - This is the most scalable solution for medium to large organizations.

***Legacy Inter-VLAN Routing***

- The first inter-VLAN routing solution relied on using a router with multiple Ethernet interfaces. Each router interface was connected to a switch port in different VLANs. The router interfaces served as the default gateways to the local hosts on the VLAN subnet.
- Legacy inter-VLAN routing using physical interfaces works, but it has a significant limitation. It is not reasonably scalable because routers have a limited number of physical interfaces. Requiring one physical router interface per VLAN quickly exhausts the physical interface capacity of a router.
- **Note**: This method of inter-VLAN routing is no longer implemented in switched networks and is included for explanation purposes only.



### *Router-on-a-Stick Inter-VLAN Routing*

The 'router-on-a-stick' inter-VLAN routing method overcomes the limitation of the legacy inter-VLAN routing method. It only requires one physical Ethernet interface to route traffic between multiple VLANs on a network.

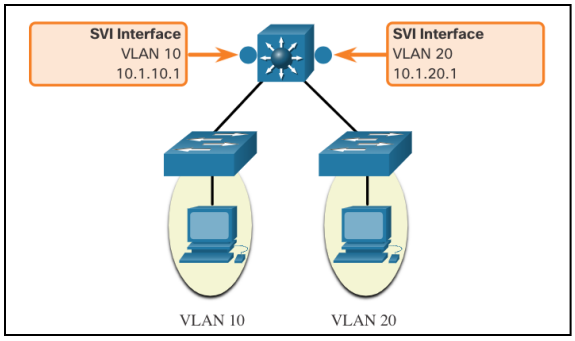
- A Cisco IOS router Ethernet interface is configured as an 802.1Q trunk and connected to a trunk port on a Layer 2 switch. Specifically, the router interface is configured using subinterfaces to identify routable VLANs.
- The configured subinterfaces are software-based virtual interfaces. Each is associated with a single physical Ethernet interface. Subinterfaces are configured in software on a router. Each subinterface is independently configured with an IP address and VLAN assignment. Subinterfaces are configured for different subnets that correspond to their VLAN assignment. This facilitates logical routing.
- When VLAN-tagged traffic enters the router interface, it is forwarded to the VLAN subinterface. After a routing decision is made based on the destination IP network address, the router determines the exit interface for the traffic. If the exit interface is configured as an 802.1q subinterface, the data frames are VLAN-tagged with the new VLAN and sent back out the physical interface

**Note**: The router-on-a-stick method of inter-VLAN routing does not scale beyond 50 VLANs.

### *Inter-VLAN Routing on a Layer 3 Switch*

The modern method of performing inter-VLAN routing is to use Layer 3 switches and switched virtual interfaces (SVI). An SVI is a virtual interface that is configured on a Layer 3 switch, as shown in the figure.

**Note**: A Layer 3 switch is also called a multilayer switch as it operates at Layer 2 and Layer 3. However, in this course we use the term Layer 3 switch.




Inter-VLAN SVIs are created the same way that the management VLAN interface is configured. The SVI is created for a VLAN that exists on the switch. Although virtual, the SVI performs the same functions for the VLAN as a router interface would. Specifically, it provides Layer 3 processing for packets that are sent to or from all switch ports associated with that VLAN.

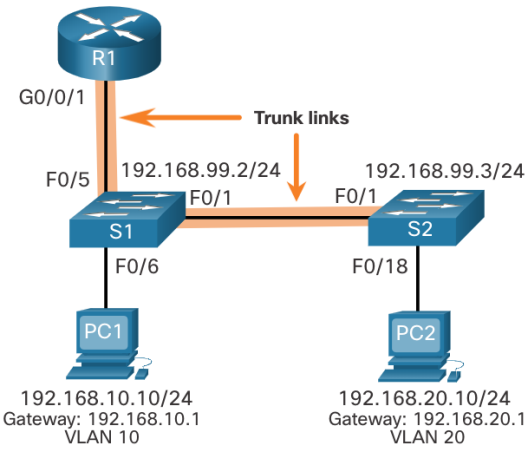
The following are advantages of using Layer 3 switches for inter-VLAN routing:

- They are much faster than router-on-a-stick because everything is hardware switched and routed.
- There is no need for external links from the switch to the router for routing.
- They are not limited to one link because Layer 2 EtherChannels can be used as trunk links between the switches to increase bandwidth.
- Latency is much lower because data does not need to leave the switch in order to be routed to a different network.
- They more commonly deployed in a campus LAN than routers.
- The only disadvantage is that Layer 3 switches are more expensive.

## Router-on-a-Stick Inter-VLAN Routing

### *Router-on-a-Stick Scenario*

- In the figure, the R1 GigabitEthernet 0/0/1 interface is connected to the S1 FastEthernet 0/5 port. The S1 FastEthernet 0/1 port is connected to the S2 FastEthernet 0/1 port. These are trunk links that are required to forward traffic within and between VLANs.
- To route between VLANs, the R1 GigabitEthernet 0/0/1 interface is logically divided into three subinterfaces, as shown in the table. The table also shows the three VLANs that will be configured on the switches.
- Assume that R1, S1, and S2 have initial basic configurations. Currently, PC1 and PC2 cannot **ping** each other because they are on separate networks. Only S1 and S2 can **ping** each other, but they but are unreachable by PC1 or PC2 because they are also on different networks.
- To enable devices to ping each other, the switches must be configured with VLANs and trunking, and the router must be configured for inter-VLAN routing.




| Subinterface | VLAN | IP Address |
|---|---|---|
| G0/0/1.10 | 10 | 192.168.10.1/24 |
| G0/0/1.20 | 20 | 192.168.20.1/24 |
| G0/0/1.30 | 99 | 192.168.99.1/24 |

### *S1 VLAN and Trunking Configuration*

Complete the following steps to configure S1 with VLANs and trunking:

- **Step 1**. Create and name the VLANs.
- **Step 2**. Create the management interface.
- **Step 3**. Configure access ports.
- **Step 4**. Configure trunking ports.

### *S2 VLAN and Trunking Configuration*

The configuration for S2 is similar to S1.

```
S2(config)# vlan 10
S2(config-vlan)# name LAN10
S2(config-vlan)# exit
S2(config)# vlan 20
S2(config-vlan)# name LAN20
S2(config-vlan)# exit
S2(config)# vlan 99
S2(config-vlan)# name Management
S2(config-vlan)# exit
S2(config)#
S2(config)# interface vlan 99
S2(config-if)# ip add 192.168.99.3 255.255.255.0
S2(config-if)# no shut
S2(config-if)# exit
S2(config)# ip default-gateway 192.168.99.1
S2(config)# interface fa0/18
S2(config-if)# switchport mode access
S2(config-if)# switchport access vlan 20
S2(config-if)# no shut
S2(config-if)# exit
S2(config)# interface fa0/1
S2(config-if)# switchport mode trunk
S2(config-if)# no shut
S2(config-if)# exit
S2(config-if)# end
*Mar  1 00:23:52.137: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1,
changed state to up
```

### *R1 Subinterface Configuration*

The router-on-a-stick method requires you to create a subinterface for each VLAN to be routed. A subinterface is created using the **interface** *interface_id subinterface_id* global configuration mode command. The subinterface syntax is the physical interface followed by a period and a subinterface number. Although not required, it is customary to match the subinterface number with the VLAN number.

Each subinterface is then configured with the following two commands:

- **encapsulation dot1q** *vlan_id* **[native]** - This command configures the subinterface to respond to 802.1Q encapsulated traffic from the specified *vlan-id*. The **native** keyword option is only appended to set the native VLAN to something other than VLAN 1.
- **ip address** *ip-address subnet-mask* - This command configures the IPv4 address of the subinterface. This address typically serves as the default gateway for the identified VLAN.

Repeat the process for each VLAN to be routed. Each router subinterface must be assigned an IP address on a unique subnet for routing to occur. When all subinterfaces have been created, enable the physical interface using the **no shutdown** interface configuration command. If the physical interface is disabled, all subinterfaces are disabled.

In the configuration, the R1 G0/0/1 subinterfaces are configured for VLANs 10, 20, and 99.

```
R1(config)# interface G0/0/1.10
R1(config-subif)# Description Default Gateway for VLAN 10
R1(config-subif)# encapsulation dot1Q 10
R1(config-subif)# ip add 192.168.10.1 255.255.255.0
R1(config-subif)# exit
R1(config)#
R1(config)# interface G0/0/1.20
R1(config-subif)# Description Default Gateway for VLAN 20
R1(config-subif)# encapsulation dot1Q 20
R1(config-subif)# ip add 192.168.20.1 255.255.255.0
R1(config-subif)# exit
R1(config)#
R1(config)# interface G0/0/1.99
R1(config-subif)# Description Default Gateway for VLAN 99
R1(config-subif)# encapsulation dot1Q 99
R1(config-subif)# ip add 192.168.99.1 255.255.255.0
R1(config-subif)# exit
R1(config)#
R1(config)# interface G0/0/1
R1(config-if)# Description Trunk link to S1
R1(config-if)# no shut
R1(config-if)# end
R1#
*Sep 15 19:08:47.015: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/1, changed state to down
*Sep 15 19:08:50.071: %LINK-3-UPDOWN: Interface GigabitEthernet0/0/1, changed state to up
*Sep 15 19:08:51.071: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1,
changed state to up
R1#
```

### *Verify Connectivity Between PC1 and PC2*

The router-on-a-stick configuration is complete after the switch trunk and the router subinterfaces have been configured. The configuration can be verified from the hosts, router, and switch.

From a host, verify connectivity to a host in another VLAN using the **ping** command. It is a good idea to first verify the current host IP configuration using the **ipconfig** Windows host command.

Next, use **ping** to verify connectivity with PC2 and S1, as shown in the figure. The **ping** output successfully confirms inter-VLAN routing is operating.

```
C:\Users\PC1> ping 192.168.20.10
Pinging 192.168.20.10 with 32 bytes of data:
Reply from 192.168.20.10: bytes=32 time<1ms TTL=127
Reply from 192.168.20.10: bytes=32 time<1ms TTL=127
Reply from 192.168.20.10: bytes=32 time<1ms TTL=127
Reply from 192.168.20.10: bytes=32 time<1ms TTL=127
Ping statistics for 192.168.20.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss).
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\Users\PC1>
C:\Users\PC1> ping 192.168.99.2
Pinging 192.168.99.2 with 32 bytes of data:
Request timed out.
Request timed out.
Reply from 192.168.99.2: bytes=32 time=2ms TTL=254
Reply from 192.168.99.2: bytes=32 time=1ms TTL=254
Ping statistics for 192.168.99.2:
    Packets: Sent = 4, Received = 2, Lost = 2 (50% loss).
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 2ms, Average = 1ms
C:\Users\PC1>
```

### *Router-on-a-Stick Inter-VLAN Routing Verification*

In addition to using **ping** between devices, the following **show** commands can be used to verify and troubleshoot the router-on-a-stick configuration.

- **show ip route**
- **show ip interface brief**
- **show interfaces**
- **show interfaces trunk**

## Inter-VLAN Routing Using Layer 3 Switches

### *Layer 3 Switch Inter-VLAN Routing*

Inter-VLAN routing using the router-on-a-stick method is simple to implement for a small to medium-sized organization. However, a large enterprise requires a faster, much more scalable method to provide inter-VLAN routing.

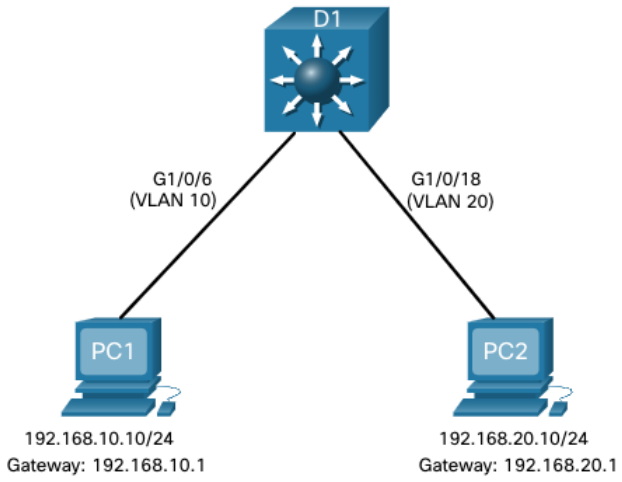
Enterprise campus LANs use Layer 3 switches to provide inter-VLAN routing. Layer 3 switches use hardware-based switching to achieve higher-packet processing rates than routers. Layer 3 switches are also commonly implemented in enterprise distribution layer wiring closets.

Capabilities of a Layer 3 switch include the ability to do the following:

- Route from one VLAN to another using multiple switched virtual interfaces (SVIs).
- Convert a Layer 2 switchport to a Layer 3 interface (i.e., a routed port). A routed port is similar to a physical interface on a Cisco IOS router.
- To provide inter-VLAN routing, Layer 3 switches use SVIs. SVIs are configured using the same **interface vlan** *vlan-id* command used to create the management SVI on a Layer 2 switch. A Layer 3 SVI must be created for each of the routable VLANs.

### *Layer 3 Switch Scenario*

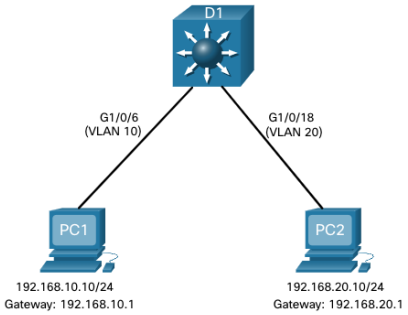
In the figure, the Layer 3 switch, D1, is connected to two hosts on different VLANs. PC1 is in VLAN 10 and PC2 is in VLAN 20, as shown. The Layer 3 switch will provide inter-VLAN routing services to the two hosts.



### *Layer 3 Switch Configuration*

Complete the following steps to configure S1 with VLANs and trunking:

- **Step 1**. Create the VLANs. In the example, VLANs 10 and 20 are used.
- **Step 2**. Create the SVI VLAN interfaces. The IP address configured will serve as the default gateway for hosts in the respective VLAN.
- **Step 3**. Configure access ports. Assign the appropriate port to the required VLAN.
- **Step 4**. Enable IP routing. Issue the **ip routing** global configuration command to allow traffic to be exchanged between VLANs 10 and 20. This command must be configured to enable inter-VAN routing on a Layer 3 switch for IPv4.




### *Layer 3 Switch Inter-VLAN Routing Verification*

Inter-VLAN routing using a Layer 3 switch is simpler to configure than the router-on-a-stick method. After the configuration is complete, the configuration can be verified by testing connectivity between the hosts.

- From a host, verify connectivity to a host in another VLAN using the **ping** command. It is a good idea to first verify the current host IP configuration using the **ipconfig** Windows host command.
- Next, verify connectivity with PC2 using the **ping** Windows host command. The successful **ping** output confirms inter-VLAN routing is operating.

### *Routing on a Layer 3 Switch*

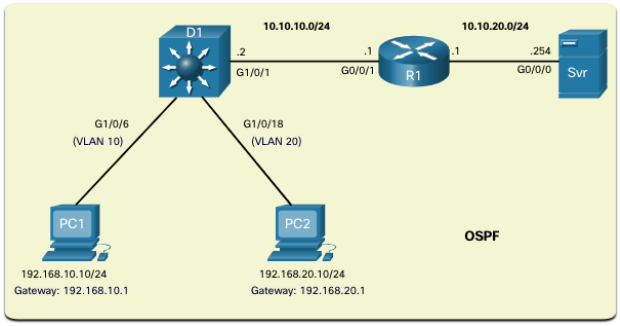
If VLANs are to be reachable by other Layer 3 devices, then they must be advertised using static or dynamic routing. To enable routing on a Layer 3 switch, a routed port must be configured.

A routed port is created on a Layer 3 switch by disabling the switchport feature on a Layer 2 port that is connected to another Layer 3 device. Specifically, configuring the **no switchport** interface configuration command on a Layer 2 port converts it into a Layer 3 interface. Then the interface can be configured with an IPv4 configuration to connect to a router or another Layer 3 switch.

### *Routing Scenario on a Layer 3 Switch*

In the figure, the previously configured D1 Layer 3 switch is now connected to R1. R1 and D1 are both in an Open Shortest Path First (OSPF) routing protocol domain. Assume inter-VLAN has been successfully implemented on D1. The G0/0/1 interface of R1 has also been configured and enabled. Additionally, R1 is using OSPF to advertise its two networks, 10.10.10.0/24 and 10.20.20.0/24.

**Note**: OSPF routing configuration is covered in another course. In this module, OSPF configuration commands will be given to you in all activities and assessments. It is not required that you understand the configuration in order to enable OSPF routing on the Layer 3 switch.




### *Routing Configuration on a Layer 3 Switch*

Complete the following steps to configure D1 to route with R1:

- **Step 1**. Configure the routed port. Use the **no switchport** command to convert the port to a routed port, then assign an IP address and subnet mask. Enable the port.
- **Step 2**. Enable routing. Use the **ip routing** global configuration command to enable routing.
- **Step 3**. Configure routing. Use an appropriate routing method. In this example, Single-Area OSPFv2 is configured
- **Step 4**. Verify routing. Use the **show ip route** command.
- **Step 5**. Verify connectivity. Use the **ping** command to verify reachability.

## Troubleshoot Inter-VLAN Routing

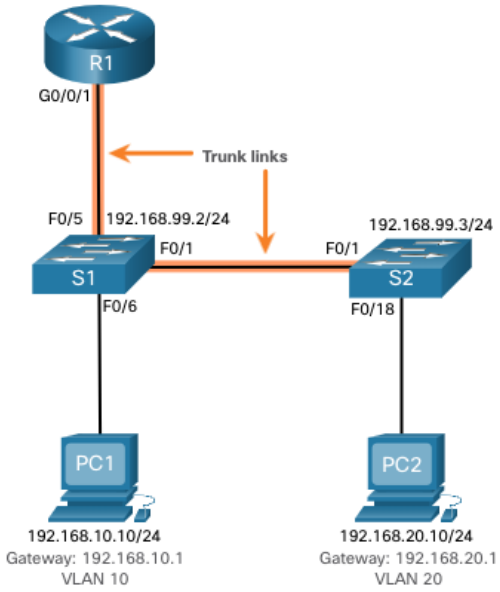
### *Common Inter-VLAN Issues*

There are a number of reasons why an inter-VAN configuration may not work. All are related to connectivity issues. First, check the physical layer to resolve any issues where a cable might be connected to the wrong port. If the connections are correct, then use the list in the table for other common reasons why inter-VLAN connectivity may fail.

| Issue Type | How to Fix | How to Verify |
|---|---|---|
| Missing VLANs | • Create (or re-create) the VLAN if it does not exist.<br>• Ensure host port is assigned to the correct VLAN. | **show vlan [brief]**<br>**show interfaces switchport**<br>**ping** |
| Switch Trunk Port Issues | • Ensure trunks are configured correctly.<br>• Ensure port is a trunk port and enabled. | **show interface trunk**<br>**show running-config** |
| Switch Access Port Issues | • Assign correct VLAN to access port.<br>• Ensure port is an access port and enabled.<br>• Host is incorrectly configured in the wrong subnet. | **show interfaces switchport**<br>**show running-config interface**<br>**ipconfig** |
| Router Configuration Issues | • Router subinterface IPv4 address is incorrectly configured.<br>• Router subinterface is assigned to the VLAN ID. | **show ip interface brief**<br>**show interfaces** |

### *Troubleshoot Inter-VLAN Routing Scenario*

Examples of some of these inter-VLAN routing problems will now be covered in more detail. This topology will be used for all of these issues.




| Router R1 Subinterfaces | | |
|---|---|---|
| **Subinterface** | **VLAN** | **IP Address** |
| G0/0/0.10 | 10 | 192.168.10.1/24 |
| G0/0/0.20 | 20 | 192.168.20.1/24 |
| G0/0/0.30 | 99 | 192.168.99.1/24 |

### *Missing VLANs*

An inter-VLAN connectivity issue could be caused by a missing VLAN. The VLAN could be missing if it was not created, it was accidently deleted, or it is not allowed on the trunk link.

When a VLAN is deleted, any ports assigned to that VLAN become inactive. They remain associated with the VLAN (and thus inactive) until you assign them to a new VLAN or recreate the missing VLAN. Recreating the missing VLAN would automatically reassign the hosts to it.

Use the **show interface** *interface-id* **switchport** command to verify the VLAN membership of the port.

```
S1(config)# do show interface fa0/6 switchport
Name: Fa0/6
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
Voice VLAN: none
(Output omitted)
```

### *Switch Trunk Port Issues*

Another issue for inter-VLAN routing includes misconfigured switch ports. In a legacy inter-VLAN solution, this could be caused when the connecting router port is not assigned to the correct VLAN.

However, with a router-on-a-stick solution, the most common cause is a misconfigured trunk port.

- Verify that the port connecting to the router is correctly configured as a trunk link using the **show interface trunk** command.
- If that port is missing from the output, examine the configuration of the port with the **show running-config interface X** command to see how the port is configured.

### *Switch Access Port Issues*

When a problem is suspected with a switch access port configuration, use verification commands to examine the configuration and identify the problem.

A common indicator of this issue is the PC having the correct address configuration (IP Address, Subnet Mask, Default Gateway), but being unable to ping its default gateway.

- Use the **show vlan brief**, **show interface X switchport** or **show running-config interface X** command to verify the interface VLAN assignment.

***Switch Trunk Port Issues***

```
S1# show interface trunk
Port        Mode             Encapsulation  Status        Native vlan
Fa0/1       on               802.1q         trunking      1
Port        Vlans allowed on trunk
Fa0/1       1-4094
Port        Vlans allowed and active in management domain
Fa0/1       1,10,20,99
Port        Vlans in spanning tree forwarding state and not pruned
Fa0/1       1,10,20,99
S1#
```

***Switch Access Port Issues***

```
S1# show interface fa0/6 switchport
Name: Fa0/6
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
Voice VLAN: none
```

### *Router Configuration Issues*

Router-on-a-stick configuration problems are usually related to subinterface misconfigurations.

- Verify the subinterface status using the **show ip interface brief** command.
- Verify which VLANs each of the subinterfaces is on. To do so, the **show interfaces** command is useful but it generates a great deal of additional unrequired output. The command output can be reduced using IOS command filters. In this example, use the **include** keyword to identify that only lines containing the letters “Gig” or “802.1Q”

```
R1# show interfaces | include Gig|802.1Q
GigabitEthernet0/0/0 is administratively down, line protocol is down
GigabitEthernet0/0/1 is up, line protocol is up
  Encapsulation 802.1Q Virtual LAN, Vlan ID  1., loopback not set
GigabitEthernet0/0/1.10 is up, line protocol is up
  Encapsulation 802.1Q Virtual LAN, Vlan ID  100.
GigabitEthernet0/0/1.20 is up, line protocol is up
  Encapsulation 802.1Q Virtual LAN, Vlan ID  20.
GigabitEthernet0/0/1.99 is up, line protocol is up
  Encapsulation 802.1Q Virtual LAN, Vlan ID  99.
R1#
```

**Laboratory Exercise: Inter-VLAN Configuration**

**Online Chapter Exam**

**Practical Exam**

## Module 5: STP Concepts

***Objectives:***

At the end of this module, the student should be able to:

- Explain common problems in a redundant, L2 switched network;
- Explain how STP operates in a simple switched network;
- Explain how Rapid PVST+ operates.

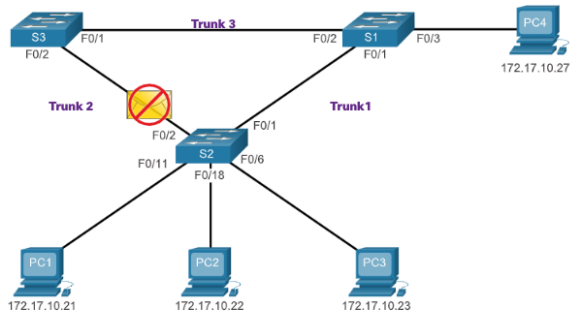
### Purpose of STP

***Redundancy in Layer 2 Switched Networks***

- This topic covers the causes of loops in a Layer 2 network and briefly explains how spanning tree protocol works. Redundancy is an important part of the hierarchical design for eliminating single points of failure and preventing disruption of network services to users. Redundant networks require the addition of physical paths, but logical redundancy must also be part of the design. Having alternate physical paths for data to traverse the network makes it possible for users to access network resources, despite path disruption. However, redundant paths in a switched Ethernet network may cause both physical and logical Layer 2 loops.
- Ethernet LANs require a loop-free topology with a single path between any two devices. A loop in an Ethernet LAN can cause continued propagation of Ethernet frames until a link is disrupted and breaks the loop.

***Spanning Tree protocol***

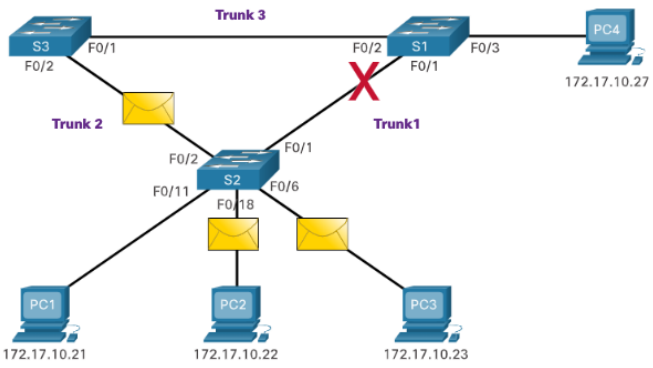
- Spanning Tree Protocol (STP) is a loop-prevention network protocol that allows for redundancy while creating a loop-free Layer 2 topology.
- STP logically blocks physical loops in a Layer 2 network, preventing frames from circling the network forever.




S2 drops the frame because it received it on a blocked port.

***STP Recalculation***

STP compensates for a failure in the network by recalculating and opening up previously blocked ports.



### *Issues with Redundant Switch Links*

- Path redundancy provides multiple network services by eliminating the possibility of a single point of failure. When multiple paths exist between two devices on an Ethernet network, and there is no spanning tree implementation on the switches, a Layer 2 loop occurs. A Layer 2 loop can result in MAC address table instability, link saturation, and high CPU utilization on switches and end-devices, resulting in the network becoming unusable.
- Layer 2 Ethernet does not include a mechanism to recognize and eliminate endlessly looping frames. Both IPv4 and IPv6 include a mechanism that limits the number of times a Layer 3 networking device can retransmit a packet. A router will decrement the TTL (Time to Live) in every IPv4 packet, and the Hop Limit field in every IPv6 packet. When these fields are decremented to 0, a router will drop the packet. Ethernet and Ethernet switches have no comparable mechanism for limiting the number of times a switch retransmits a Layer 2 frame. STP was developed specifically as a loop prevention mechanism for Layer 2 Ethernet.

### *Layer 2 Loops*

- Without STP enabled, Layer 2 loops can form, causing broadcast, multicast and unknown unicast frames to loop endlessly. This can bring down a network quickly.
- When a loop occurs, the MAC address table on a switch will constantly change with the updates from the broadcast frames, which results in MAC database instability. This can cause high CPU utilization, which makes the switch unable to forward frames.
- An unknown unicast frame is when the switch does not have the destination MAC address in its MAC address table and must forward the frame out all ports, except the ingress port.

### *Broadcast Storm*

- A broadcast storm is an abnormally high number of broadcasts overwhelming the network during a specific amount of time. Broadcast storms can disable a network within seconds by overwhelming switches and end devices. Broadcast storms can be caused by a hardware problem such as a faulty NIC or from a Layer 2 loop in the network.
- Layer 2 broadcasts in a network, such as ARP Requests are very common. Layer 2 multicasts are typically forwarded the same way as a broadcast by the switch. IPv6 packets are never forwarded as a Layer 2 broadcast, ICMPv6 Neighbor Discovery uses Layer 2 multicasts.
- A host caught in a Layer 2 loop is not accessible to other hosts on the network. Additionally, due to the constant changes in its MAC address table, the switch does not know out of which port to forward unicast frames.
- To prevent these issues from occurring in a redundant network, some type of spanning tree must be enabled on the switches. Spanning tree is enabled, by default, on Cisco switches to prevent Layer 2 loops from occurring.

### *The Spanning Tree Algorithm*

- STP is based on an algorithm invented by Radia Perlman while working for Digital Equipment Corporation and published in the 1985 paper "An Algorithm for Distributed Computation of a Spanning Tree in an Extended LAN." Her spanning tree algorithm (STA) creates a loop-free topology by selecting a single root bridge where all other switches determine a single least-cost path.
- STP prevents loops from occurring by configuring a loop-free path through the network using strategically placed "blocking-state" ports. The switches running STP are able to

compensate for failures by dynamically unblocking the previously blocked ports and permitting traffic to traverse the alternate paths.

How does the STA create a loop-free topology?

- Selecting a Root Bridge: This bridge (switch) is the reference point for the entire network to build a spanning tree around.
- Block Redundant Paths: STP ensures that there is only one logical path between all destinations on the network by intentionally blocking redundant paths that could cause a loop. When a port is blocked, user data is prevented from entering or leaving that port.
- Create a Loop-Free Topology: A blocked port has the effect of making that link a non-forwarding link between the two switches. This creates a topology where each switch has only a single path to the root bridge, similar to branches on a tree that connect to the root of the tree.

Recalculate in case of Link Failure: The physical paths still exist to provide redundancy, but these paths are disabled to prevent the loops from occurring. If the path is ever needed to compensate for a network cable or switch failure, STP recalculates the paths and unblocks the necessary ports to allow the redundant path to become active. STP recalculations can also occur any time a new switch or new inter-switch link is added to the network.

**STP Operations**

***Steps to a Loop-Free Topology***

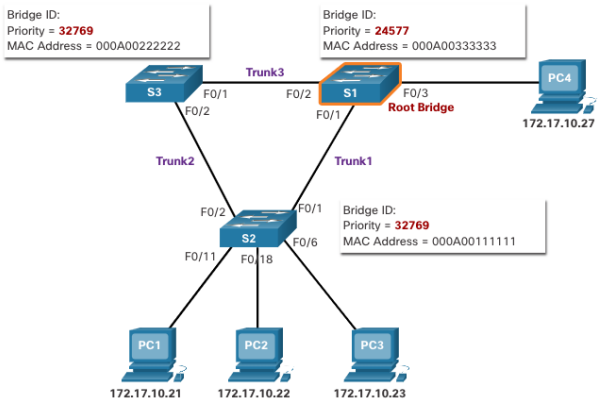
Using the STA, STP builds a loop-free topology in a four-step process:

1. Elect the root bridge.
2. Elect the root ports.
3. Elect designated ports.
4. Elect alternate (blocked) ports.

- During STA and STP functions, switches use Bridge Protocol Data Units (BPDUs) to share information about themselves and their connections. BPDUs are used to elect the root bridge, root ports, designated ports, and alternate ports.
- Each BPDU contains a bridge ID (BID) that identifies which switch sent the BPDU. The BID is involved in making many of the STA decisions including root bridge and port roles.
- The BID contains a priority value, the MAC address of the switch, and an extended system ID. The lowest BID value is determined by the combination of these three fields.
- **Bridge Priority:** The default priority value for all Cisco switches is the decimal value 32768. The range is 0 to 61440 in increments of 4096. A lower bridge priority is preferable. A bridge priority of 0 takes precedence over all other bridge priorities.
- **Extended System ID:** The extended system ID value is a decimal value added to the bridge priority value in the BID to identify the VLAN for this BPDU.
- **MAC address:** When two switches are configured with the same priority and have the same extended system ID, the switch having the MAC address with the lowest value, expressed in hexadecimal, will have the lower BID.

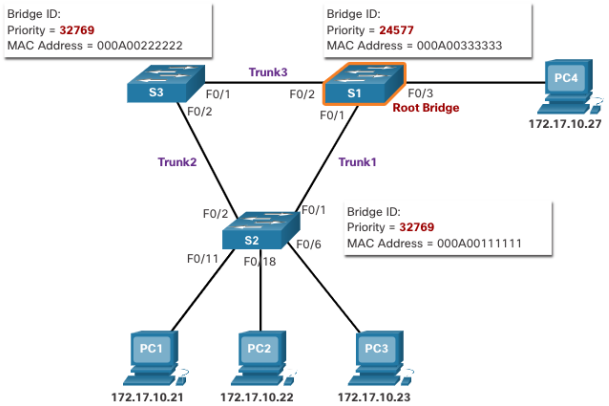
### *1. Elect the Root Bridge*

- The STA designates a single switch as the root bridge and uses it as the reference point for all path calculations. Switches exchange BPDUs to build the loop-free topology beginning with selecting the root bridge.
- All switches in the broadcast domain participate in the election process. After a switch boots, it begins to send out BPDU frames every two seconds. These BPDU frames contain the BID of the sending switch and the BID of the root bridge, known as the Root ID.
- The switch with the lowest BID will become the root bridge. At first, all switches declare themselves as the root bridge with their own BID set as the Root ID. Eventually, the switches learn through the exchange of BPDUs which switch has the lowest BID and will agree on one root bridge.




### *Impact of Default BIDs*

- Because the default BID is 32768, it is possible for two or more switches to have the same priority. In this scenario, where the priorities are the same, the switch with the lowest MAC address will become the root bridge. The administrator should configure the desired root bridge switch with a lower priority.
- In the figure, all switches are configured with the same priority of 32769. Here the MAC address becomes the deciding factor as to which switch becomes the root bridge. The switch with the lowest hexadecimal MAC address value is the preferred root bridge. In this example, S2 has the lowest value for its MAC address and is elected as the root bridge for that spanning tree instance.
- **Note**: The priority of all the switches is 32769. The value is based on the 32768 default bridge priority and the extended system ID (VLAN 1 assignment) associated with each switch (32768+1).



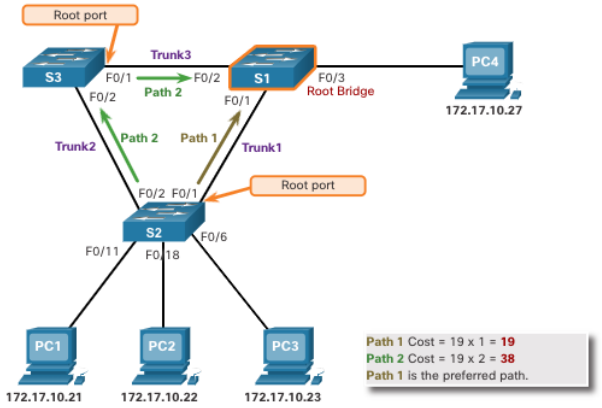
### *Determine the Root Path Cost*

- When the root bridge has been elected for a given spanning tree instance, the STA starts determining the best paths to the root bridge from all destinations in the broadcast domain. The path information, known as the internal root path cost, is determined by the sum of all the individual port costs along the path from the switch to the root bridge.
- When a switch receives the BPDU, it adds the ingress port cost of the segment to determine its internal root path cost.
- The default port costs are defined by the speed at which the port operates. The table shows the default port costs suggested by IEEE. Cisco switches by default use the values as defined by the IEEE 802.1D standard, also known as the short path cost, for both STP and RSTP.
- Although switch ports have a default port cost associated with them, the port cost is configurable. The ability to configure individual port costs gives the administrator the flexibility to manually control the spanning tree paths to the root bridge.

| Link Speed | STP Cost: IEEE 802.1D-1998 | RSTP Cost: IEEE 802.1w-2004 |
|---|---|---|
| 10 Gbps | 2 | 2,000 |
| 1 Gbps | 4 | 20,000 |
| 100 Mbps | 19 | 200,000 |
| 10 Mbps | 100 | 2,000,000 |

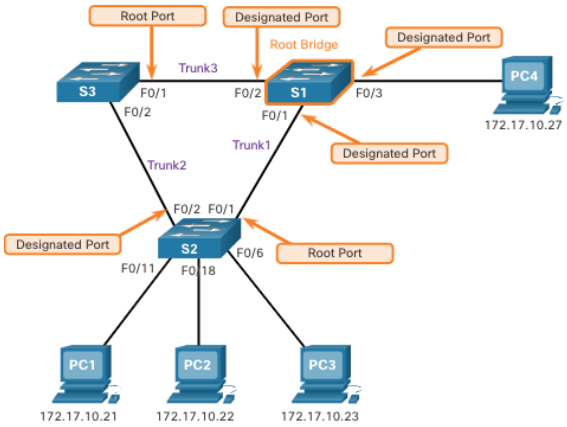
### *2. Elect the Root Ports*

- After the root bridge has been determined, the STA algorithm is used to select the root port. Every non-root switch will select one root port. The root port is the port closest to the root bridge in terms of overall cost to the root bridge. This overall cost is known as the internal root path cost.
- The internal root path cost is equal to the sum of all the port costs along the path to the root bridge, as shown in the figure. Paths with the lowest cost become preferred, and all other redundant paths are blocked. In the example, the internal root path cost from S2 to the root bridge S1 over path 1 is 19 while the internal root path cost over path 2 is 38. Because path 1 has a lower overall path cost to the root bridge, it is the preferred path and F0/1 becomes the root port on S2.



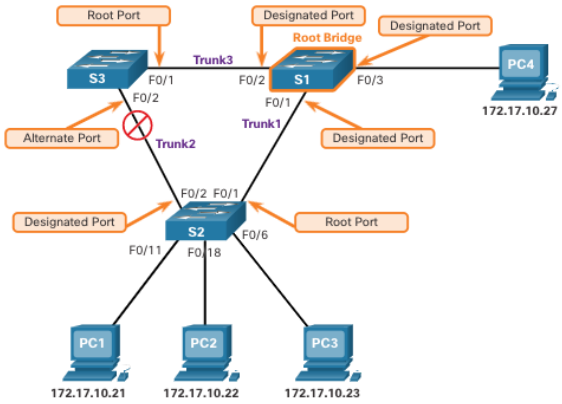
### 3. *Elect Designated Ports*

- Every segment between two switches will have one designated port. The designated port is a port on the segment that has the internal root path cost to the root bridge. In other words, the designated port has the best path to receive traffic leading to the root bridge.
- What is not a root port or a designated port becomes an alternate or blocked port.
- All ports on the root bridge are designated ports.
- If one end of a segment is a root port, the other end is a designated port.
- All ports attached to end devices are designated ports.
- On segments between two switches where neither of the switches is the root bridge, the port on the switch with the least-cost path to the root bridge is a designated port.




### 4. *Elect Alternate (Blocked) Ports*

If a port is not a root port or a designated port, then it becomes an alternate (or backup) port. Alternate ports are in discarding or blocking state to prevent loops. In the figure, the STA has configured port F0/2 on S3 in the alternate role. Port F0/2 on S3 is in the blocking state and will not forward Ethernet frames. All other inter-switch ports are in forwarding state. This is the loop-prevention part of STP.

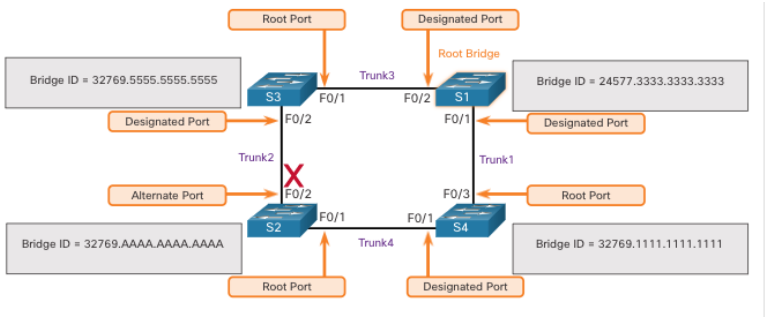



### *Elect a Root Port from Multiple Equal-Cost Paths*

When a switch has multiple equal-cost paths to the root bridge, the switch will determine a port using the following criteria:

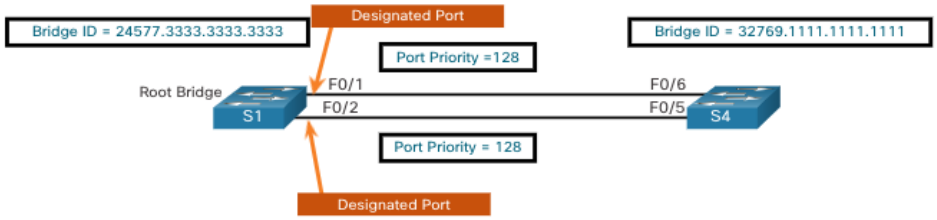
- Lowest sender BID
- Lowest sender port priority
- Lowest sender port ID

**Lowest Sender BID:** This topology has four switches with switch S1 as the root bridge. Port F0/1 on switch S3 and port F0/3 on switch S4 have been selected as root ports because they have the root path cost to the root bridge for their respective switches. S2 has two ports, F0/1 and F0/2 with equal cost paths to the root bridge. The bridge IDs of S3 and S4, will be used to break the tie. This is known as the sender's BID. S3 has a BID of 32769.5555.5555.5555 and S4 has a BID of 32769.1111.1111.1111. Because S4 has a lower BID, the F0/1 port of S2, which is the port connected to S4, will be the root port.

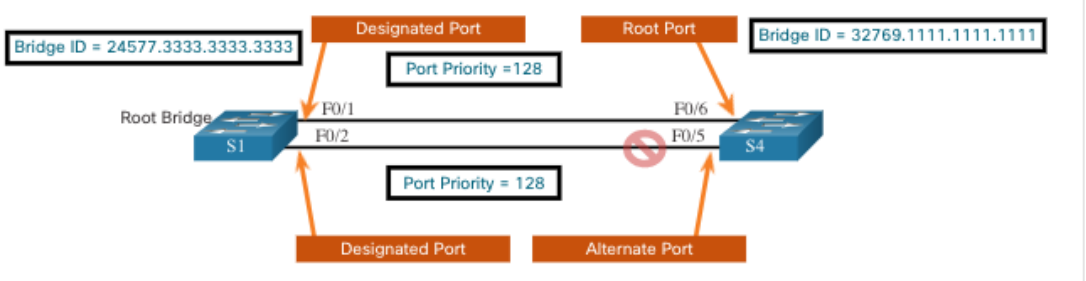



**Lowest Sender Port Priority:** This topology has two switches which are connected with two equal-cost paths between them. S1 is the root bridge, so both of its ports are designated ports.

- S4 has two ports with equal-cost paths to the root bridge. Because both ports are connected to the same switch, the sender's BID (S1) is equal. So the first step is a tie.
- Next, is the sender's (S1) port priority. The default port priority is 128, so both ports on S1 have the same port priority. This is also a tie. However, if either port on S1 was configured with a lower port priority, S4 would put its adjacent port in forwarding state. The other port on S4 would be a blocking state.




- **Lowest Sender Port ID:** The last tie-breaker is the lowest sender's port ID. Switch S4 has received BPDUs from port F0/1 and port F0/2 on S1. The decision is based on the sender's port ID, not the receiver's port ID. Because the port ID of F0/1 on S1 is lower than port F0/2, the port F0/6 on switch S4 will be the root port. This is the port on S4 that is connected to the F0/1 port on S1.
- Port F0/5 on S4 will become an alternate port and placed in the blocking state.



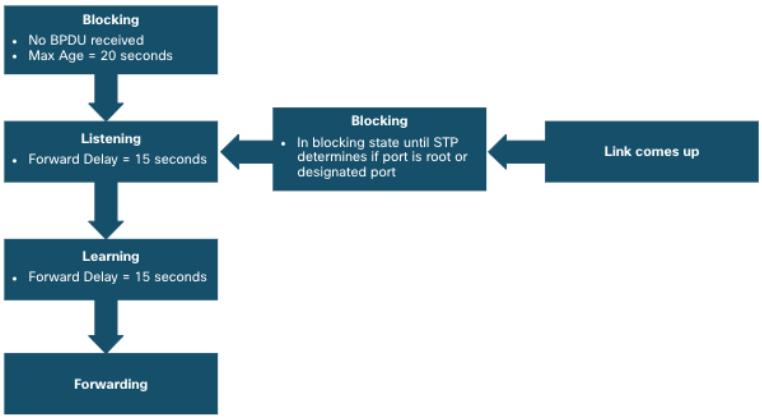
### *STP Timers and Port States*

**STP convergence requires three timers, as follows:**

- **Hello Timer** -The hello time is the interval between BPDUs. The default is 2 seconds but can be modified to between 1 and 10 seconds.
- **Forward Delay Timer** -The forward delay is the time that is spent in the listening and learning state. The default is 15 seconds but can be modified to between 4 and 30 seconds.
- **Max Age Timer** -The max age is the maximum length of time that a switch waits before attempting to change the STP topology. The default is 20 seconds but can be modified to between 6 and 40 seconds.

**Note**: The default times can be changed on the root bridge, which dictates the value of these timers for the STP domain.

STP facilitates the logical loop-free path throughout the broadcast domain. The spanning tree is determined through the information learned by the exchange of the BPDU frames between the interconnected switches. If a switch port transitions directly from the blocking state to the forwarding state without information about the full topology during the transition, the port can temporarily create a data loop. For this reason, STP has five ports states, four of which are operational port states as shown in the figure. The disabled state is considered non-operational.




### *Operational Details of Each Port State*

The table summarizes the operational details of each port state.

| Port State | BPDU | MAC Address Table | Forwarding Data Frames |
|---|---|---|---|
| Blocking | Receive only | No update | No |
| Listening | Receive and send | No update | No |
| Learning | Receive and send | Updating table | No |
| Forwarding | Receive and send | Updating table | Yes |
| Disabled | None sent or received | No update | No |

### *Per-VLAN Spanning Tree*

STP can be configured to operate in an environment with multiple VLANs. In Per-VLAN Spanning Tree (PVST) versions of STP, there is a root bridge elected for each spanning tree instance. This makes it possible to have different root bridges for different sets of VLANs. STP operates a separate instance of STP for each individual VLAN. If all ports on all switches are members of VLAN 1, then there is only one spanning tree instance.

## Evolution of STP

### *Different Versions of STP*

- Many professionals generically use spanning tree and STP to refer to the various implementations of spanning tree, such as Rapid Spanning Tree Protocol (RSTP) and Multiple Spanning Tree Protocol (MSTP). In order to communicate spanning tree concepts correctly, it is important to refer to the implementation or standard of spanning tree in context.
- The latest IEEE documentation on spanning tree (IEEE-802-1D-2004) says, "STP has now been superseded by the Rapid Spanning Tree Protocol (RSTP)."The IEEE uses "STP" to refer to the original implementation of spanning tree and "RSTP" to describe the version of spanning tree specified in IEEE-802.1D-2004.
- Because the two protocols share much of the same terminology and methods for the loop-free path, the primary focus will be on the current standard and the Cisco proprietary implementations of STP and RSTP.
- Cisco switches running IOS 15.0 or later, run PVST+ by default. This version incorporates many of the specifications of IEEE 802.1D-2004, such as alternate ports in place of the former non-designated ports. Switches must be explicitly configured for rapid spanning tree mode in order to run the rapid spanning tree protocol.

| STP Variety | Description |
|---|---|
| STP | This is the original IEEE 802.1D version (802.1D-1998 and earlier) that provides a loop-free topology in a network with redundant links. Also called Common Spanning Tree (CST), it assumes one spanning tree instance for the entire bridged network, regardless of the number of VLANs. |
| PVST+ | Per-VLAN Spanning Tree (PVST+) is a Cisco enhancement of STP that provides a separate 802.1D spanning tree instance for each VLAN configured in the network. PVST+ supports PortFast, UplinkFast, BackboneFast, BPDU guard, BPDU filter, root guard, and loop guard. |
| 802.1D-2004 | This is an updated version of the STP standard, incorporating IEEE 802.1w. |
| RSTP | Rapid Spanning Tree Protocol (RSTP) or IEEE 802.1w is an evolution of STP that provides faster convergence than STP. |
| Rapid PVST+ | This is a Cisco enhancement of RSTP that uses PVST+ and provides a separate instance of 802.1w per VLAN. Each separate instance supports PortFast, BPDU guard, BPDU filter, root guard, and loop guard. |
| MSTP | Multiple Spanning Tree Protocol (MSTP) is an IEEE standard inspired by the earlier Cisco proprietary Multiple Instance STP (MISTP) implementation. MSTP maps multiple VLANs into the same spanning tree instance. |
| MST | Multiple Spanning Tree (MST) is the Cisco implementation of MSTP, which provides up to 16 instances of RSTP and combines many VLANs with the same physical and logical topology into a common RSTP instance. Each instance supports PortFast, BPDU guard, BPDU filter, root guard, and loop guard. |

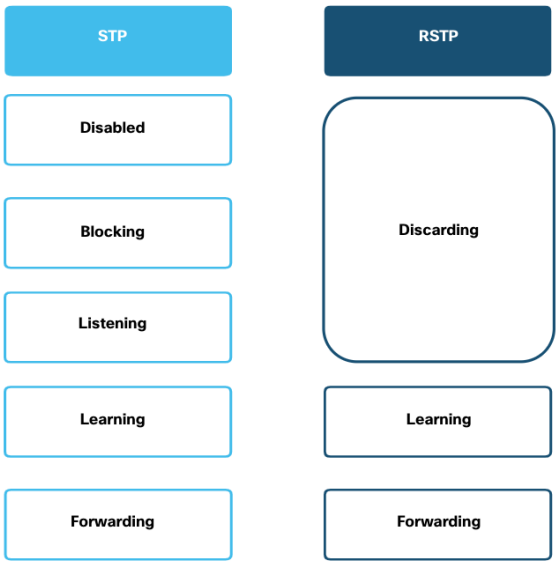
### *RSTP Concepts*

- RSTP (IEEE 802.1w) supersedes the original 802.1D while retaining backward compatibility. The 802.1w STP terminology remains primarily the same as the original IEEE 802.1D STP terminology. Most parameters have been left unchanged. Users that are familiar with the original STP standard can easily configure RSTP. The same spanning tree algorithm is used for both STP and RSTP to determine port roles and topology.
- RSTP increases the speed of the recalculation of the spanning tree when the Layer 2 network topology changes. RSTP can achieve much faster convergence in a properly configured network, sometimes in as little as a few hundred milliseconds. If a port is configured to be an alternate port it can immediately change to a forwarding state without waiting for the network to converge.

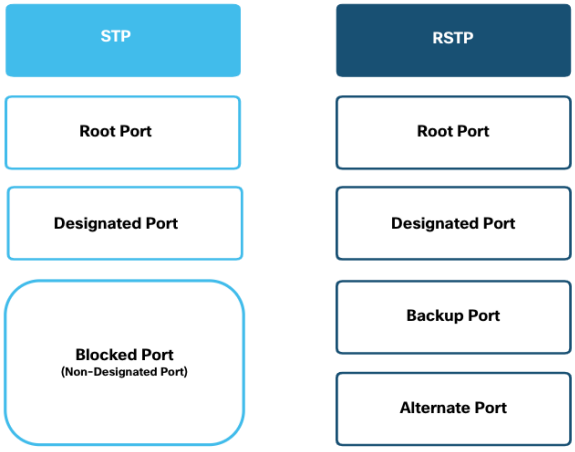
**Note**: Rapid PVST+ is the Cisco implementation of RSTP on a per-VLAN basis. With Rapid PVST+ an independent instance of RSTP runs for each VLAN.

### *RSTP Port States and Port Roles*

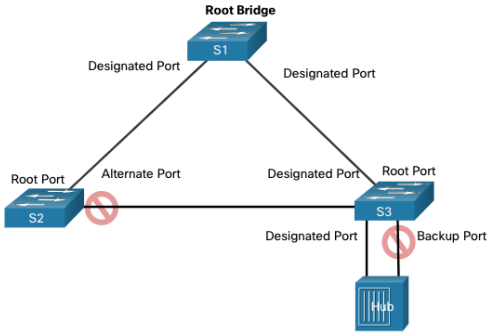
There are only three port states in RSTP that correspond to the three possible operational states in STP. The 802.1D disabled, blocking, and listening states are merged into a unique 802.1w discarding state.




Root ports and designated ports are the same for both STP and RSTP. However, there are two RSTP port roles that correspond to the blocking state of STP. In STP, a blocked port is defined as not being the designated or root port. RSTP has two port roles for this purpose.



The alternate port has an alternate path to the root bridge. The backup port is a backup to a shared medium, such as a hub. A backup port is less common because hubs are now considered legacy devices.




***PortFast and BPDU Guard***

- When a device is connected to a switch port or when a switch powers up, the switch port goes through both the listening and learning states, each time waiting for the Forward Delay timer to expire. This delay is 15 seconds for each state for a total of 30 seconds. This can present a problem for DHCP clients trying to discover a DHCP server because the DHCP process may timeout. The result is that an IPv4 client will not receive a valid IPv4 address.
- When a switch port is configured with PortFast, that port transitions from blocking to forwarding state immediately, avoiding the 30 second delay. You can use PortFast on access ports to allow devices connected to these ports to access the network immediately. PortFast should only be used on access ports. If you enable PortFast on a port connecting to another switch, you risk creating a spanning tree loop.
- A PortFast-enabled switch port should never receive BPDUs because that would indicate that switch is connected to the port, potentially causing a spanning tree loop. Cisco switches support a feature called BPDU guard. When enabled, it immediately puts the switch port in an errdisabled (error-disabled) state upon receipt of any BPDU. This protects against potential loops by effectively shutting down the port. The administrator must manually put the interface back into service.

***Alternatives to STP***

- Over the years, organizations required greater resiliency and availability in the LAN. Ethernet LANs went from a few interconnected switches connected to a single router, to a sophisticated hierarchical network design including access, distribution and core layer switches.
- Depending on the implementation, Layer 2 may include not only the access layer, but also the distribution or even the core layers. These designs may include hundreds of switches, with hundreds or even thousands of VLANs. STP has adapted to the added redundancy and complexity with enhancements, as part of RSTP and MSTP.
- An important aspect to network design is fast and predictable convergence when there is a failure or change in the topology. Spanning tree does not offer the same efficiencies and predictabilities provided by routing protocols at Layer 3.
- Layer 3 routing allows for redundant paths and loops in the topology, without blocking ports. For this reason, some environments are transitioning to Layer 3 everywhere except where devices connect to the access layer switch. In other words, the connections between access layer switches and distribution switches would be Layer 3 instead of Layer 2.

**Laboratory Exercise: Root Bridge and Designated Port Simulation**

## Module 6: EtherChannel

***Objectives:***

At the end of this module, the student should be able to:

- Describe EtherChannel technology;
- Configure EtherChannel;
- Troubleshoot EtherChannel.

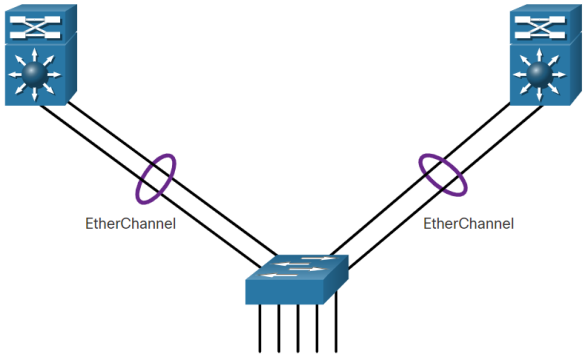
### EtherChannel Operation

#### *Link Aggregation*

- There are scenarios in which more bandwidth or redundancy between devices is needed than what can be provided by a single link. Multiple links could be connected between devices to increase bandwidth. However, Spanning Tree Protocol (STP), which is enabled on Layer 2 devices like Cisco switches by default, will block redundant links to prevent switching loops.
- A link aggregation technology is needed that allows redundant links between devices that will not be blocked by STP. That technology is known as EtherChannel.
- EtherChannel is a link aggregation technology that groups multiple physical Ethernet links together into one single logical link. It is used to provide fault-tolerance, load sharing, increased bandwidth, and redundancy between switches, routers, and servers.
- EtherChannel technology makes it possible to combine the number of physical links between the switches to increase the overall speed of switch-to-switch communication.

#### *EtherChannel*

EtherChannel technology was originally developed by Cisco as a LAN switch-to-switch technique of grouping several Fast Ethernet or Gigabit Ethernet ports into one logical channel.
When an EtherChannel is configured, the resulting virtual interface is called a port channel. The physical interfaces are bundled together into a port channel interface, as shown in the figure.




#### *Advantages of EtherChannel*

EtherChannel technology has many advantages, including the following:

- Most configuration tasks can be done on the EtherChannel interface instead of on each individual port, ensuring configuration consistency throughout the links.
- EtherChannel relies on existing switch ports. There is no need to upgrade the link to a faster and more expensive connection to have more bandwidth.
- Load balancing takes place between links that are part of the same EtherChannel.
- EtherChannel creates an aggregation that is seen as one logical link. When several EtherChannel bundles exist between two switches, STP may block one of the bundles to prevent switching loops. When STP blocks one of the redundant links, it blocks the entire

EtherChannel. This blocks all the ports belonging to that EtherChannel link. Where there is only one EtherChannel link, all physical links in the EtherChannel are active because STP sees only one (logical) link.
- EtherChannel provides redundancy because the overall link is seen as one logical connection. Additionally, the loss of one physical link within the channel does not create a change in the topology.

### *Implementation Restrictions*

EtherChannel has certain implementation restrictions, including the following:
- Interface types cannot be mixed. For example, Fast Ethernet and Gigabit Ethernet cannot be mixed within a single EtherChannel.
- Currently each EtherChannel can consist of up to eight compatibly-configured Ethernet ports. EtherChannel provides full-duplex bandwidth up to 800 Mbps (Fast EtherChannel) or 8 Gbps (Gigabit EtherChannel) between one switch and another switch or host.
- The Cisco Catalyst 2960 Layer 2 switch currently supports up to six EtherChannels.
- The individual EtherChannel group member port configuration must be consistent on both devices. If the physical ports of one side are configured as trunks, the physical ports of the other side must also be configured as trunks within the same native VLAN. Additionally, all ports in each EtherChannel link must be configured as Layer 2 ports.
- Each EtherChannel has a logical port channel interface. A configuration applied to the port channel interface affects all physical interfaces that are assigned to that interface.

### *AutoNegotiation Protocols*

EtherChannels can be formed through negotiation using one of two protocols, Port Aggregation Protocol (PAgP) or Link Aggregation Control Protocol (LACP). These protocols allow ports with similar characteristics to form a channel through dynamic negotiation with adjoining switches.
**Note**: It is also possible to configure a static or unconditional EtherChannel without PAgP or LACP.

### *PAgP Operation*

PAgP (pronounced “Pag - P”) is a Cisco-proprietary protocol that aids in the automatic creation of EtherChannel links. When an EtherChannel link is configured using PAgP, PAgP packets are sent between EtherChannel-capable ports to negotiate the forming of a channel. When PAgP identifies matched Ethernet links, it groups the links into an EtherChannel. The EtherChannel is then added to the spanning tree as a single port.
When enabled, PAgP also manages the EtherChannel. PAgP packets are sent every 30 seconds. PAgP checks for configuration consistency and manages link additions and failures between two switches. It ensures that when an EtherChannel is created, all ports have the same type of configuration.
**Note**: In EtherChannel, it is mandatory that all ports have the same speed, duplex setting, and VLAN information. Any port modification after the creation of the channel also changes all other channel ports.

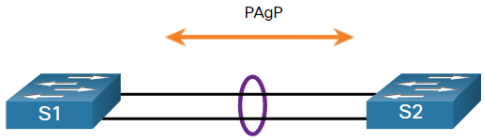
PAgP helps create the EtherChannel link by detecting the configuration of each side and ensuring that links are compatible so that the EtherChannel link can be enabled when needed.

The modes for PAgP as follows:

- **On** - This mode forces the interface to channel without PAgP. Interfaces configured in the on mode do not exchange PAgP packets.
- **PAgP desirable** - This PAgP mode places an interface in an active negotiating state in which the interface initiates negotiations with other interfaces by sending PAgP packets.
- **PAgP auto** - This PAgP mode places an interface in a passive negotiating state in which the interface responds to the PAgP packets that it receives but does not initiate PAgP negotiation.

The modes must be compatible on each side. If one side is configured to be in auto mode, it is placed in a passive state, waiting for the other side to initiate the EtherChannel negotiation. If the other side is also set to auto, the negotiation never starts and the EtherChannel does not form. If all modes are disabled by using the **no** command, or if no mode is configured, then the EtherChannel is disabled. The on mode manually places the interface in an EtherChannel, without any negotiation. It works only if the other side is also set to on. If the other side is set to negotiate parameters through PAgP, no EtherChannel forms, because the side that is set to on mode does not negotiate. No negotiation between the two switches means there is no checking to make sure that all the links in the EtherChannel are terminating on the other side, or that there is PAgP compatibility on the other switch.

***PAgP Mode Settings Example***




The table shows the various combination of PAgP modes on S1 and S2 and the resulting channel establishment outcome.

| S1 | S2 | Channel Establishment |
|---|---|---|
| On | On | Yes |
| On | Desirable/Auto | No |
| Desirable | Desirable | Yes |
| Desirable | Auto | Yes |
| Auto | Desirable | Yes |
| Auto | Auto | No |

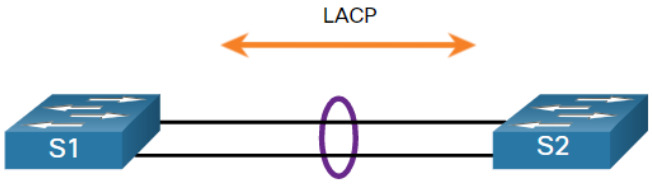
***LACP Operation***

LACP is part of an IEEE specification (802.3ad) that allows several physical ports to be bundled to form a single logical channel. LACP allows a switch to negotiate an automatic bundle by sending LACP packets to the other switch. It performs a function similar to PAgP with Cisco EtherChannel. Because LACP is an IEEE standard, it can be used to facilitate EtherChannels in multivendor environments. On Cisco devices, both protocols are supported.

LACP provides the same negotiation benefits as PAgP. LACP helps create the EtherChannel link by detecting the configuration of each side and making sure that they are compatible so that the EtherChannel link can be enabled when needed. The modes for LACP are as follows:

- **On** - This mode forces the interface to channel without LACP. Interfaces configured in the on mode do not exchange LACP packets.
- **LACP active** - This LACP mode places a port in an active negotiating state. In this state, the port initiates negotiations with other ports by sending LACP packets.
- **LACP passive** - This LACP mode places a port in a passive negotiating state. In this state, the port responds to the LACP packets that it receives but does not initiate LACP packet negotiation.

***LACP Mode Settings Example***




The table shows the various combination of LACP modes on S1 and S2 and the resulting channel establishment outcome.

| S1 | S2 | Channel Establishment |
|---|---|---|
| On | On | Yes |
| On | Active/Passive | No |
| Active | Active | Yes |
| Active | Passive | Yes |
| Passive | Active | Yes |
| Passive | Passive | No |

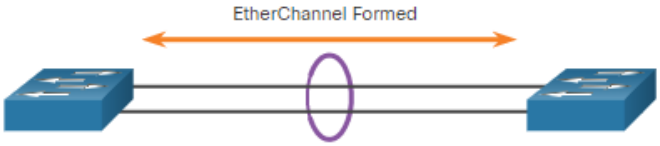
**Configure EtherChannel**

***Configuration Guidelines***

The following guidelines and restrictions are useful for configuring EtherChannel:

- **EtherChannel support** - All Ethernet interfaces must support EtherChannel with no requirement that interfaces be physically contiguous.
- **Speed and duplex** - Configure all interfaces in an EtherChannel to operate at the same speed and in the same duplex mode.
- **VLAN match** - All interfaces in the EtherChannel bundle must be assigned to the same VLAN or be configured as a trunk (shown in the figure).
- **Range of VLANs** - An EtherChannel supports the same allowed range of VLANs on all the interfaces in a trunking EtherChannel. If the allowed range of VLANs is not the same, the interfaces do not form an EtherChannel, even when they are set to **auto** or **desirable** mode.

- The figure shows a configuration that would allow an EtherChannel to form between S1 and S2.
- If these settings must be changed, configure them in port channel interface configuration mode. Any configuration that is applied to the port channel interface also affects individual interfaces. However, configurations that are applied to the individual interfaces do not affect the port channel interface. Therefore, making configuration changes to an interface that is part of an EtherChannel link may cause interface compatibility issues.
- The port channel can be configured in access mode, trunk mode (most common), or on a routed port.




| S1 Port Configurations | |
|---|---|
| Speed | 1 Gbps |
| Duplex | Full |
| VLAN | 10 |

| S2 Port Configurations | |
|---|---|
| Speed | 1 Gbps |
| Duplex | Full |
| VLAN | 10 |

***LACP Configuration Example***

Configuring EtherChannel with LACP requires the following three steps:

- **Step 1.** Specify the interfaces that compose the EtherChannel group using the **interface range** *interface* global configuration mode command. The **range** keyword allows you to select several interfaces and configure them all together.
- **Step 2.** Create the port channel interface with the **channel-group** *identifier* **mode active** command in interface range configuration mode. The identifier specifies a channel group number. The **mode active** keywords identify this as an LACP EtherChannel configuration.
- **Step3.** To change Layer 2 settings on the port channel interface, enter port channel interface configuration mode using the **interface port-channel** command, followed by the interface identifier. In the example, S1 is configured with an LACP EtherChannel. The port channel is configured as a trunk interface with the allowed VLANs specified.

```
S1(config)# interface range FastEthernet 0/1 - 2
S1(config-if-range)# channel-group 1 mode active
Creating a port-channel interface Port-channel 1
S1(config-if-range)# exit
S1(config-if)# interface port-channel 1
S1(config-if)# switchport mode trunk
S1(config-if)# switchport trunk allowed vlan 1,2,20
```

## Verify and Troubleshoot EtherChannel

### *Verify EtherChannel*

As always, when you configure devices in your network, you must verify your configuration. If there are problems, you will also need to be able to troubleshoot and fix them. There are a number of commands to verify an EtherChannel configuration:

- The **show interfaces port-channel** command displays the general status of the port channel interface.
- The **show etherchannel summary** command displays one line of information per port channel.
- The **show etherchannel port-channel** command displays information about a specific port channel interface.
- The **show interfaces etherchannel** command can provide information about the role of a physical member interface of the EtherChannel.

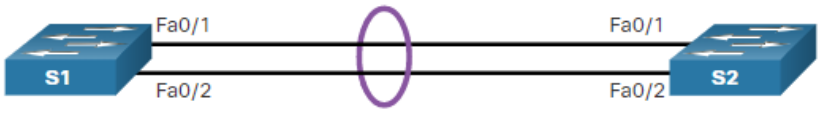
### *Common Issues with EtherChannel Configurations*

All interfaces within an EtherChannel must have the same configuration of speed and duplex mode, native and allowed VLANs on trunks, and access VLAN on access ports. Ensuring these configurations will significantly reduce network problems related to EtherChannel. Common EtherChannel issues include the following:

- Assigned ports in the EtherChannel are not part of the same VLAN, or not configured as trunks. Ports with different native VLANs cannot form an EtherChannel.
- Trunking was configured on some of the ports that make up the EtherChannel, but not all of them. It is not recommended that you configure trunking mode on individual ports that make up the EtherChannel. When configuring a trunk on an EtherChannel, verify the trunking mode on the EtherChannel.
- If the allowed range of VLANs is not the same, the ports do not form an EtherChannel even when PAgP is set to the **auto** or **desirable** mode.
- The dynamic negotiation options for PAgP and LACP are not compatibly configured on both ends of the EtherChannel.

### *Troubleshoot EtherChannel Example*

In the figure, interfaces F0/1 and F0/2 on switches S1 and S2 are connected with an EtherChannel. However, the EtherChannel is not operational.



**Step 1. View the EtherChannel Summary Information:** The output of the **show etherchannel summary** command indicates that the EtherChannel is down.

```
S1# show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      N - not in use, no aggregation
        f - failed to allocate aggregator
        M - not in use, minimum links not met
        m - not in use, port not aggregated due to minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port
        A - formed by Auto LAG
Number of channel-groups in use: 1
Number of aggregators:           1
Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
1      Po1(SD)         -      Fa0/1(D)    Fa0/2(D)
```

**Step 2. View Port Channel Configuration:** In the **show run | begin interface port-channel** output, more detailed output indicates that there are incompatible PAgP modes configured on S1 and S2.

```
S1# show run | begin interface port-channel
interface Port-channel1
 switchport trunk allowed vlan 1,2,20
 switchport mode trunk
!
interface FastEthernet0/1
 switchport trunk allowed vlan 1,2,20
 switchport mode trunk
 channel-group 1 mode on
!
interface FastEthernet0/2
 switchport trunk allowed vlan 1,2,20
 switchport mode trunk
 channel-group 1 mode on
!==================================
S2# show run | begin interface port-channel
interface Port-channel1
 switchport trunk allowed vlan 1,2,20
 switchport mode trunk
!
interface FastEthernet0/1
 switchport trunk allowed vlan 1,2,20
 switchport mode trunk
 channel-group 1 mode desirable
!
interface FastEthernet0/2
 switchport trunk allowed vlan 1,2,20
 switchport mode trunk
 channel-group 1 mode desirable
```

**Step 3: Correct the Misconfiguration:** To correct the issue, the PAgP mode on the EtherChannel is changed to desirable.

**Note**: EtherChannel and STP must interoperate. For this reason, the order in which EtherChannel-related commands are entered is important, which is why you see interface Port-Channel 1 removed and then re-added with the **channel-group** command, as opposed to directly changed. If one tries to change the configuration directly, STP errors cause the associated ports to go into blocking or errdisabled state.

```
S1(config)# no interface port-channel 1
S1(config)# interface range fa0/1 - 2
S1(config-if-range)# channel-group 1 mode desirable
Creating a port-channel interface Port-channel 1
S1(config-if-range)# no shutdown
S1(config-if-range)# exit
S1(config)# interface range fa0/1 - 2
S1(config-if-range)# channel-group 1 mode desirable
S1(config-if-range)# no shutdown
S1(config-if-range)# interface port-channel 1
S1(config-if)# switchport mode trunk
S1(config-if)# end
S1#
```

**Step 4. Verify EtherChannel is Operational:** The EtherChannel is now active as verified by the output of the **show etherchannel summary** command.

```
S1# show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      N - not in use, no aggregation
        f - failed to allocate aggregator
        M - not in use, minimum links not met
        m - not in use, port not aggregated due to minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port
        A - formed by Auto LAG
Number of channel-groups in use: 1
Number of aggregators:           1
Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
1      Po1(SU)         PAgP      Fa0/1(P)    Fa0/2(P)
```

**Laboratory Exercise: EtherChannel Configuration**

**Online Chapter Exam**

**Practical Exam**

## Module 7: DHCPv4

***Objectives:***

At the end of this module, the student should be able to:

- Explain how DHCPv4 operates in a small-to-medium sized business network;
- Configure a router as a DHCPv4 server;
- Configure a router as a DHCPv4 client.

### DHCPv4 Concepts

#### *DHCPv4 Server and Client*

- Dynamic Host Configuration Protocol v4 (DHCPv4) assigns IPv4 addresses and other network configuration information dynamically. Because desktop clients typically make up the bulk of network nodes, DHCPv4 is an extremely useful and timesaving tool for network administrators.
- A dedicated DHCPv4 server is scalable and relatively easy to manage. However, in a small branch or SOHO location, a Cisco router can be configured to provide DHCPv4 services without the need for a dedicated server. Cisco IOS software supports an optional, full-featured DHCPv4 server.
- The DHCPv4 server dynamically assigns, or leases, an IPv4 address from a pool of addresses for a limited period of time chosen by the server, or until the client no longer needs the address.
- Clients lease the information from the server for an administratively defined period. Administrators configure DHCPv4 servers to set the leases to time out at different intervals. The lease is typically anywhere from 24 hours to a week or more. When the lease expires, the client must ask for another address, although the client is typically reassigned the same address.

#### *DHCPv4 Operation*

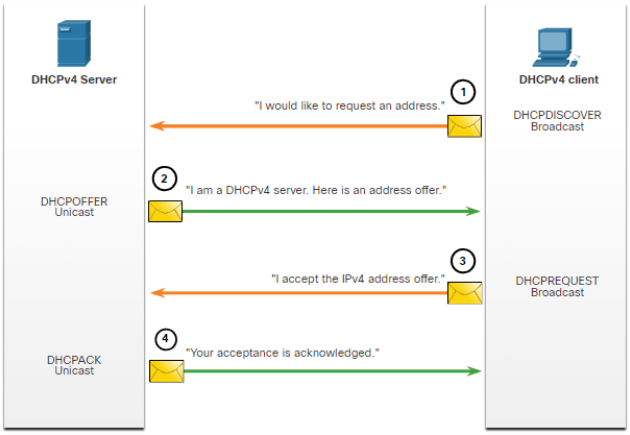
DHCPv4 works in a client/server mode. When a client communicates with a DHCPv4 server, the server assigns or leases an IPv4 address to that client.

- The client connects to the network with that leased IPv4 address until the lease expires. The client must contact the DHCP server periodically to extend the lease.
- This lease mechanism ensures that clients that move or power off do not keep addresses that they no longer need.
- When a lease expires, the DHCP server returns the address to the pool where it can be reallocated as necessary.

### *Steps to Obtain a Lease*

When the client boots (or otherwise wants to join a network), it begins a four-step process to obtain a lease:

1. DHCP Discover (DHCPDISCOVER)
2. DHCP Offer (DHCPOFFER)
3. DHCP Request (DHCPREQUEST)
4. DHCP Acknowledgment (DHCPACK)




### *Steps to Renew a Lease*

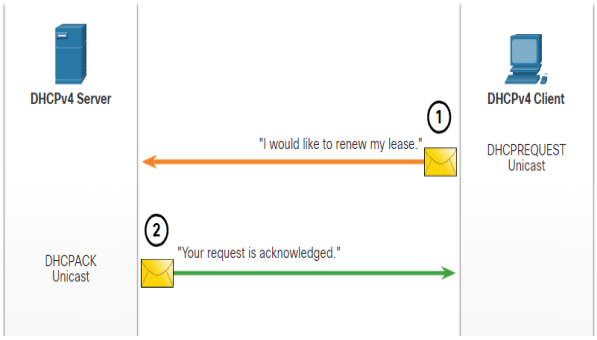
Prior to lease expiration, the client begins a two-step process to renew the lease with the DHCPv4 server, as shown in the figure:

**1. DHCP Request (DHCPREQUEST)**

Before the lease expires, the client sends a DHCPREQUEST message directly to the DHCPv4 server that originally offered the IPv4 address. If a DHCPACK is not received within a specified amount of time, the client broadcasts another DHCPREQUEST so that one of the other DHCPv4 servers can extend the lease.

**2. DHCP Acknowledgment (DHCPACK)**

On receiving the DHCPREQUEST message, the server verifies the lease information by returning a DHCPACK.

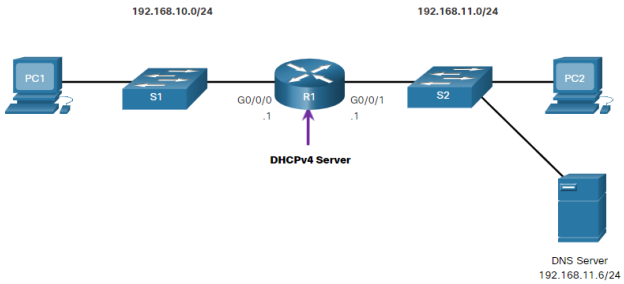



**Note**: These messages (primarily the DHCPOFFER and DHCPACK) can be sent as unicast or broadcast according to IETF RFC 2131.

## Configure a Cisco IOS DHCPv4 Server

### *Cisco IOS DHCPv4 Server*

Now you have a basic understanding of how DHCPv4 works and how it can make your job a bit easier. A Cisco router running Cisco IOS software can be configured to act as a DHCPv4 server. The Cisco IOS DHCPv4 server assigns and manages IPv4 addresses from specified address pools within the router to DHCPv4 clients.



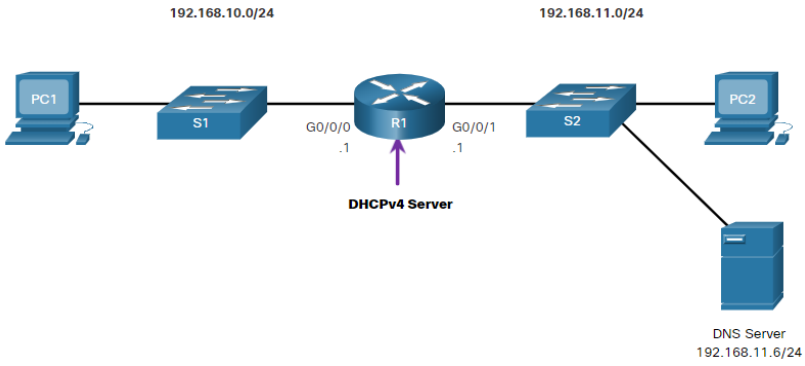
### *Steps to Configure a Cisco IOS DHCPv4 Server*

Use the following steps to configure a Cisco IOS DHCPv4 server:

- **Step 1**. Exclude IPv4 addresses. A single address or a range of addresses can be excluded by specifying the *low-address* and *high-address* of the range. Excluded addresses should be those addresses that are assigned to routers, servers, printers, and other devices that have been, or will be, manually configured. You can also enter the command multiple times. The command is **ip dhcp excluded-address *low-address [high-address]***
- **Step 2**. Define a DHCPv4 pool name. The **ip dhcp pool *pool-name*** command creates a pool with the specified name and puts the router in DHCPv4 configuration mode, which is identified by the prompt **Router(dhcp-config)#.**
- **Step 3**. Configure the DHCPv4 pool. The address pool and default gateway router must be configured. Use the **network** statement to define the range of available addresses. Use the **default-router** command to define the default gateway router. These commands and other optional commands are shown in the table.

| Task | IOS Command |
|---|---|
| Define the address pool. | **network** *network-number* [*mask* \| / *prefix-length*] |
| Define the default router or gateway. | **default-router** address [ *address2….address8*] |
| Define a DNS server. | **dns-server** *address* [ *address2…address8*] |
| Define the domain name. | **domain-name** *domain* |
| Define the duration of the DHCP lease. | **lease** {*days* [*hours* [ *minutes*]] \| **infinite**} |
| Define the NetBIOS WINS server. | **netbios-name-server** *address* [ *address2…address8*] |

### *Configuration Example*




```
R1(config)# ip dhcp excluded-address 192.168.10.1 192.168.10.9
R1(config)# ip dhcp excluded-address 192.168.10.254
R1(config)# ip dhcp pool LAN-POOL-1
R1(dhcp-config)# network 192.168.10.0 255.255.255.0
R1(dhcp-config)# default-router 192.168.10.1
R1(dhcp-config)# dns-server 192.168.11.5
R1(dhcp-config)# domain-name example.com
R1(dhcp-config)# end
R1#
```

### *DHCPv4 Verification*

Use the commands in the table to verify that the Cisco IOS DHCPv4 server is operational.

| Command | Description |
|---|---|
| **show running-config \| section dhcp** | Displays the DHCPv4 commands configured on the router. |
| **show ip dhcp binding** | Displays a list of all IPv4 address to MAC address bindings provided by the DHCPv4 service. |
| **show ip dhcp server statistics** | Displays count information regarding the number of DHCPv4 messages that have been sent and received |

Verify DHCPv4 is Operational

**Verify the DHCPv4 Configuration:** As shown in the example, the **show running-config | section dhcp** command output displays the DHCPv4 commands configured on R1. The **| section** parameter displays only the commands associated with DHCPv4 configuration.

```
R1# show running-config | section dhcp
ip dhcp excluded-address 192.168.10.1 192.168.10.9
ip dhcp excluded-address 192.168.10.254
ip dhcp pool LAN-POOL-1
 network 192.168.10.0 255.255.255.0
 default-router 192.168.10.1
 dns-server 192.168.11.5
 domain-name example.com
```

**Verify DHCPv4 Bindings:** As shown in the example, the operation of DHCPv4 can be verified using the **show ip dhcp binding** command. This command displays a list of all IPv4 address to MAC address bindings that have been provided by the DHCPv4 service.

```
R1# show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address      Client-ID/              Lease expiration        Type       State      Interface
                Hardware address/
                User name
192.168.10.10   0100.5056.b3ed.d8       Sep 15 2019 8:42 AM    Automatic  Active
GigabitEthernet0/0/0
```

**Verify DHCPv4 Statistics:** The output of the **show ip dhcp server statistics** is used to verify that messages are being received or sent by the router. This command displays count information regarding the number of DHCPv4 messages that have been sent and received.

```
R1# show ip dhcp server statistics
Memory usage         19465
Address pools        1
Database agents      0
Automatic bindings   2
Manual bindings      0
Expired bindings     0
Malformed messages   0
Secure arp entries   0
Renew messages       0
Workspace timeouts   0
Static routes        0
Relay bindings       0
Relay bindings active       0
Relay bindings terminated   0
Relay bindings selecting    0
Message              Received
BOOTREQUEST          0
DHCPDISCOVER         4
DHCPREQUEST          2
DHCPDECLINE          0
DHCPRELEASE          0
DHCPINFORM           0
```

**Verify DHCPv4 Client Received IPv4 Addressing:** The **ipconfig /all** command, when issued on PC1, displays the TCP/IP parameters, as shown in the example. Because PC1 was connected to the network segment 192.168.10.0/24, it automatically received a DNS suffix, IPv4 address, subnet mask, default gateway, and DNS server address from that pool. No DHCP-specific router interface configuration is required. If a PC is connected to a network segment that has a DHCPv4 pool available, the PC can obtain an IPv4 address from the appropriate pool automatically.

```
C:\Users\Student> ipconfig /all
Windows IP Configuration
   Host Name . . . . . . . . . . . . : ciscolab
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
Ethernet adapter Ethernet0:
   Connection-specific DNS Suffix  . : example.com
   Description . . . . . . . . . . . : Realtek PCIe GBE Family Controller
   Physical Address. . . . . . . . . : 00-05-9A-3C-7A-00
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   IPv4 Address. . . . . . . . . . . : 192.168.10.10
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained  . . . . . . . . . : Saturday, September 14, 2019 8:42:22AM
   Lease Expires   . . . . . . . . . : Sunday, September 15, 2019 8:42:22AM
   Default Gateway . . . . . . . . . : 192.168.10.1
   DHCP Server . . . . . . . . . . . : 192.168.10.1
   DNS Servers . . . . . . . . . . . : 192.168.11.5
```

### *Disable the Cisco IOS DHCPv4 Server*

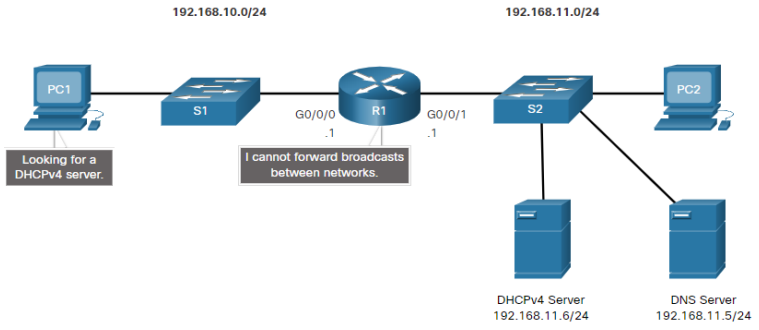
The DHCPv4 service is enabled by default. To disable the service, use the **no service dhcp** global configuration mode command. Use the **service dhcp** global configuration mode command to re-enable the DHCPv4 server process, as shown in the example. Enabling the service has no effect if the parameters are not configured.

**Note**: Clearing the DHCP bindings or stopping and restarting the DHCP service may result in duplicate IP addresses being temporarily assigned on the network.

```
R1(config)# no service dhcp
R1(config)# service dhcp
R1(config)#
```

### *DHCPv4 Relay*

- In a complex hierarchical network, enterprise servers are usually located centrally. These servers may provide DHCP, DNS, TFTP, and FTP services for the network. Network clients are not typically on the same subnet as those servers. In order to locate the servers and receive services, clients often use broadcast messages.
- In the figure, PC1 is attempting to acquire an IPv4 address from a DHCPv4 server using a broadcast message. In this scenario, R1 is not configured as a DHCPv4 server and does not forward the broadcast. Because the DHCPv4 server is located on a different network, PC1 cannot receive an IP address using DHCP. R1 must be configured to relay DHCPv4 messages to the DHCPv4 server.



- Configure R1 with the **ip helper-address** *address* interface configuration command. This will cause R1 to relay DHCPv4 broadcasts to the DHCPv4 server. As shown in the example, the interface on R1 receiving the broadcast from PC1 is configured to relay DHCPv4 address to the DHCPv4 server at 192.168.11.6.
- When R1 has been configured as a DHCPv4 relay agent, it accepts broadcast requests for the DHCPv4 service and then forwards those requests as a unicast to the IPv4 address 192.168.11.6. The network administrator can use the **show ip interface** command to verify the configuration.

```
R1(config)# interface g0/0/0
R1(config-if)# ip helper-address 192.168.11.6
R1(config-if)# end
R1#
```

```
R1# show ip interface g0/0/0
GigabitEthernet0/0/0 is up, line protocol is up
  Internet address is 192.168.10.1/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is 192.168.11.6
(output omitted)
```

### *Other Service Broadcasts Relayed*

DHCPv4 is not the only service that the router can be configured to relay. By default, the **ip helper-address** command forwards the following eight UDP services:

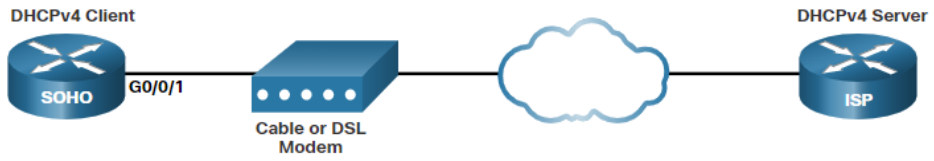
- Port 37: Time
- Port 49: TACACS
- Port 53: DNS
- Port 67: DHCP/BOOTP server
- Port 68: DHCP/BOOTP client
- Port 69: TFTP
- Port 137: NetBIOS name service
- Port 138: NetBIOS datagram service

## Configure a DHCPv4 Client

### *Cisco Router as a DHCPv4 Client*

There are scenarios where you might have access to a DHCP server through your ISP. In these instances, you can configure a Cisco IOS router as a DHCPv4 client.

- Sometimes, Cisco routers in a small office or home office (SOHO) and branch sites have to be configured as DHCPv4 clients in a similar manner to client computers. The method used depends on the ISP. However, in its simplest configuration, the Ethernet interface is used to connect to a cable or DSL modem.
- To configure an Ethernet interface as a DHCP client, use the **ip address dhcp interface** configuration mode command.
- In the figure, assume that an ISP has been configured to provide select customers with IP addresses from the 209.165.201.0/27 network range after the G0/0/1 interface is configured with the **ip address dhcp** command.



Configuration Example

- To configure an Ethernet interface as a DHCP client, use the **ip address dhcp** interface configuration mode command, as shown in the example. This configuration assumes that the ISP has been configured to provide select customers with IPv4 addressing information.
- The **show ip interface g0/1** command confirms that the interface is up and that the address was allocated by a DHCPv4 server.

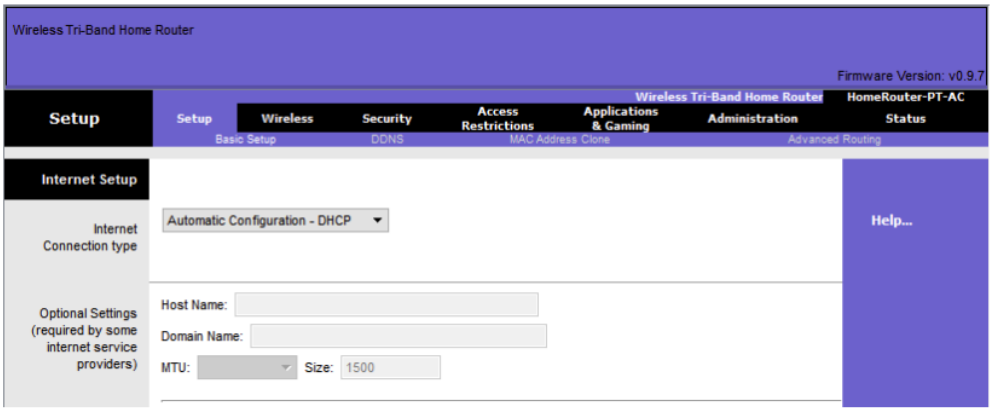
```
SOHO(config)# interface G0/0/1
SOHO(config-if)# ip address dhcp
SOHO(config-if)# no shutdown
Sep 12 10:01:25.773: %DHCP-6-ADDRESS_ASSIGN: Interface GigabitEthernet0/0/1 assigned DHCP address
209.165.201.12, mask 255.255.255.224, hostname SOHO
```

```
SOHO# show ip interface g0/0/1
GigabitEthernet0/0/1 is up, line protocol is up
  Internet address is 209.165.201.12/27
  Broadcast address is 255.255.255.255
  Address determined by DHCP
(output omitted)
```

### *Home Router as a DHCPv4 Client*

Home routers are typically already set to receive IPv4 addressing information automatically from the ISP. This is so that customers can easily set up the router and connect to the internet.

- For example, the figure shows the default WAN setup page for a Packet Tracer wireless router. Notice that the internet connection type is set to **Automatic Configuration - DHCP**. This selection is used when the router is connected to a DSL or cable modem and acts as a DHCPv4 client, requesting an IPv4 address from the ISP.
- Various manufacturers of home routers will have a similar setup.




**Laboratory Exercise: DHCPv4 Configuration**

## Module 8: SLAAC and DHCPv6

### *Objectives:*

At the end of this module, the student should be able to:

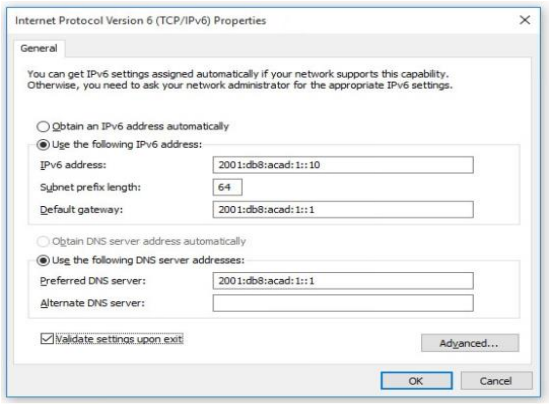
- Explain how an IPv6 host can acquire its IPv6 configuration;
- Explain the operation of SLAAC;
- Explain the operation of DHCPv6;
- Configure a stateful and stateless DHCPv6 server.

## IPv6 GUA Assignment

### *IPv6 Host Configuration*

On a router, an IPv6 global unicast address (GUA) is manually configured using the **ipv6 address** *ipv6-address/prefix-length* interface configuration command.

- A Windows host can also be manually configured with an IPv6 GUA address configuration, as shown in the figure.
- However, manually entering an IPv6 GUA can be time consuming and somewhat error prone.
- Therefore, most Windows host are enabled to dynamically acquire an IPv6 GUA configuration.




### *IPv6 Host Link-Local Address*

If automatic IPv6 addressing is selected, the host will use an Internet Control Message Protocol version 6 (ICMPv6) Router Advertisement (RA) message to help it autoconfigure an IPv6 configuration.

- The IPv6 link-local address is automatically created by the host when it boots and the Ethernet interface is active.
- The interface did not create an IPv6 GUA in the output because the network segment did not have a router to provide network configuration instructions for the host.

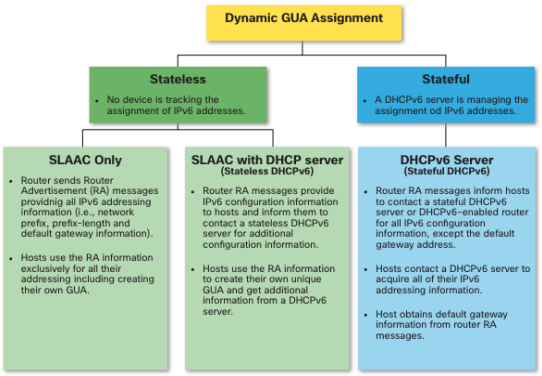
```
C:\PC1> ipconfig
Windows IP Configuration
Ethernet adapter Ethernet0:
   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . :
   Link-local IPv6 Address . . . . . : fe80::fb:1d54:839f:f595%21
   IPv4 Address. . . . . . . . . . . : 169.254.202.140
   Subnet Mask . . . . . . . . . . . : 255.255.0.0
   Default Gateway . . . . . . . . . :
C:\PC1>
```

- **Note:** The "%" and number at the end of the link-local address is known as a Zone ID or Scope ID and is used by the OS to associate the LLA with a specific interface.
- **Note**: DHCPv6 is defined in RFC 3315.

### *IPv6 GUA Assignment*

By default, an IPv6-enabled router periodically send ICMPv6 RAs which simplifies how a host can dynamically create or acquire its IPv6 configuration.

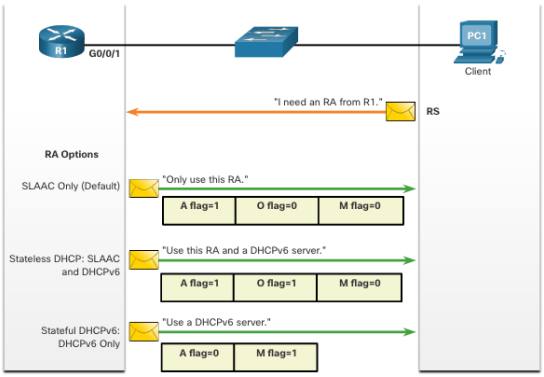
- A host can dynamically be assigned a GUA using stateless and stateful services.
- All stateless and stateful methods in this module use ICMPv6 RA messages to suggest to the host how to create or acquire its IPv6 configuration.
- Although host operating systems follow the suggestion of the RA, the actual decision is ultimately up to the host




### *Three RA Message Flags*

How a client obtains an IPv6 GUA depends on settings in the RA message.
An ICMPv6 RA message includes the following three flags:

- **A flag** - The Address Autoconfiguration flag signifies to use Stateless Address Autoconfiguration (SLAAC) to create an IPv6 GUA
- **O flag** - The Other Configuration flag signifies that additional information is available from a stateless DHCPv6 server.
- **M flag** - The Managed Address Configuration flag signifies to use a stateful DHCPv6 server to obtain an IPv6 GUA.




Using different combinations of the A, O and M flags, RA messages inform the host about the dynamic options available.

## SLAAC

### *SLAAC Overview*

Not every network has access to a DHCPv6 server but every device in an IPv6 network needs a GUA. The SLAAC method enables hosts to create their own unique IPv6 global unicast address without the services of a DHCPv6 server.

- SLAAC is a stateless service which means there is no server that maintains network address information to know which IPv6 addresses are being used and which ones are available.
- SLAAC sends periodic ICMPv6 RA messages (i.e., every 200 seconds) providing addressing and other configuration information for hosts to autoconfigure their IPv6 address based on the information in the RA.
- A host can also send a Router Solicitation (RS) message requesting an RA.
- SLAAC can be deployed as SLAAC only, or SLAAC with DHCPv6.

### *Enabling SLAAC*

R1 G0/0/1 has been configured with the indicated IPv6 GUA and link-local addresses.

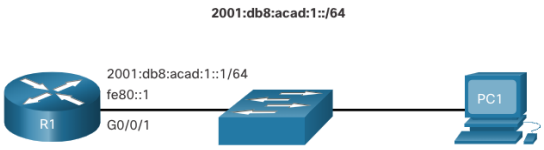
The R1 G0/0/01 IPv6 addresses include:

- **Link-local IPv6 address** - fe80::1
- **GUA / subnet** - 2001:db8:acad:1::1, 2001:db8:acad:1::/64
- **IPv6 all-nodes group** - ff02::1

R1 is configured to join the all IPv6 multicast group and start sending RA messages containing address configuration information to hosts using SLAAC.

The IPv6 all-routers group responds to the IPv6 multicast address ff02::2.

- The **show ipv6 interface** command verifies that R1 has joined the IPv6 all-routers group (i.e., ff02::2).
- R1 will now begin to send RA messages every 200 seconds to the IPv6 all-nodes multicast address ff02::1.




```
R1# show ipv6 interface G0/0/1
GigabitEthernet0/0/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::1
  No Virtual link-local address(es):
  Description: Link to LAN
  Global unicast address(es):
    2001:DB8:ACAD:1::1, subnet is 2001:DB8:ACAD:1::/64
  Joined group address(es):
    FF02::1
    FF02::1:FF00:1
(output omitted)
R1#
```

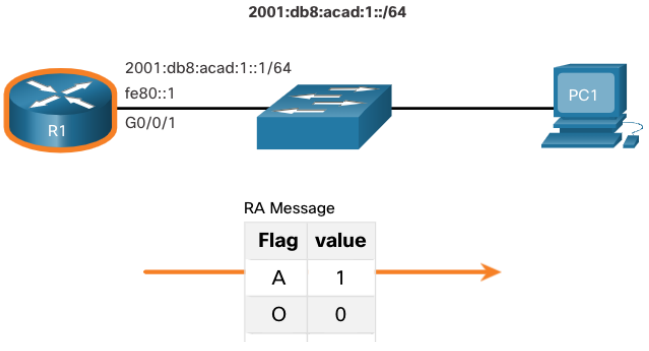
```
R1(config)# ipv6 unicast-routing
R1(config)# exit
R1#
```

```
R1# show ipv6 interface G0/0/1 | section Joined
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:1
R1#
```

### *SLAAC Only Method*

RA messages from R1 have the following flags set:

- **A = 1 –** Informs the client to use the IPv6 GUA prefix in the RA and dynamically create its own Interface ID.
- **O = 0** and **M = 0** – Informs the client to also use the additional information in the RA message (i.e., DNS server, MTU, and default gateway information).
- The **ipconfig** Windows command confirms that PC1 has generated an IPv6 GUS using the R1 RA.
- The default gateway address is LLA of the R1 G0/0/1 interface.




```
C:\PC1> ipconfig
Windows IP Configuration
Ethernet adapter Ethernet0:
   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . : 2001:db8:acad:1:1de9:c69:73ee:ca8c
   Link-local IPv6 Address . . . . . : fe80::fb:1d54:839f:f595%21
   IPv4 Address. . . . . . . . . . . : 169.254.202.140
   Subnet Mask . . . . . . . . . . . : 255.255.0.0
   Default Gateway . . . . . . . . . : fe80::1%6
C:\PC1>
```

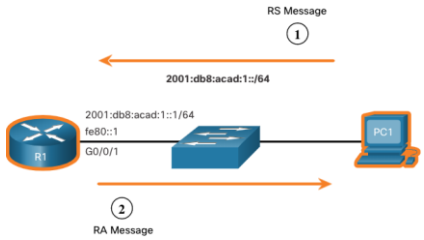
### *ICMPv6 RS Messages*

A router sends RA messages every 200 seconds or when it receives an RS message from a host.

- IPv6 enabled hosts wishing to obtain IPv6 addressing information send an RS message to the IPv6 all-routers multicast address of ff02::2.

The figure illustrates how a host initiates the SLAAC method.

1. PC1 has just booted and sends an RS message to the IPv6 all-routers multicast address of ff02::2 requesting an RA.
2. R1 generates an RA and then sends the RA message to the IPv6 all-nodes multicast address of ff02::1. PC1 uses this information to create a unique IPv6 GUA.




### *Host Process to Generate Interface ID*

Using SLAAC, a host acquires its 64-bit IPv6 subnet information from the router RA and must generate the remainder 64-bit interface identifier (ID) using either:

- **Randomly generated** - The 64-bit interface ID is randomly generated by the client operating system. This is the method now used by Windows 10 hosts.
- **EUI-64** - The host creates an interface ID using its 48-bit MAC address and inserts the hex value of fffe in the middle of the address. Some operating systems default to the randomly generated interface ID instead of the EUI-64 method, due to privacy concerns. This is because the Ethernet MAC address of the host is used by EUI-64 to create the interface ID.

**Note**: Windows, Linux, and Mac OS allow for the user to modify the generation of the interface ID to be either randomly generated or to use EUI-64.

### *Duplicate Address Detection*

A SLAAC host may use the following Duplicate Address Detection (DAD) process to ensure that the IPv6 GUA is unique.

- The host sends an ICMPv6 Neighbor Solicitation (NS) message with a specially constructed solicited-node multicast address containing the last 24 bits of IPv6 address of the host.
- If no other devices respond with a Neighbor Advertisement (NA) message, then the address is virtually guaranteed to be unique and can be used by the host.
- If an NA is received by the host, then the address is not unique, and the host must generate a new interface ID to use.

**Note**: DAD is really not required because a 64-bit interface ID provides 18 quintillion possibilities. Therefore, the chance of a duplicate address is remote. However, the Internet Engineering Task Force (IETF) recommends that DAD is used. Therefore, most operating systems perform DAD on all IPv6 unicast addresses, regardless of how the address is configured.

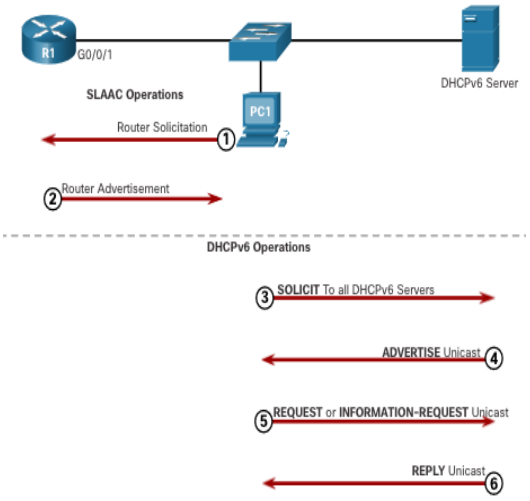
## DHCPv6

### *DHCPv6 Operation Steps*

Stateful DHCPv6 does not require SLAAC while stateless DHCPv6 does.
Regardless, when an RA indicates to use DHCPv6 or stateful DHCPv6:

1. The host sends an RS message.
2. The router responds with an RA message.
3. The host sends a DHCPv6 SOLICIT message.
4. The DHCPv6 server responds with an ADVERTISE message.
5. The host responds to the DHCPv6 server.
6. The DHCPv6 server sends a REPLY message.

**Note**: Server to client DHCPv6 messages use UDP destination port 546 while client to server DHCPv6 messages use UDP destination port 547.



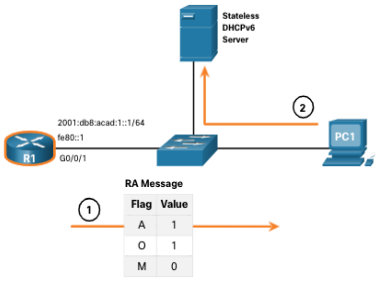

Stateless DHCPv6 Operation

If an RA indicates the stateless DHCPv6 method, the host uses the information in the RA message for addressing and contacts a DHCPv6 server for additional information.

**Note**: The DHCPv6 server only provides configuration parameters for clients and does not maintain a list of IPv6 address bindings (i.e. stateless).

For example, PC1 receives a stateless RA message containing:

- The IPv6 GUA network prefix and prefix length.
- A flag set to 1 informing the host to use SLAAC.
- O flag set to 1 informing the host to seek that additional configuration information from a DHCPv6 server.
- M flag set to the default value 0.
- PC1 sends a DHCPv6 SOLICIT message seeking additional information from a stateless DHCPv6 server.




### *Enable Stateless DHCPv6 on an Interface*

Stateless DHCPv6 is enabled using the **ipv6 nd other-config-flag** interface configuration command setting the O flag to 1.

The highlighted output confirms the RA will tell receiving hosts to use stateless autoconfigure (A flag = 1) and contact a DHCPv6 server to obtain another configuration information (O flag = 1).

**Note:** You can use the **no ipv6 nd other-config-flag** to reset the interface to the default SLAAC only option (O flag = 0).

```
R1(config-if)# ipv6 nd other-config-flag
R1(config-if)# end
R1#
R1# show ipv6 interface g0/0/1 | begin ND
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds (using 30000)
  ND advertised reachable time is 0 (unspecified)
  ND advertised retransmit interval is 0 (unspecified)
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  ND advertised default router preference is Medium
  Hosts use stateless autoconfig for addresses.
  Hosts use DHCP to obtain other configuration.
R1#
```

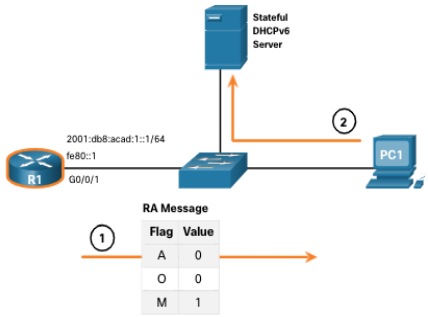
### *Stateful DHCPv6 Operation*

If an RA indicates the stateful DHCPv6 method, the host contacts a DHCPv6 server for all configuration information.

- **Note**: The DHCPv6 server is stateful and maintains a list of IPv6 address bindings.

For example, PC1 receives a stateful RA message containing:

- The IPv6 GUA network prefix and prefix length.
- A flag set to 0 informing the host to contact a DHCPv6 server.
- O flag set to 0 informing the host to contact a DHCPv6 server.
- M flag set to the value 1.
- PC1 sends a DHCPv6 SOLICIT message seeking additional information from a stateful DHCPv6 server.




### *Enable Stateful DHCPv6 on an Interface*

Stateful DHCPv6 is enabled using the **ipv6 nd managed-config-flag** interface configuration command setting the M flag to 1.

The highlighted output in the example confirms that the RA will tell the host to obtain all IPv6 configuration information from a DHCPv6 server (M flag = 1).

```
R1(config)# int g0/0/1
R1(config-if)# ipv6 nd managed-config-flag
R1(config-if)# end
R1#
R1# show ipv6 interface g0/0/1 | begin ND
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds (using 30000)
  ND advertised reachable time is 0 (unspecified)
  ND advertised retransmit interval is 0 (unspecified)
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  ND advertised default router preference is Medium
  Hosts use DHCP to obtain routable addresses.
R1#
```

## Configure DHCPv6 Server

### *DHCPv6 Router Roles*

Cisco IOS routers are powerful devices. In smaller networks, you do not have to have separate devices to have a DHCPv6 server, client, or relay agent. A Cisco IOS router can be configured to provide DHCPv6 server services.

Specifically, it can be configured to be one of the following:

- **DHCPv6 Server** - Router provides stateless or stateful DHCPv6 services.
- **DHCPv6 Client** - Router interface acquires an IPv6 IP configuration from a DHCPv6 server.
- **DHCPv6 Relay Agent** - Router provides DHCPv6 forwarding services when the client and the server are located on different networks.

### *Configure a Stateless DHCPv6 Server*

The stateless DHCPv6 server option requires that the router advertise the IPv6 network addressing information in RA messages.
There are five steps to configure and verify a router as a stateless DHCPv6 server:

1. Enable IPv6 routing using the **ipv6 unicast-routing** command.
2. Define a DHCPv6 pool name using the **ipv6 dhcp pool** *POOL-NAME* global config command.
3. Configure the DHCPv6 pool with options. Common options include **dns-server X:X:X:X:X:X:X:X** and **domain-name *name***.
4. Bind the interface to the pool using the **ipv6 dhcp server** *POOL-NAME* interface config command.
   - Manually change the O flag from 0 to 1 using the ipv6 nd other-config-flag interface command. RA messages sent on this interface indicate that additional information is available from a stateless DHCPv6 server. The A flag is 1 by default, telling clients to use SLAAC to create their own GUA.
5. Verify that the hosts have received IPv6 addressing information using the **ipconfig /all** command.

### *Configure a Stateless DHCPv6 Client*

A router can also be a DHCPv6 client and get an IPv6 configuration from a DHCPv6 server, such as a router functioning as a DHCPv6 server.

1. Enable IPv6 routing using the **ipv6 unicast-routing** command.
2. Configure the client router to create an LLA. An IPv6 link-local address is created on a router interface when a global unicast address is configured, or without a GUA using the **ipv6 enable** interface configuration command. Cisco IOS uses EUI-64 to create the Interface ID.
3. Configure the client router to use SLAAC using the **ipv6 address autoconfig** command.
4. Verify that the client router is assigned a GUA using the **show ipv6 interface brief** command.
5. Verify that the client router received other necessary DHCPv6 information. The **show ipv6 dhcp interface g0/0/1** command confirms DHCP option information, such as DNS server and domain name, have been received by the client.

### *Configure a Stateful DHCPv6 Server*

The stateful DHCP server option requires that the IPv6 enabled router tells the host to contact a DHCPv6 server to obtain all necessary IPv6 network addressing information.
There are five steps to configure and verify a router as a stateful DHCPv6 server:

1. Enable IPv6 routing using the **ipv6 unicast-routing** command.
2. Define a DHCPv6 pool name using the **ipv6 dhcp pool** *POOL-NAME* global config command.
3. Configure the DHCPv6 pool with options. Common options include the **address prefix** command, domain name, DHS server IP address, and more.
4. Bind the interface to the pool using the **ipv6 dhcp server** *POOL-NAME* interface config command.

   Manually change the M flag from 0 to 1 using the interface command **ipv6 nd managed-config-flag**.

Manually change the A flag from 1 to 0 using the **ipv6 nd prefix default no-autoconfig** interface command to inform the client to not to use SLAAC to create a GUA. The router will now respond to stateful DHCPv6 requests with the information contained in the pool.

5. Verify that the hosts have received IPv6 addressing information using the **ipconfig /all** command.

### *Configure a Stateful DHCPv6 Client*

A router can also be a DHCPv6 client. The client router needs to have **ipv6 unicast-routing** enabled and an IPv6 link-local address to send and receive IPv6 messages.
There are five steps to configure and verify a router as a stateless DHCPv6 client.

1. Enable IPv6 routing using the **ipv6 unicast-routing** command.
2. Configure the client router to create an LLA. An IPv6 link-local address is created on a router interface when a global unicast address is configured, or without a GUA using the **ipv6 enable** interface configuration command. Cisco IOS uses EUI-64 to create an Interface ID.
3. Configure the client router to use DHCPv6 using the **ipv6 address dhcp** interface config command.
4. Verify that the client router is assigned a GUA using the **show ipv6 interface brief** command.
5. Verify that the client router received other necessary DHCPv6 information using the **show ipv6 dhcp interface g0/0/1** command.

### *DHCPv6 Server Verification Commands*

The **show ipv6 dhcp pool** command verifies the name of the DHCPv6 pool and its parameters. The command also identifies the number of active clients.

```
R1# show ipv6 dhcp pool
DHCPv6 pool: IPV6-STATEFUL
  Address allocation prefix: 2001:DB8:ACAD:1::/64 valid 172800 preferred 86400 (2 in use,  0
conflicts)
  DNS server: 2001:4860:4860::8888
  Domain name: example.com
  Active clients: 2
R1#
```

Use the **show ipv6 dhcp binding** command output to display the IPv6 link-local address of the client and the global unicast address assigned by the server.

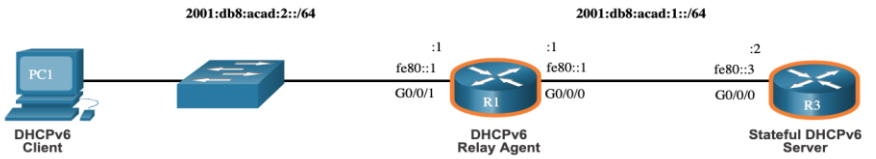
- This information is maintained by a stateful DHCPv6 server.
- A stateless DHCPv6 server would not maintain this information.

```
R1# show ipv6 dhcp binding
Client: FE80::192F:6FBC:9DB:B749
  DUID: 0001000125148183005056B327D6
  Username : unassigned
  VRF : default
  IA NA: IA ID 0x03000C29, T1 43200, T2 69120
    Address: 2001:DB8:ACAD:1:A43C:FD28:9D79:9E42
             preferred lifetime 86400, valid lifetime 172800
             expires at Sep 27 2019 09:10 AM (171192 seconds)
Client: FE80::2FC:BAFF:FE94:29B1
  DUID: 0003000100FCBA9429B0
  Username : unassigned
  VRF : default
  IA NA: IA ID 0x00060001, T1 43200, T2 69120
    Address: 2001:DB8:ACAD:1:B4CB:25FA:3C9:747C
             preferred lifetime 86400, valid lifetime 172800
             expires at Sep 27 2019 09:29 AM (172339 seconds)
R1#
```

### *Configure a DHCPv6 Relay Agent*

If the DHCPv6 server is located on a different network than the client, then the IPv6 router can be configured as a DHCPv6 relay agent.

- The configuration of a DHCPv6 relay agent is similar to the configuration of an IPv4 router as a DHCPv4 relay.
- This command is configured on the interface facing the DHCPv6 clients and specifies the DHCPv6 server address and egress interface to reach the server, as shown in the output. The egress interface is only required when the next-hop address is an LLA.




```
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# ipv6 dhcp relay destination 2001:db8:acad:1::2 G0/0/0
R1(config-if)# exit
R1(config)#
```

### *Verify the DHCPv6 Relay Agent*

Verify that the DHCPv6 relay agent is operational with the **show ipv6 dhcp interface** and **show ipv6 dhcp binding** commands.

```
R1# show ipv6 dhcp interface
GigabitEthernet0/0/1 is in relay mode
  Relay destinations:
    2001:DB8:ACAD:1::2
    2001:DB8:ACAD:1::2 via GigabitEthernet0/0/0
R1#
```

```
R3# show ipv6 dhcp binding
Client: FE80::5C43:EE7C:2959:DA68
  DUID: 0001000124F5CEA2005056B3636D
  Username : unassigned
  VRF : default
  IA NA: IA ID 0x03000C29, T1 43200, T2 69120
    Address: 2001:DB8:ACAD:2:9C3C:64DE:AADA:7857
             preferred lifetime 86400, valid lifetime 172800
             expires at Sep 29 2019 08:26 PM (172710 seconds)
R3#
```

Verify Windows hosts received IPv6 addressing information with the **ipconfig /all** command.

**Laboratory Exercise: DHCPv6 Configuration**

## Module 9: FHRP Concepts

### *Objectives:*

At the end of this module, the student should be able to:

- Explain the purpose and operation of first hop redundancy protocols;
- Explain how HSRP operates.

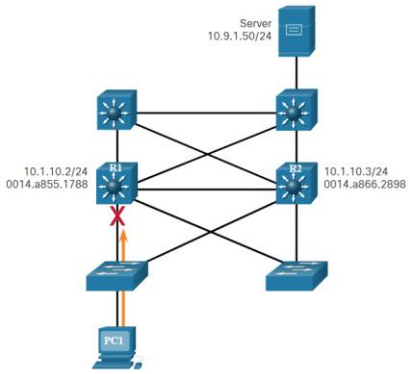
## First Hop Redundancy Protocols

### *Default Gateway Limitations*

End devices are typically configured with a single default gateway IPv4 address.

- If the default gateway router interface fails, LAN hosts lose outside LAN connectivity.
- This occurs even if a redundant router or Layer 3 switch that could serve as a default gateway exists.

First hop redundancy protocols (FHRPs) are mechanisms that provide alternate default gateways in switched networks where two or more routers are connected to the same VLANs.




### *Router Redundancy*

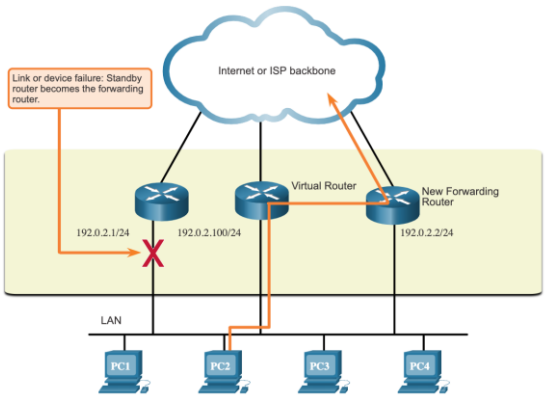
One way to prevent a single point of failure at the default gateway is to implement a virtual router. To implement this type of router redundancy, multiple routers are configured to work together to present the illusion of a single router to the hosts on the LAN. By sharing an IP address and a MAC address, two or more routers can act as a single virtual router.

- The IPv4 address of the virtual router is configured as the default gateway for the workstations on a specific IPv4 segment.
- When frames are sent from host devices to the default gateway, the hosts use ARP to resolve the MAC address that is associated with the IPv4 address of the default gateway. The ARP resolution returns the MAC address of the virtual router. Frames that are sent to the MAC address of the virtual router can then be physically processed by the currently active router within the virtual router group.
- A protocol is used to identify two or more routers as the devices that are responsible for processing frames that are sent to the MAC or IP address of a single virtual router. Host devices send traffic to the address of the virtual router. The physical router that forwards this traffic is transparent to the host devices.
- A redundancy protocol provides the mechanism for determining which router should take the active role in forwarding traffic. It also determines when the forwarding role must be taken over by a standby router. The transition from one forwarding router to another is transparent to the end devices.
- The ability of a network to dynamically recover from the failure of a device acting as a default gateway is known as first-hop redundancy.

### *Steps for Router Failover*

When the active router fails, the redundancy protocol transitions the standby router to the new active router role, as shown in the figure. These are the steps that take place when the active router fails:

1. The standby router stops seeing Hello messages from the forwarding router.
2. The standby router assumes the role of the forwarding router.
3. Because the new forwarding router assumes both the IPv4 and MAC addresses of the virtual router, the host devices see no disruption in service.




### *FHRP Options*

| FHRP Options | Description |
|---|---|
| Hot Standby Router Protocol (HSRP) | HRSP is a Cisco-proprietary FHRP that is designed to allow for transparent failover of a first-hop IPv4 device. HSRP is used in a group of routers for selecting an active device and a standby device. The active device is the device that is used for routing packets; the standby device is the device that takes over when the active device fails, or when pre-set conditions are met. |
| HSRP for IPv6 | This is a Cisco-proprietary FHRP that provides the same functionality of HSRP, but in an IPv6 environment. An HSRP IPv6 group has a virtual MAC address derived from the HSRP group number and a virtual IPv6 link-local address derived from the HSRP virtual MAC address. Periodic router advertisements (RAs) are sent for the HSRP virtual IPv6 link-local address when the HSRP group is active. When the group becomes inactive, these RAs stop after a final RA is sent. |
| Virtual Router Redundancy Protocol version 2 (VRRPv2) | This is a non-proprietary election protocol that dynamically assigns responsibility for one or more virtual routers to the VRRP routers on an IPv4 LAN. This allows several routers on a multiaccess link to use the same virtual IPv4 address. In a VRRP configuration, one router is elected as the virtual router master, with the other routers acting as backups, in case the virtual router master fails. |
| VRRPv3 | This provides the capability to support IPv4 and IPv6 addresses. VRRPv3 works in multi-vendor environments and is more scalable than VRRPv2. |
| Gateway Load Balancing Protocol (GLBP) | This is a Cisco-proprietary FHRP that protects data traffic from a failed router or circuit, like HSRP and VRRP, while also allowing load balancing (also called load sharing) between a group of redundant routers. |
| GLBP for IPv6 | This is a Cisco-proprietary FHRP that provides the same functionality of GLBP, but in an IPv6 environment. GLBP for IPv6 provides automatic router backup for IPv6 hosts configured with a single default gateway on a LAN. Multiple first-hop routers on the LAN combine to offer a single virtual first-hop IPv6 router while sharing the IPv6 packet forwarding load. |
| ICMP Router Discovery Protocol (IRDP) | Specified in RFC 1256, IRDP is a legacy FHRP solution. IRDP allows IPv4 hosts to locate routers that provide IPv4 connectivity to other (nonlocal) IP networks. |

## HSRP

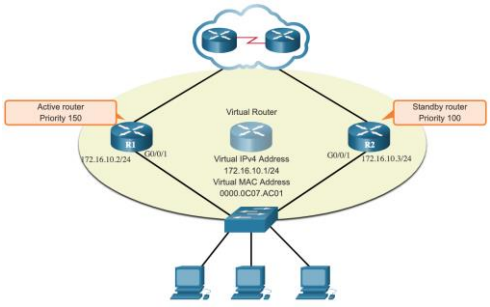
### *HSRP Overview*

Cisco provides HSRP and HSRP for IPv6 as a way to avoid losing outside network access if your default router fails. HSRP is a Cisco-proprietary FHRP that is designed to allow for transparent failover of a first-hop IP device.
HSRP ensures high network availability by providing first-hop routing redundancy for IP hosts on networks configured with an IP default gateway address. HSRP is used in a group of routers for selecting an active device and a standby device. In a group of device interfaces, the active device is the device that is used for routing packets; the standby device is the device that takes over when the active device fails, or when pre-set conditions are met. The function of the HSRP standby router is to monitor the operational status of the HSRP group and to quickly assume packet-forwarding responsibility if the active router fails.

### *HSRP Priority and Preemption*

The role of the active and standby routers is determined during the HSRP election process. By default, the router with the numerically highest IPv4 address is elected as the active router. However, it is always better to control how your network will operate under normal conditions rather than leaving it to chance.

- HSRP priority can be used to determine the active router.
- The router with the highest HSRP priority will become the active router.
- By default, the HSRP priority is 100.
- If the priorities are equal, the router with the numerically highest IPv4 address is elected as the active router.
- To configure a router to be the active router, use the **standby priority** interface command. The range of the HSRP priority is 0 to 255.




By default, after a router becomes the active router, it will remain the active router even if another router comes online with a higher HSRP priority.

- To force a new HSRP election process to take place when a higher priority router comes online, preemption must be enabled using the **standby preempt** interface command. Preemption is the ability of an HSRP router to trigger the re-election process. With preemption enabled, a router that comes online with a higher HSRP priority will assume the role of the active router.
- Preemption only allows a router to become the active router if it has a higher priority. A router enabled for preemption, with equal priority but a higher IPv4 address will not preempt an active router. Refer to the topology in the figure.

**Note**: With preemption disabled, the router that boots up first will become the active router if there are no other routers online during the election process.

### *HSRP States and Times*

| HSRP State | Description |
|---|---|
| Initial | This state is entered through a configuration change or when an interface first becomes available. |
| Learn | The router has not determined the virtual IP address and has not yet seen a hello message from the active router. In this state, the router waits to hear from the active router. |
| Listen | The router knows the virtual IP address, but the router is neither the active router nor the standby router. It listens for hello messages from those routers. |
| Speak | The router sends periodic hello messages and actively participates in the election of the active and/or standby router. |
| Standby | The router is a candidate to become the next active router and sends periodic hello messages. |

The active and standby HSRP routers send hello packets to the HSRP group multicast address every 3 seconds by default. The standby router will become active if it does not receive a hello message from the active router after 10 seconds. You can lower these timer settings to speed up the failover or preemption. However, to avoid increased CPU usage and unnecessary standby state changes, do not set the hello timer below 1 second or the hold timer below 4 seconds.

**Online Chapter Exam**

**Midterm Examination**

**Practical Examination**

## Module 10: LAN Security

### *Objectives:*

At the end of this module, the student should be able to:

- Explain how to use endpoint security to mitigate attacks;
- Explain how AAA and 802.1x are used to authenticate LAN endpoints and devices;
- Identify Layer 2 vulnerabilities;
- Explain how a MAC address table attack compromised LAN security;
- Explain how LAN attacks compromise LAN security.

## Endpoint Security

### *Network Attacks Today*

The news media commonly covers attacks on enterprise networks. Simply search the internet for “latest network attacks” to find up-to-date information on current attacks. Most likely, these attacks will involve one or more of the following:

- **Distributed Denial of Service (DDoS)** – This is a coordinated attack from many devices, called zombies, with the intention of degrading or halting public access to an organization’s website and resources.
- **Data Breach** – This is an attack in which an organization’s data servers or hosts are compromised to steal confidential information.
- **Malware** – This is an attack in which an organization’s hosts are infected with malicious software that cause a variety of problems. For example, ransomware such as WannaCry encrypts the data on a host and locks access to it until a ransom is paid.

### *Network Security Devices*

Various network security devices are required to protect the network perimeter from outside access. These devices could include the following:

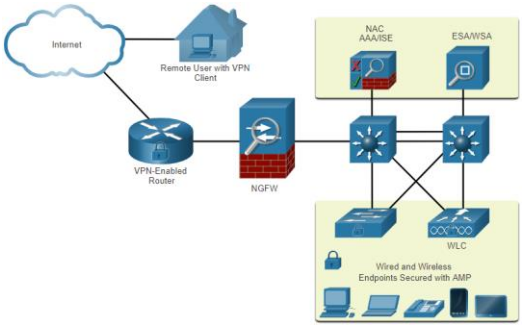
- Virtual Private Network (VPN) enabled router - provides a secure connection to remote users across a public network and into the enterprise network. VPN services can be integrated into the firewall.

- Next-Generation Firewall (NGFW) - provides stateful packet inspection, application visibility and control, a next-generation intrusion prevention system (NGIPS), advanced malware protection (AMP), and URL filtering.

- Network Access Control (NAC) - includes authentication, authorization, and accounting (AAA) services. In larger enterprises, these services might be incorporated into an appliance that can manage access policies across a wide variety of users and device types. The Cisco Identity Services Engine (ISE) is an example of a NAC device.

### *Endpoint Protection*

- Endpoints are hosts which commonly consist of laptops, desktops, servers, and IP phones, as well as employee-owned devices. Endpoints are particularly susceptible to malware-related attacks that originate through email or web browsing.
- Endpoints have typically used traditional host-based security features, such as antivirus/antimalware, host-based firewalls, and host-based intrusion prevention systems (HIPSs).
- Endpoints today are best protected by a combination of NAC, AMP software, an email security appliance (ESA), and a web security appliance (WSA).




### *Cisco Email Security Appliance*

The Cisco ESA device is designed to monitor Simple Mail Transfer Protocol (SMTP). The Cisco ESA is constantly updated by real-time feeds from the Cisco Talos, which detects and correlates threats and solutions by using a worldwide database monitoring system. This threat intelligence data is pulled by the Cisco ESA every three to five minutes.
These are some of the functions of the Cisco ESA:

- Block known threats
- Remediate against stealth malware that evaded initial detection
- Discard emails with bad links
- Block access to newly infected sites.
- Encrypt content in outgoing email to prevent data loss.

### *Cisco Web Security Appliance*

- The Cisco Web Security Appliance (WSA) is a mitigation technology for web-based threats. It helps organizations address the challenges of securing and controlling web traffic.
- The Cisco WSA combines advanced malware protection, application visibility and control, acceptable use policy controls, and reporting.
- Cisco WSA provides complete control over how users access the internet. Certain features and applications, such as chat, messaging, video and audio, can be allowed, restricted with time and bandwidth limits, or blocked, according to the organization's requirements.
- The WSA can perform blacklisting of URLs, URL-filtering, malware scanning, URL categorization, Web application filtering, and encryption and decryption of web traffic.

## Access Control

### *Authentication with a Local Password*

Many types of authentication can be performed on networking devices, and each method offers varying levels of security.

The simplest method of remote access authentication is to configure a login and password combination on console, vty lines, and aux ports

```
R1(config)# line vty 0 4
R1(config-line)# password ci5c0
R1(config-line)# login
```

SSH is a more secure form of remote access:

- It requires a username and a password.
- The username and password can be authenticated locally.

```
R1(config)# ip domain-name example.com
R1(config)# crypto key generate rsa general-keys modulus 2048
R1(config)# username Admin secret Str0ng3rPa55w0rd
R1(config)# ssh version 2
R1(config)# line vty 0 4
R1(config-line)# transport input ssh
R1(config-line)# login local
```

The local database method has some limitations:

- User accounts must be configured locally on each device which is not scalable.
- The method provides no fallback authentication method.

### *AAA Components*

AAA stands for Authentication, Authorization, and Accounting, and provides the primary framework to set up access control on a network device.

AAA is a way to control who is permitted to access a network (authenticate), what they can do while they are there (authorize), and to audit what actions they performed while accessing the network (accounting).

### *Authentication*

Local and server-based are two common methods of implementing AAA authentication.

**Local AAA Authentication:**

- Method stores usernames and passwords locally in a network device (e.g., Cisco router).
- Users authenticate against the local database.
- Local AAA is ideal for small networks.

**Server-Based AAA Authentication:**

- With the server-based method, the router accesses a central AAA server.
- The AAA server contains the usernames and password for all users.
- The router uses either the Remote Authentication Dial-In User Service (RADIUS) or Terminal Access Controller Access Control System (TACACS+) protocols to communicate with the AAA server.
- When there are multiple routers and switches, server-based AAA is more appropriate.

### *Authorization*

- AAA authorization is automatic and does not require users to perform additional steps after authentication.
- Authorization governs what users can and cannot do on the network after they are authenticated.
- Authorization uses a set of attributes that describes the user's access to the network. These attributes are used by the AAA server to determine privileges and restrictions for that user.

### *Accounting*

AAA accounting collects and reports usage data. This data can be used for such purposes as auditing or billing. The collected data might include the start and stop connection times, executed commands, number of packets, and number of bytes.
A primary use of accounting is to combine it with AAA authentication.

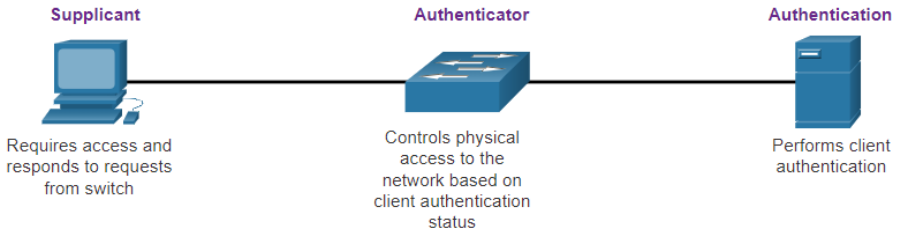
- The AAA server keeps a detailed log of exactly what the authenticated user does on the device, as shown in the figure. This includes all EXEC and configuration commands issued by the user.
- The log contains numerous data fields, including the username, the date and time, and the actual command that was entered by the user. This information is useful when troubleshooting devices. It also provides evidence for when individuals perform malicious acts.

### *802.1X*

The IEEE 802.1X standard is a port-based access control and authentication protocol. This protocol restricts unauthorized workstations from connecting to a LAN through publicly accessible switch ports. The authentication server authenticates each workstation that is connected to a switch port before making available any services offered by the switch or the LAN.

With 802.1X port-based authentication, the devices in the network have specific roles:

- **Client (Supplicant)** - This is a device running 802.1X-compliant client software, which is available for wired or wireless devices.
- **Switch (Authenticator)** –The switch acts as an intermediary between the client and the authentication server. It requests identifying information from the client, verifies that information with the authentication server, and relays a response to the client. Another device that could act as authenticator is a wireless access point.
- **Authentication server** –The server validates the identity of the client and notifies the switch or wireless access point that the client is or is not authorized to access the LAN and switch services.

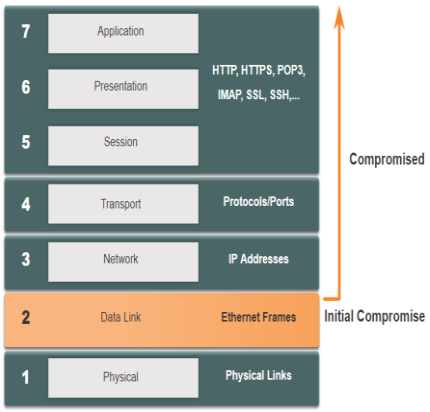


## Layer 2 Security Threats

### *Layer 2 Vulnerabilities*

Recall that the OSI reference model is divided into seven layers which work independently of each other. The figure shows the function of each layer and the core elements that can be exploited.

Network administrators routinely implement security solutions to protect the elements in Layer 3 up through Layer 7. They use VPNs, firewalls, and IPS devices to protect these elements. However, if Layer 2 is compromised, then all the layers above it are also affected. For example, if a threat actor with access to the internal network captured Layer 2 frames, then all the security implemented on the layers above would be useless. The threat actor could cause a lot of damage on the Layer 2 LAN networking infrastructure.




### *Switch Attack Categories*

Security is only as strong as the weakest link in the system, and Layer 2 is considered to be that weak link. This is because LANs were traditionally under the administrative control of a single organization. We inherently trusted all persons and devices connected to our LAN. Today, with BYOD and more sophisticated attacks, our LANs have become more vulnerable to penetration.

| Category | Examples |
|---|---|
| **MAC Table Attacks** | Includes MAC address flooding attacks. |
| **VLAN Attacks** | Includes VLAN hopping and VLAN double-tagging attacks. It also includes attacks between devices on a common VLAN. |
| **DHCP Attacks** | Includes DHCP starvation and DHCP spoofing attacks. |
| **ARP Attacks** | Includes ARP spoofing and ARP poisoning attacks. |
| **Address Spoofing Attacks** | Includes MAC address and IP address spoofing attacks. |
| **STP Attacks** | Includes Spanning Tree Protocol manipulation attacks. |

### *Switch Attack Mitigation Techniques*

| Solution | Description |
|---|---|
| **Port Security** | Prevents many types of attacks including MAC address flooding attacks and DHCP starvation attacks. |
| **DHCP Snooping** | Prevents DHCP starvation and DHCP spoofing attacks. |
| **Dynamic ARP Inspection (DAI)** | Prevents ARP spoofing and ARP poisoning attacks. |
| **IP Source Guard (IPSG)** | Prevents MAC and IP address spoofing attacks. |

These Layer 2 solutions will not be effective if the management protocols are not secured. The following strategies are recommended:

- Always use secure variants of management protocols such as SSH, Secure Copy Protocol (SCP), Secure FTP (SFTP), and Secure Socket Layer/Transport Layer Security (SSL/TLS).
- Consider using out-of-band management network to manage devices.
- Use a dedicated management VLAN where nothing but management traffic resides.
- Use ACLs to filter unwanted access.

## MAC Address Table Attack

### *Switch Operation Review*

Recall that to make forwarding decisions, a Layer 2 LAN switch builds a table based on the source MAC addresses in received frames. This is called a MAC address table. MAC address tables are stored in memory and are used to more efficiently switch frames.

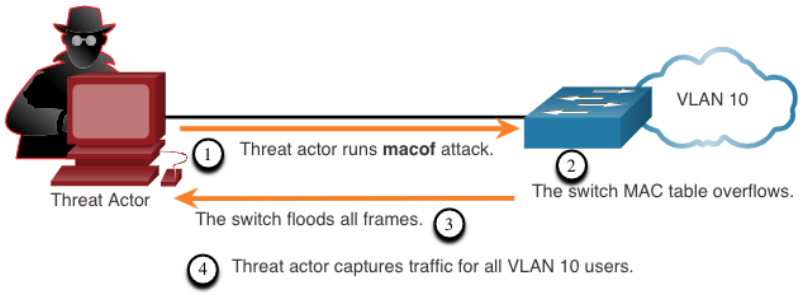
```
S1# show mac address-table dynamic
          Mac Address Table
-------------------------------------------
Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
   1    0001.9717.22e0    DYNAMIC     Fa0/4
   1    000a.f38e.74b3    DYNAMIC     Fa0/1
   1    0090.0c23.ceca    DYNAMIC     Fa0/3
   1    00d0.ba07.8499    DYNAMIC     Fa0/2
S1#
```

### *MAC Address Table Flooding*

All MAC tables have a fixed size and consequently, a switch can run out of resources in which to store MAC addresses. MAC address flooding attacks take advantage of this limitation by bombarding the switch with fake source MAC addresses until the switch MAC address table is full.

When this occurs, the switch treats the frame as an unknown unicast and begins to flood all incoming traffic out all ports on the same VLAN without referencing the MAC table. This condition now allows a threat actor to capture all of the frames sent from one host to another on the local LAN or local VLAN.

**Note**: Traffic is flooded only within the local LAN or VLAN. The threat actor can only capture traffic within the local LAN or VLAN to which the threat actor is connected.



### *MAC Address Table Attack Mitigation*

What makes tools such as **macof** so dangerous is that an attacker can create a MAC table overflow attack very quickly. For instance, a Catalyst 6500 switch can store 132,000 MAC addresses in its MAC address table. A tool such as **macof** can flood a switch with up to 8,000 bogus frames per second; creating a MAC address table overflow attack in a matter of a few seconds.

Another reason why these attack tools are dangerous is because they not only affect the local switch, they can also affect other connected Layer 2 switches. When the MAC address table of a switch is full, it starts flooding out all ports including those connected to other Layer 2 switches.

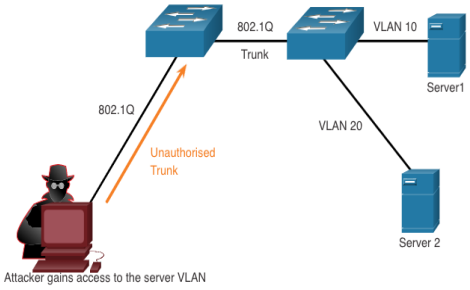
To mitigate MAC address table overflow attacks, network administrators must implement port security. Port security will only allow a specified number of source MAC addresses to be learned on the port.

## LAN Attacks

### *VLAN Hopping Attacks*

A VLAN hopping attack enables traffic from one VLAN to be seen by another VLAN without the aid of a router. In a basic VLAN hopping attack, the threat actor configures a host to act like a switch to take advantage of the automatic trunking port feature enabled by default on most switch ports.

The threat actor configures the host to spoof 802.1Q signaling and Cisco-proprietary Dynamic Trunking Protocol (DTP) signaling to trunk with the connecting switch. If successful, the switch establishes a trunk link with the host, as shown in the figure. Now the threat actor can access all the VLANs on the switch. The threat actor can send and receive traffic on any VLAN, effectively hopping between VLANs.




VLAN Double-Tagging Attacks

A threat actor is specific situations could embed a hidden 802.1Q tag inside the frame that already has an 802.1Q tag. This tag allows the frame to go to a VLAN that the original 802.1Q tag did not specify.

- **Step 1:** The threat actor sends a double-tagged 802.1Q frame to the switch. The outer header has the VLAN tag of the threat actor, which is the same as the native VLAN of the trunk port.
- **Step 2**: The frame arrives on the first switch, which looks at the first 4-byte 802.1Q tag. The switch sees that the frame is destined for the native VLAN. The switch forwards the packet out all native VLAN ports after stripping the VLAN tag. The frame is not retagged because it is part of the native VLAN. At this point, the inner VLAN tag is still intact and has not been inspected by the first switch.
- **Step 3**: The frame arrives at the second switch which has no knowledge that it was supposed to be for the native VLAN. Native VLAN traffic is not tagged by the sending switch as specified in the 802.1Q specification. The second switch looks only at the inner 802.1Q tag that the threat actor inserted and sees that the frame is destined the target VLAN. The

second switch sends the frame on to the target or floods it, depending on whether there is an existing MAC address table entry for the target.

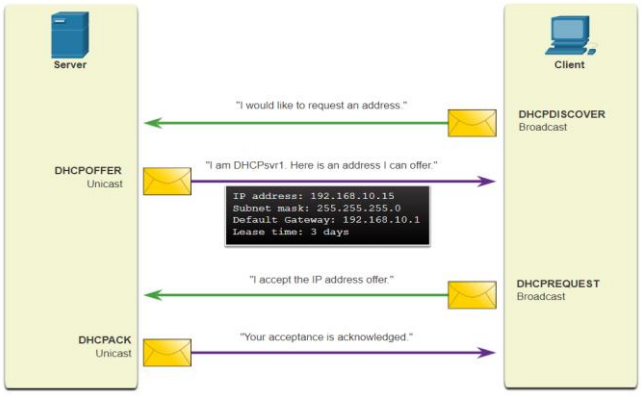
A VLAN double-tagging attack is unidirectional and works only when the attacker is connected to a port residing in the same VLAN as the native VLAN of the trunk port. The idea is that double tagging allows the attacker to send data to hosts or servers on a VLAN that otherwise would be blocked by some type of access control configuration. Presumably the return traffic will also be permitted, thus giving the attacker the ability to communicate with devices on the normally blocked VLAN.

**VLAN Attack Mitigation -** VLAN hopping and VLAN double-tagging attacks can be prevented by implementing the following trunk security guidelines, as discussed in a previous module:

- Disable trunking on all access ports.
- Disable auto trunking on trunk links so that trunks must be manually enabled.
- Be sure that the native VLAN is only used for trunk links.

### *DHCP Messages*

DHCP servers dynamically provide IP configuration information including IP address, subnet mask, default gateway, DNS servers, and more to clients. A review of the sequence of the DHCP message exchange between client and server is show in the figure.




### *DHCP Attacks*

Two types of DHCP attacks are DHCP starvation and DHCP spoofing. Both attacks are mitigated by implementing DHCP snooping.

- **DHCP Starvation Attack –** The goal of this attack is to create a DoS for connecting clients. DHCP starvation attacks require an attack tool such as Gobbler. Gobbler has the ability to look at the entire scope of leasable IP addresses and tries to lease them all. Specifically, it creates DHCP discovery messages with bogus MAC addresses.
- **DHCP Spoofing Attack –** This occurs when a rogue DHCP server is connected to the network and provides false IP configuration parameters to legitimate clients. A rogue server can provide a variety of misleading information, including the following:

  **Wrong default gateway** - The rogue server provides an invalid gateway or the IP address of its host to create a man-in-the-middle attack. This may go entirely undetected as the intruder intercepts the data flow through the network.

  **Wrong DNS server** - The rogue server provides an incorrect DNS server address pointing the user to a nefarious website.

  **Wrong IP address** - The rogue server provides an invalid IP address effectively creating a DoS attack on the DHCP client.

### *ARP Attacks*

- Hosts broadcast ARP Requests to determine the MAC address of a host with a destination IP address. All hosts on the subnet receive and process the ARP Request. The host with the matching IP address in the ARP Request sends an ARP Reply.
- A client can send an unsolicited ARP Reply called a “gratuitous ARP”. Other hosts on the subnet store the MAC address and IP address contained in the gratuitous ARP in their ARP tables.
- An attacker can send a gratuitous ARP message containing a spoofed MAC address to a switch, and the switch would update its MAC table accordingly. In a typical attack, a threat actor sends unsolicited ARP Replies to other hosts on the subnet with the MAC Address of the threat actor and the IP address of the default gateway, effectively setting up a man-in-the-middle attack.
- There are many tools available on the internet to create ARP man-in-the-middle attacks.
- IPv6 uses ICMPv6 Neighbor Discovery Protocol for Layer 2 address resolution. IPv6 includes strategies to mitigate Neighbor Advertisement spoofing, similar to the way IPv6 prevents a spoofed ARP Reply.
- ARP spoofing and ARP poisoning are mitigated by implementing Dynamic ARP Inspection (DAI).

### *Address Spoofing Attacks*

- IP address spoofing is when a threat actor hijacks a valid IP address of another device on the subnet or uses a random IP address. IP address spoofing is difficult to mitigate, especially when it is used inside a subnet in which the IP belongs.
- MAC address spoofing attacks occur when the threat actors alter the MAC address of their host to match another known MAC address of a target host. The switch overwrites the current MAC table entry and assigns the MAC address to the new port. It then inadvertently forwards frames destined for the target host to the attacking host.
- When the target host sends traffic, the switch will correct the error, realigning the MAC address to the original port. To stop the switch from returning the port assignment to its correct state, the threat actor can create a program or script that will constantly send frames to the switch so that the switch maintains the incorrect or spoofed information.
- There is no security mechanism at Layer 2 that allows a switch to verify the source of MAC addresses, which is what makes it so vulnerable to spoofing.
- IP and MAC address spoofing can be mitigated by implementing IP Source Guard (IPSG).

### *STP Attack*

- Network attackers can manipulate the Spanning Tree Protocol (STP) to conduct an attack by spoofing the root bridge and changing the topology of a network. Attackers can then capture all traffic for the immediate switched domain.
- To conduct an STP manipulation attack, the attacking host broadcasts STP bridge protocol data units (BPDUs) containing configuration and topology changes that will force spanning-tree recalculations. The BPDUs sent by the attacking host announce a lower bridge priority in an attempt to be elected as the root bridge.
- This STP attack is mitigated by implementing BPDU Guard on all access ports. BPDU Guard is discussed in more detail later in the course.

***CDP Reconnaissance***

The Cisco Discovery Protocol (CDP) is a proprietary Layer 2 link discovery protocol. It is enabled on all Cisco devices by default. Network administrators also use CDP to help configure and troubleshoot network devices. CDP information is sent out CDP-enabled ports in periodic, unencrypted, unauthenticated broadcasts. CDP information includes the IP address of the device, IOS software version, platform, capabilities, and the native VLAN. The device receiving the CDP message updates its CDP database.

To mitigate the exploitation of CDP, limit the use of CDP on devices or ports. For example, disable CDP on edge ports that connect to untrusted devices.

- To disable CDP globally on a device, use the **no cdp run** global configuration mode command. To enable CDP globally, use the **cdp run** global configuration command.
- To disable CDP on a port, use the **no cdp enable** interface configuration command. To enable CDP on a port, use the **cdp enable** interface configuration command.

**Note**: Link Layer Discovery Protocol (LLDP) is also vulnerable to reconnaissance attacks. Configure **no lldp run** to disable LLDP globally. To disable LLDP on the interface, configure **no lldp transmit** and **no lldp receive**.

**Laboratory Exercise: Access Control Configuration**

## Module 11: Switch Security Configuration

***Objectives:***

At the end of this module, the student should be able to:

- Implement port security to mitigate MAC address table attacks;
- Explain how to configure DTP and native VLAN to mitigate VLAN attacks;
- Explain how to configure DHCP snooping to mitigate DHCP attacks;
- Explain how to configure ARP inspection to mitigate ARP attacks;
- Explain how to configure PortFast and BPDU Guard to mitigate STP attacks.

### Implement Port Security

#### *Secure Unused Ports*

Layer 2 attacks are some of the easiest for hackers to deploy but these threats can also be mitigated with some common Layer 2 solutions.

- All switch ports (interfaces) should be secured before the switch is deployed for production use. How a port is secured depends on its function.
- A simple method that many administrators use to help secure the network from unauthorized access is to disable all unused ports on a switch. Navigate to each unused port and issue the Cisco IOS **shutdown** command. If a port must be reactivated at a later time, it can be enabled with the **no shutdown** command.
- To configure a range of ports, use the **interface range** command.

  Switch(config)# **interface range** *type module/first-number – last-number*

#### *Mitigate MAC Address Table Attacks*

The simplest and most effective method to prevent MAC address table overflow attacks is to enable port security.

- Port security limits the number of valid MAC addresses allowed on a port. It allows an administrator to manually configure MAC addresses for a port or to permit the switch to dynamically learn a limited number of MAC addresses. When a port configured with port security receives a frame, the source MAC address of the frame is compared to the list of secure source MAC addresses that were manually configured or dynamically learned on the port.
- By limiting the number of permitted MAC addresses on a port to one, port security can be used to control unauthorized access to the network.

### *Enable Port Security*

Port security is enabled with the **switchport port-security** interface configuration command.

Notice in the example, the **switchport port-security** command was rejected. This is because port security can only be configured on manually configured access ports or manually configured trunk ports. By default, Layer 2 switch ports are set to dynamic auto (trunking on). Therefore, in the example, the port is configured with the **switchport mode access** interface configuration command.

```
S1(config)# interface f0/1
S1(config-if)# switchport port-security
Command rejected: FastEthernet0/1 is a dynamic port.
S1(config-if)# switchport mode access
S1(config-if)# switchport port-security
S1(config-if)# end
S1#
```

**Note**: Trunk port security is beyond the scope of this course.

Use the **show port-security interface** command to display the current port security settings for FastEthernet 0/1.

- Notice how port security is enabled, the violation mode is shutdown, and how the maximum number of MAC addresses is 1.
- If a device is connected to the port, the switch will automatically add the device's MAC address as a secure MAC. In this example, no device is connected to the port.

```
S1# show port-security interface f0/1
Port Security              : Enabled
Port Status                : Secure-shutdown
Violation Mode             : Shutdown
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 1
Total MAC Addresses        : 0
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0000:0
Security Violation Count   : 0
S1#
```

**Note**: If an active port is configured with the **switchport port-security** command and more than one device is connected to that port, the port will transition to the error-disabled state.

After port security is enabled, other port security specifics can be configured, as shown in the example.

```
S1(config-if)# switchport port-security ?
  aging         Port-security aging commands
  mac-address   Secure mac address
  maximum       Max secure addresses
  violation     Security violation mode
  <cr>
S1(config-if)# switchport port-security
```

### *Limit and Learn MAC Addresses*

To set the maximum number of MAC addresses allowed on a port, use the following command:

Switch(config-if)# **switchport port-security maximum** *value*

- The default port security value is 1.
- The maximum number of secure MAC addresses that can be configured depends the switch and the IOS.
- In this example, the maximum is 8192.

```
S1(config)# interface f0/1
S1(config-if)# switchport port-security maximum ?
  <1-8192>  Maximum addresses
S1(config-if)# switchport port-security maximum
```

The switch can be configured to learn about MAC addresses on a secure port in one of three ways:

**1. Manually Configured:** The administrator manually configures a static MAC address(es) by using the following command for each secure MAC address on the port:

Switch(config-if)# **switchport port-security mac-address** *mac-address*

**2. Dynamically Learned:** When the **switchport port-security** command is entered, the current source MAC for the device connected to the port is automatically secured but is not added to the running configuration. If the switch is rebooted, the port will have to re-learn the device's MAC address.

**3. Dynamically Learned – Sticky:** The administrator can enable the switch to dynamically learn the MAC address and "stick" them to the running configuration by using the following command:

Switch(config-if)# **switchport port-security mac-address sticky**

Saving the running configuration will commit the dynamically learned MAC address to NVRAM.

The example demonstrates a complete port security configuration for FastEthernet 0/1.

- The administrator specifies a maximum of 4 MAC addresses, manually configures one secure MAC address, and then configures the port to dynamically learn additional secure MAC addresses up to the 4 secure MAC address maximum.
- Use the **show port-security interface** and the **show port-security address** command to verify the configuration.

```
S1(config)# interface fa0/1
S1(config-if)# switchport mode access
S1(config-if)# switchport port-security
S1(config-if)# switchport port-security maximum 4
S1(config-if)# switchport port-security mac-address aaaa.bbbb.1234
S1(config-if)# switchport port-security mac-address sticky
S1(config-if)# end
S1# show port-security interface fa0/1
Port Security              : Enabled
Port Status                : Secure-up
Violation Mode             : Shutdown
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 4
Total MAC Addresses        : 1
Configured MAC Addresses   : 1
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0000:0
Security Violation Count   : 0
S1# show port-security address
            Secure Mac Address Table
-----------------------------------------------------------------------------
Vlan    Mac Address       Type                          Ports   Remaining Age
                                                                    (mins)
----    -----------       ----                          -----   -------------
   1    aaaa.bbbb.1234    SecureConfigured              Fa0/1       -
-----------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 8192

S1#
```

### *Port Security Aging*

Port security aging can be used to set the aging time for static and dynamic secure addresses on a port and two types of aging are supported per port:

- **Absolute** - The secure addresses on the port are deleted after the specified aging time.
- **Inactivity** - The secure addresses on the port are deleted if they are inactive for a specified time.

Use aging to remove secure MAC addresses on a secure port without manually deleting the existing secure MAC addresses.

- Aging of statically configured secure addresses can be enabled or disabled on a per-port basis.

Use the **switchport port-security aging** command to enable or disable static aging for the secure port, or to set the aging time or type.

Switch(config-if)# **switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}}

The example shows an administrator configuring the aging type to 10 minutes of inactivity.

The **show port-security** command confirms the changes. **interface**

```
S1(config)# interface fa0/1
S1(config-if)# switchport port-security aging time 10
S1(config-if)# switchport port-security aging type inactivity
S1(config-if)# end
S1# show port-security interface fa0/1
Port Security              : Enabled
Port Status                : Secure-shutdown
Violation Mode             : Restrict
Aging Time                 : 10 mins
Aging Type                 : Inactivity
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 4
Total MAC Addresses        : 1
Configured MAC Addresses   : 1
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0050.56be.e4dd:1
Security Violation Count   : 1
```

### *Port Security Violation Modes*

If the MAC address of a device attached to a port differs from the list of secure addresses, then a port violation occurs and the port enters the error-disabled state.

- To set the port security violation mode, use the following command:

  Switch(config-if)# **switchport port-security violation** {**shutdown** | **restrict** | **protect** }

The following table shows how a switch reacts based on the configured violation mode.

| Mode | Description |
|---|---|
| **shutdown** (default) | The port transitions to the error-disabled state immediately, turns off the port LED, and sends a syslog message. It increments the violation counter. When a secure port is in the error-disabled state, an administrator must re-enable it by entering the **shutdown** and **no shutdown** commands. |
| **restrict** | The port drops packets with unknown source addresses until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the maximum value. This mode causes the Security Violation counter to increment and generates a syslog message. |
| **protect** | This is the least secure of the security violation modes. The port drops packets with unknown MAC source addresses until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the maximum value. No syslog message is sent. |

The example shows an administrator changing the security violation to “Restrict”.

The output of the **show port-security interface** command confirms that the change has been made.

```
S1(config)# interface f0/1
S1(config-if)# switchport port-security violation restrict
S1(config-if)# end
S1#
S1# show port-security interface f0/1
Port Security              : Enabled
Port Status                : Secure-shutdown
Violation Mode             : Restrict
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 4
Total MAC Addresses        : 1
Configured MAC Addresses   : 1
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0050.56be.e4dd:1
Security Violation Count   : 1
S1#
```

### *Ports in error-disabled State*

When a port is shutdown and placed in the error-disabled state, no traffic is sent or received on that port.

A series of port security related messages display on the console, as shown in the following example.

**Note**: The port protocol and link status are changed to down and the port LED is turned off.

```
*Sep 20 06:44:54.966: %PM-4-ERR_DISABLE: psecure-violation error detected on Fa0/18, putting Fa0/18 in
err-disable state
*Sep 20 06:44:54.966: %PORT_SECURITY-2-PSECURE_VIOLATION: Security violation occurred, caused by MAC
address 000c.292b.4c75 on port FastEthernet0/18.
*Sep 20 06:44:55.973: %LINEPROTO-5-PPDOWN: Line protocol on Interface FastEthernet0/18, changed state
to down
*Sep 20 06:44:56.971: %LINK-3-UPDOWN: Interface FastEthernet0/18, changed state to down
```

- In the example, the **show interface** command identifies the port status as **err-disabled**. The output of the **show port-security interface** command now shows the port status as **secure-shutdown**. The Security Violation counter increments by 1.
- The administrator should determine what caused the security violation If an unauthorized device is connected to a secure port, the security threat is eliminated before re-enabling the port.
- To re-enable the port, first use the **shutdown** command, then, use the **no shutdown** command.

```
S1# show interface fa0/18
FastEthernet0/18 is down, line protocol is down (err-disabled)
(output omitted)
S1# show port-security interface fa0/18
Port Security              : Enabled
Port Status                : Secure-shutdown
Violation Mode             : Shutdown
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 1
Total MAC Addresses        : 1
Configured MAC Addresses   : 1
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : c025.5cd7.ef01:1
Security Violation Count   : 1
S1#
```

### *Verify Port Security*

After configuring port security on a switch, check each interface to verify that the port security is set correctly, and check to ensure that the static MAC addresses have been configured correctly.

To display port security settings for the switch, use the **show port-security** command.

- The example indicates that all 24 interfaces are configured with the **switchport port-security** command because the maximum allowed is 1 and the violation mode is shutdown.
- No devices are connected, therefore, the CurrentAddr (Count) is 0 for each interface.

```
S1# show port-security
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
                (Count)       (Count)          (Count)
---------------------------------------------------------------------------
      Fa0/1            1            0                  0         Shutdown
      Fa0/2            1            0                  0         Shutdown
      Fa0/3            1            0                  0         Shutdown
(output omitted)
     Fa0/24            1            0                  0         Shutdown
---------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 4096
Switch#
```

Use the **show port-security interface** command to view details for a specific interface, as shown previously and in this example.

```
S1# show port-security interface fastethernet 0/18
Port Security              : Enabled
Port Status                : Secure-up
Violation Mode             : Shutdown
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 1
Total MAC Addresses        : 1
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0025.83e6.4b01:1
Security Violation Count   : 0
S1#
```

To verify that MAC addresses are "sticking" to the configuration, use the **show run** command as shown in the example for FastEthernet 0/19.

```
S1# show run | begin interface FastEthernet0/19
interface FastEthernet0/19
switchport mode access
switchport port-security maximum 10
switchport port-security
switchport port-security mac-address sticky
switchport port-security mac-address sticky 0025.83e6.4b02
(output omitted)
S1#
```

To display all secure MAC addresses that are manually configured or dynamically learned on all switch interfaces, use the **show port-security address** command as shown in the .

```
S1# show port-security address
Secure Mac Address Table
------------------------------------------------------------
Vlan    Mac Address       Type            Ports     Remaining Age
                                                        (mins)
----    -----------       ----            -----     -------------
1       0025.83e6.4b01    SecureDynamic   Fa0/18          -
1       0025.83e6.4b02    SecureSticky    Fa0/19          -
------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 8192
S1#
```

## Mitigate VLAN Attacks

### *VLAN Attacks Review*

A VLAN hopping attack can be launched in one of three ways:

- Spoofing DTP messages from the attacking host to cause the switch to enter trunking mode. From here, the attacker can send traffic tagged with the target VLAN, and the switch then delivers the packets to the destination.
- Introducing a rogue switch and enabling trunking. The attacker can then access all the VLANs on the victim switch from the rogue switch.
- Another type of VLAN hopping attack is a double-tagging (or double-encapsulated) attack. This attack takes advantage of the way hardware on most switches operate.

### *Steps to Mitigate VLAN Hopping Attacks*

Use the following steps to mitigate VLAN hopping attacks:

**Step 1**: Disable DTP (auto trunking) negotiations on non-trunking ports by using the **switchport mode access** interface configuration command.

**Step 2**: Disable unused ports and put them in an unused VLAN.

**Step 3**: Manually enable the trunk link on a trunking port by using the **switchport mode trunk** command.

**Step 4**: Disable DTP (auto trunking) negotiations on trunking ports by using the **switchport nonegotiate** command.

**Step 5**: Set the native VLAN to a VLAN other than VLAN 1 by using the **switchport trunk native vlan** *vlan_number* command.

```
S1(config)# interface range fa0/1 - 16
S1(config-if-range)# switchport mode access
S1(config-if-range)# exit
S1(config)#
S1(config)# interface range fa0/17 - 20
S1(config-if-range)# switchport mode access
S1(config-if-range)# switchport access vlan 1000
S1(config-if-range)# exit
S1(config)#
S1(config)# interface range fa0/21 - 24
S1(config-if-range)# switchport mode trunk
S1(config-if-range)# switchport nonegotiate
S1(config-if-range)# switchport trunk native vlan 999
S1(config-if-range)# end
S1#
```

## Mitigate DHCP Attacks

### *DHCP Attack Review*

The goal of a DHCP starvation attack is to an attack tool such as Gobbler to create a Denial of Service (DoS) for connecting clients.

Recall that DHCP starvation attacks can be effectively mitigated by using port security because Gobbler uses a unique source MAC address for each DHCP request sent. However, mitigating DHCP spoofing attacks requires more protection.

Gobbler could be configured to use the actual interface MAC address as the source Ethernet address, but specify a different Ethernet address in the DHCP payload. This would render port security ineffective because the source MAC address would be legitimate.

DHCP spoofing attacks can be mitigated by using DHCP snooping on trusted ports.

### *DHCP Snooping*

DHCP snooping filters DHCP messages and rate-limits DHCP traffic on untrusted ports.

- Devices under administrative control (e.g., switches, routers, and servers) are trusted sources.
- Trusted interfaces (e.g., trunk links, server ports) must be explicitly configured as trusted.
- Devices outside the network and all access ports are generally treated as untrusted sources.

A DHCP table is built that includes the source MAC address of a device on an untrusted port and the IP address assigned by the DHCP server to that device.

- The MAC address and IP address are bound together.
- Therefore, this table is called the DHCP snooping binding table.

### *Steps to Implement DHCP Snooping*

Use the following steps to enable DHCP snooping:

**Step 1**. Enable DHCP snooping by using the **ip dhcp snooping** global configuration command.

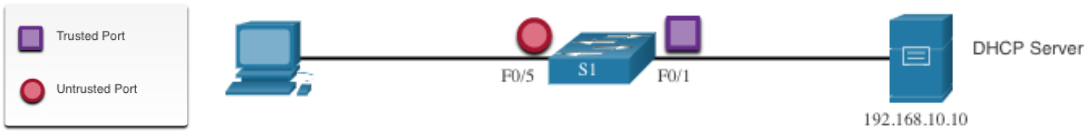
**Step 2**. On trusted ports, use the **ip dhcp snooping trust** interface configuration command.

**Step 3**: On untrusted interfaces, limit the number of DHCP discovery messages that can be received using the **ip dhcp snooping limit rate** *packets-per-second* interface configuration command.

**Step 4**. Enable DHCP snooping by VLAN, or by a range of VLANs, by using the **ip dhcp snooping** *vlan* global configuration command.

### *DHCP Snooping Configuration Example*

Refer to the DHCP snooping sample topology with trusted and untrusted ports.




- DHCP snooping is first enabled on S1.
- The upstream interface to the DHCP server is explicitly trusted.
- F0/5 to F0/24 are untrusted and are, therefore, rate limited to six packets per second.
- Finally, DHCP snooping is enabled on VLANS 5, 10, 50, 51, and 52.

```
S1(config)# ip dhcp snooping
S1(config)# interface f0/1
S1(config-if)# ip dhcp snooping trust
S1(config-if)# exit
S1(config)# interface range f0/5 - 24
S1(config-if-range)# ip dhcp snooping limit rate 6
S1(config-if)# exit
S1(config)# ip dhcp snooping vlan 5,10,50-52
S1(config)# end
S1#
```

Use the **show ip dhcp snooping** privileged EXEC command to verify DHCP snooping settings.

Use the **show ip dhcp snooping binding** command to view the clients that have received DHCP information.

**Note**: DHCP snooping is also required by Dynamic ARP Inspection (DAI).

```
S1# show ip dhcp snooping
Switch DHCP snooping is enabled
DHCP snooping is configured on following VLANs:
5,10,50-52
DHCP snooping is operational on following VLANs:
none
DHCP snooping is configured on the following L3 Interfaces:
Insertion of option 82 is enabled
   circuit-id default format: vlan-mod-port
   remote-id: 0cd9.96d2.3f80 (MAC)
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Verification of giaddr field is enabled
DHCP snooping trust/rate is configured on the following Interfaces:
Interface                  Trusted    Allow option    Rate limit (pps)
-----------------------    -------    ------------    ----------------
FastEthernet0/1            yes        yes             unlimited
  Custom circuit-ids:
FastEthernet0/5            no         no              6
  Custom circuit-ids:
FastEthernet0/6            no         no              6
  Custom circuit-ids:
S1# show ip dhcp snooping binding
MacAddress         IpAddress       Lease(sec) Type          VLAN Interface
------------------ --------------- ---------- ------------- ---- --------------------
00:03:47:B5:9F:AD  192.168.10.10   193185     dhcp-snooping 5    FastEthernet0/5
```

## Mitigate ARP Attacks

### *Dynamic ARP Inspection*

In a typical ARP attack, a threat actor can send unsolicited ARP replies to other hosts on the subnet with the MAC Address of the threat actor and the IP address of the default gateway. To prevent ARP spoofing and the resulting ARP poisoning, a switch must ensure that only valid ARP Requests and Replies are relayed.

Dynamic ARP inspection (DAI) requires DHCP snooping and helps prevent ARP attacks by:

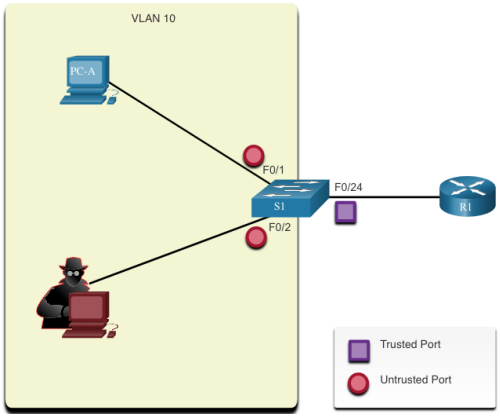
- Not relaying invalid or gratuitous ARP Replies out to other ports in the same VLAN.
- Intercepting all ARP Requests and Replies on untrusted ports.
- Verifying each intercepted packet for a valid IP-to-MAC binding.
- Dropping and logging ARP Replies coming from invalid to prevent ARP poisoning.
- Error-disabling the interface if the configured DAI number of ARP packets is exceeded.

### *DAI Implementation Guidelines*

To mitigate the chances of ARP spoofing and ARP poisoning, follow these DAI implementation guidelines:

- Enable DHCP snooping globally.
- Enable DHCP snooping on selected VLANs.
- Enable DAI on selected VLANs.
- Configure trusted interfaces for DHCP snooping and ARP inspection.

It is generally advisable to configure all access switch ports as untrusted and to configure all uplink ports that are connected to other switches as trusted.




### *DAI Configuration Example*

In the previous topology, S1 is connecting two users on VLAN 10.

- DAI will be configured to mitigate against ARP spoofing and ARP poisoning attacks.

- DHCP snooping is enabled because DAI requires the DHCP snooping binding table to operate.
- Next, DHCP snooping and ARP inspection are enabled for the PCs on VLAN10.
- The uplink port to the router is trusted, and therefore, is configured as trusted for DHCP snooping and ARP inspection.

```
S1(config)# ip dhcp snooping
S1(config)# ip dhcp snooping vlan 10
S1(config)# ip arp inspection vlan 10
S1(config)# interface fa0/24
S1(config-if)# ip dhcp snooping trust
S1(config-if)# ip arp inspection trust
```

DAI can also be configured to check for both destination or source MAC and IP addresses:

- **Destination MAC** - Checks the destination MAC address in the Ethernet header against the target MAC address in ARP body.
- **Source MAC** - Checks the source MAC address in the Ethernet header against the sender MAC address in the ARP body.
- **IP address** - Checks the ARP body for invalid and unexpected IP addresses including addresses 0.0.0.0, 255.255.255.255, and all IP multicast addresses.

The **ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]} global configuration command is used to configure DAI to drop ARP packets when the IP addresses are invalid.

- It can be used when the MAC addresses in the body of the ARP packets do not match the addresses that are specified in the Ethernet header.
- Notice in the following example how only one command can be configured.

- Therefore, entering multiple **ip arp inspection validate** commands overwrites the previous command.
- To include more than one validation method, enter them on the same command line as shown in the output.

```
S1(config)# ip arp inspection validate ?
  dst-mac  Validate destination MAC address
  ip       Validate IP addresses
  src-mac  Validate source MAC address
S1(config)# ip arp inspection validate src-mac
S1(config)# ip arp inspection validate dst-mac
S1(config)# ip arp inspection validate ip
S1(config)# do show run | include validate
ip arp inspection validate ip
S1(config)# ip arp inspection validate src-mac dst-mac ip
S1(config)# do show run | include validate
ip arp inspection validate src-mac dst-mac ip
S1(config)#
```

**Mitigate STP Attacks**

***PortFast and BPDU Guard***

Recall that network attackers can manipulate the Spanning Tree Protocol (STP) to conduct an attack by spoofing the root bridge and changing the topology of a network.

To mitigate STP attacks, use PortFast and Bridge Protocol Data Unit (BPDU) Guard:

**PortFast**

- PortFast immediately brings a port to the forwarding state from a blocking state, bypassing the listening and learning states.
- Apply to all end-user access ports.

**BPDU Guard**

- BPDU guard immediately error disables a port that receives a BPDU.
- Like PortFast, BPDU guard should only be configured on interfaces attached to end devices.

### *Configure PortFast*

PortFast bypasses the STP listening and learning states to minimize the time that access ports must wait for STP to converge.

- Only enable PortFast on access ports.
- PortFast on inter switch links can create a spanning-tree loop.

PortFast can be enabled:

- **On an interface** – Use the **spanning-tree portfast** interface configuration command.
- **Globally** – Use the **spanning-tree portfast default** global configuration command to enable PortFast on all access ports.

```
S1(config)# interface fa0/1
S1(config-if)# switchport mode access
S1(config-if)# spanning-tree portfast
%Warning: portfast should only be enabled on ports connected to a single
 host. Connecting hubs, concentrators, switches, bridges, etc... to this
 interface when portfast is enabled, can cause temporary bridging loops.
 Use with CAUTION
%Portfast has been configured on FastEthernet0/1 but will only
 have effect when the interface is in a non-trunking mode.
S1(config-if)# exit
S1(config)# spanning-tree portfast default
%Warning: this command enables portfast by default on all interfaces. You
 should now disable portfast explicitly on switched ports leading to hubs,
 switches and bridges as they may create temporary bridging loops.
S1(config)# exit
```

To verify whether PortFast is enabled globally you can use either the:

- **show running-config | begin span** command
- **show spanning-tree summary** command

To verify if PortFast is enabled an interface, use the **show running-config interface** *type/number* command.

The show **spanning-tree interface** *type/number* **detail** command can also be used for verification.

### *Configure BPDU Guard*

An access port could receive an unexpected BPDUs accidentally or because a user connected an unauthorized switch to the access port.

- If a BPDU is received on a BPDU Guard enabled access port, the port is put into error-disabled state.
- This means the port is shut down and must be manually re-enabled or automatically recovered through the **errdisable recovery cause psecure_violation** global command.

BPDU Guard can be enabled:

- **On an interface** – Use the **spanning-tree bpduguard enable** interface configuration command.
- **Globally** – Use the **spanning-tree portfast bpduguard default** global configuration command to enable BPDU

```
S1(config)# interface fa0/1
S1(config-if)# spanning-tree bpduguard enable
S1(config-if)# exit
S1(config)# spanning-tree portfast bpduguard default
S1(config)# end
S1# show spanning-tree summary
Switch is in pvst mode
Root bridge for: none
Extended system ID           is enabled
Portfast Default             is enabled
PortFast BPDU Guard Default  is enabled
Portfast BPDU Filter Default is disabled
Loopguard Default            is disabled
EtherChannel misconfig guard is enabled
UplinkFast                   is disabled
BackboneFast                 is disabled
Configured Pathcost method used is short
(output omitted)
S1#
```

**Laboratory Exercise: Switch Security Configuration**

## Module 12: WLAN Concepts

***Objectives:***

At the end of this module, the student should be able to:

- Describe WLAN technology and standards;
- Describe the components of a WLAN infrastructure;
- Explain how wireless technology enables WLAN operation;
- Explain how a WLC uses CAPWAP to manage multiple Aps;
- Describe channel management in a WLAN;
- Describe threats to WLANs;
- Describe WLAN security mechanisms.

### Introduction to Wireless

***Benefits of Wireless***

- A Wireless LAN (WLAN) is a type of wireless network that is commonly used in homes, offices, and campus environments.
- WLANs make mobility possible within the home and business environments.
- Wireless infrastructures adapt to rapidly changing needs and technologies.

***Types of Wireless Networks***

- **Wireless Personal-Area Network (WPAN)** – Low power and short-range (20-30ft or 6-9 meters). Based on IEEE 802.15 standard and 2.4 GHz frequency. Bluetooth and Zigbee are WPAN examples.
- **Wireless LAN (WLAN)** – Medium sized networks up to about 300 feet. Based on IEEE 802.11 standard and 2.4 or 5.0 GHz frequency.
- **Wireless MAN (WMAN)** – Large geographic area such as city or district. Uses specific licensed frequencies.
- **Wireless WAN (WWAN)** – Extensive geographic area for national or global communication. Uses specific licensed frequencies.

### *Wireless Technologies*

**Bluetooth** – IEEE WPAN standard used for device pairing at up to 300ft (100m) distance.

- Bluetooth Low Energy (BLE) – Supports mesh topology to large scale network devices.
- Bluetooth Basic Rate/Enhanced Rate (BR/EDR) – Supports point-to-point topologies and is optimized for audio streaming.

**WiMAX (Worldwide Interoperability for Microwave Access)** – Alternative broadband wired internet connections. IEEE 802.16 WLAN standard for up 30 miles (50 km).

**Cellular Broadband** – Carry both voice and data. Used by phones, automobiles, tablets, and laptops.

- Global System of Mobile (GSM) – Internationally recognized
- Code Division Multiple Access (CDMA) – Primarily used on the US.

**Satellite Broadband** – Uses directional satellite dish aligned with satellite in geostationary orbit. Needs clear line of site. Typically used in rural locations where cable and DSL are unavailable.

### *802.11 Standards*

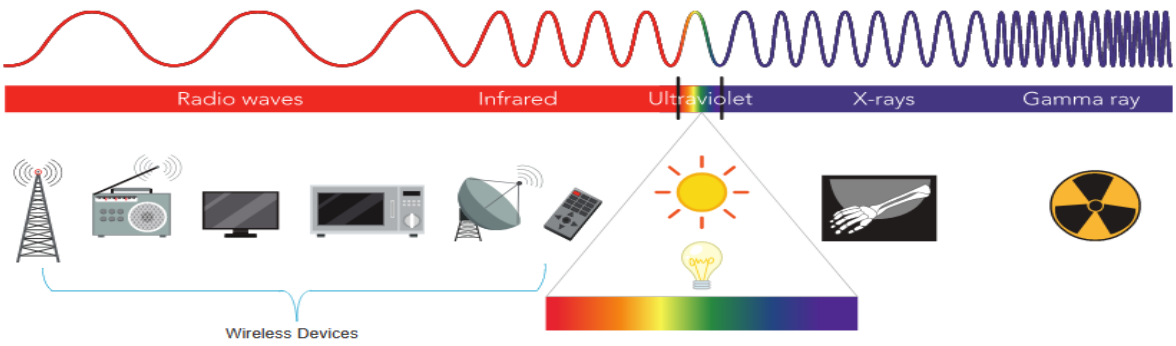
802.11 WLAN standards define how radio frequencies are used for wireless links.

| IEEE Standard | Radio Frequency | Description |
|---|---|---|
| 802.11 | 2.4 GHz | Data rates up to 2 Mb/s |
| 802.11a | 5 GHz | Data rates up to 54 Mb/s<br>Not interoperable with 802.11b or 802.11g |
| 802.11b | 2.4 GHz | Data rates up to 11 Mb/s<br>Longer range than 802.11a and better able to penetrate building structures |
| 802.11g | 2.4 GHz | Data rates up to 54 Mb/s<br>Backward compatible with 802.11b |
| 802.11n | 2.4 and 5 GHz | Data rates 150 – 600 Mb/s<br>Require multiple antennas with MIMO technology |
| 802.11ac | 5 GHz | Data rates 450 Mb/s – 1.3 Gb/s<br>Supports up to eight antennas |
| 802.11ax | 2.4 and 5 GHz | High-Efficiency Wireless (HEW)<br>Capable of using 1 GHz and 7 GHz frequencies |

### *Radio Frequencies*

All wireless devices operate in the range of the electromagnetic spectrum. WLAN networks operate in the 2.4 and 5 GHz frequency bands.

- 2.4 GHz (UHF) – 802.11b/g/n/ax
- 5 GHz (SHF) – 802.11a/n/ac/ax




### *Wireless Standards Organizations*

Standards ensure interoperability between devices that are made by different manufacturers. Internationally, the three organizations influencing WLAN standards:

- **International Telecommunication Union (ITU)** – Regulates the allocation of radio spectrum and satellite orbits.
- **Institute of Electrical and Electronics Engineers (IEEE)** – Specifies how a radio frequency is modulated to carry information. Maintains the standards for local and metropolitan area networks (MAN) with the IEEE 802 LAN/MAN family of standards.
- **Wi-Fi Alliance** – Promotes the growth and acceptance of WLANs. It is an association of vendors whose objective is to improve the interoperability of products that are based on the 802.11 standard

## WLAN Components

### *Wireless NICs*

To communicate wirelessly, laptops, tablets, smart phones, and even the latest automobiles include integrated wireless NICs that incorporate a radio transmitter/receiver.

If a device does not have an integrated wireless NIC, then a USB wireless adapter can be used.

### *Wireless Home Router*

A home user typically interconnects wireless devices using a small, wireless router.
Wireless routers serve as the following:

- **Access point** – To provide wires access
- **Switch** – To interconnect wired devices
- **Router** - To provide a default gateway to other networks and the Internet

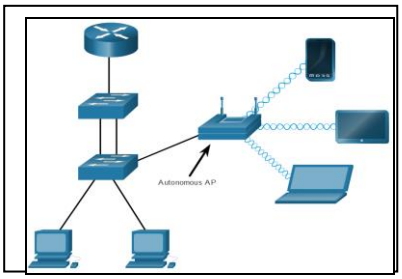
### *Wireless Access Point*

Wireless clients use their wireless NIC to discover nearby access points (APs).
Clients then attempt to associate and authenticate with an AP.
After being authenticated, wireless users have access to network resources.

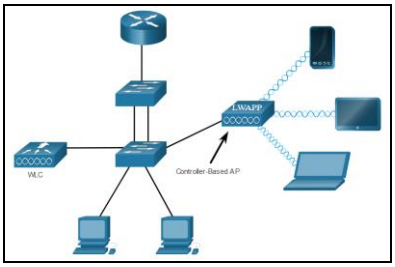
### *AP Categories*

APs can be categorized as either autonomous APs or controller-based APs.

- **Autonomous APs** – Standalone devices configured through a command line interface or GUI. Each autonomous AP acts independently of the others and is configured and managed manually by an administrator.




- **Controller-based APs** – Also known as lightweight APs (LAPs). Use Lightweight Access Point Protocol (LWAPP) to communicate with a LWAN controller (WLC). Each LAP is automatically configured and managed by the WLC.




### *Wireless Antennas*

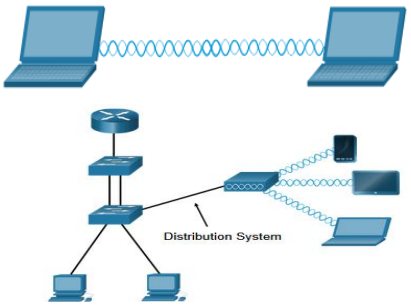
Types of external antennas:

- **Omnidirectional** – Provide 360-degree coverage. Ideal in houses and office areas.
- **Directional** – Focus the radio signal in a specific direction. Examples are the Yagi and parabolic dish.
- **Multiple Input Multiple Output (MIMO)** – Uses multiple antennas (Up to eight) to increase bandwidth.

## WLAN Operation

### *802.11 Wireless Topology Modes*

**Ad hoc mode -** Used to connect clients in peer-to-peer manner without an AP.




**Infrastructure mode -** Used to connect clients to the network using an AP.

**Tethering -** Variation of the ad hoc topology is when a smart phone or tablet with cellular data access is enabled to create a personal hotspot.


### *BSS and ESS*

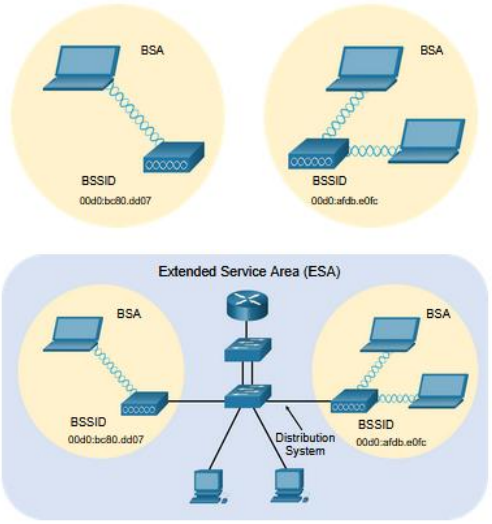
Infrastructure mode defines two topology blocks:

**Basic Service Set (BSS)**

- Uses single AP to interconnect all associated wireless clients.
- Clients in different BSSs cannot communicate.

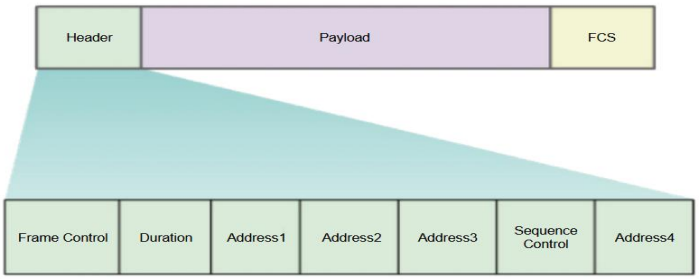
**Extended Service Set (ESS)**

- A union of two or more BSSs interconnected by a wired distribution system.
- Clients in each BSS can communication through the ESS.




### *802.11 Frame Structure*

The 802.11 frame format is similar to the Ethernet frame format, except that it contains more fields.



### *CSMA/CA*

WLANs are half-duplex and a client cannot “hear” while it is sending, making it impossible to detect a collision.

WLANs use carrier sense multiple access with collision avoidance (CSMA/CA) to determine how and when to send data. A wireless client does the following:

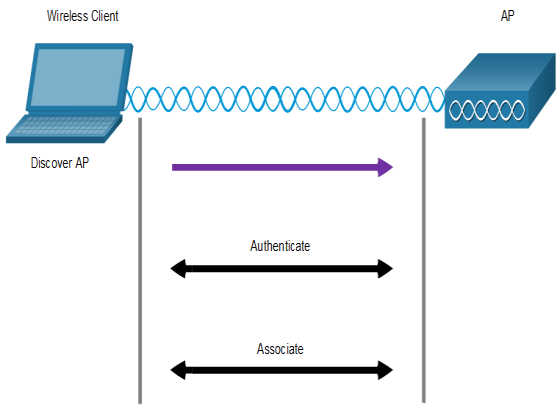
1. Listens to the channel to see if it is idle, i.e. no other traffic currently on the channel.
2. Sends a ready to send (RTS) message the AP to request dedicated access to the network.
3. Receives a clear to send (CTS) message from the AP granting access to send.
4. Waits a random amount of time before restarting the process if no CTS message received.
5. Transmits the data.
6. Acknowledges all transmissions. If a wireless client does not receive an acknowledgment, it assumes a collision occurred and restarts the process

### *Wireless Client and AP Association*

For wireless devices to communicate over a network, they must first associate with an AP or wireless router.

Wireless devices complete the following three stage process:

- Discover a wireless AP
- Authenticate with the AP
- Associate with the AP



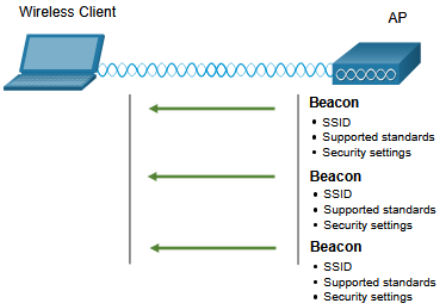

To achieve successful association, a wireless client and an AP must agree on specific parameters:

- **SSID** – The client needs to know the name of the network to connect.
- **Password** – This is required for the client to authenticate to the AP.
- **Network mode** – The 802.11 standard in use.
- **Security mode** – The security parameter settings, i.e. WEP, WPA, or WPA2.
- **Channel settings** – The frequency bands in use.

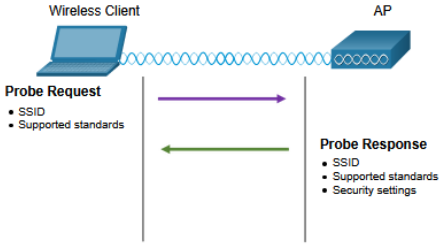
### *Passive and Active Discover Mode*

Wireless clients connect to the AP using a passive or active scanning (probing) process.

- **Passive mode** – AP openly advertises its service by periodically sending broadcast beacon frames containing the SSID, supported standards, and security settings.



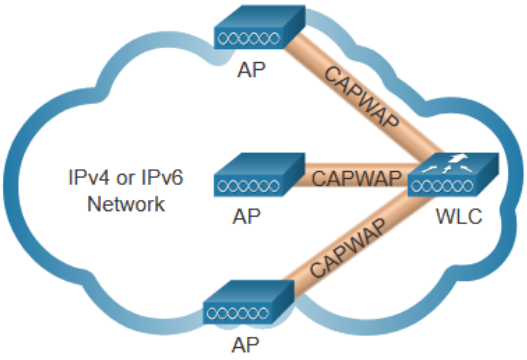

- **Active mode** – Wireless clients must know the name of the SSID. The wireless client initiates the process by broadcasting a probe request frame on multiple channels.




## CAPWAP Operation

### *Introduction to CAPWAP*

- CAPWAP is an IEEE standard protocol that enables a WLC to manage multiple APs and WLANs.
- Based on LWAPP but adds additional security with Datagram Transport Layer Security (DLTS).
- Encapsulates and forwards WLAN client traffic between an AP and a WLC over tunnels using UDP ports 5246 and 5247.
- Operates over both IPv4 and IPv6. IPv4 uses IP protocol 17 and IPv6 uses IP protocol 136.




### *Split MAC Architecture*

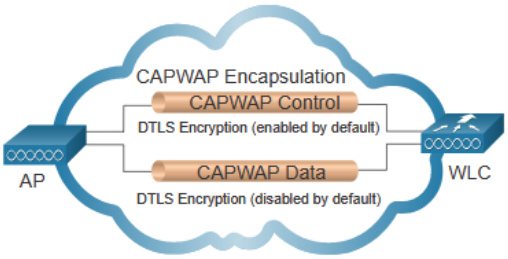
The CAPWAP split MAC concept does all the functions normally performed by individual APs and distributes them between two functional components:

- AP MAC Functions
- WLC MAC Functions

| AP MAC Functions | WLC MAC Functions |
|---|---|
| Beacons and probe responses | Authentication |
| Packet acknowledgements and retransmissions | Association and re-association of roaming clients |
| Frame queueing and packet prioritization | Frame translation to other protocols |
| MAC layer data encryption and decryption | Termination of 802.11 traffic on a wired interface |

### *DTLS Encryption*

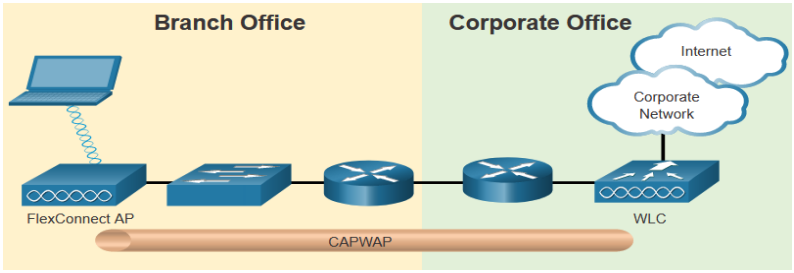
- DTLS provides security between the AP and the WLC.
- It is enabled by default to secure the CAPWAP control channel and encrypt all management and control traffic between AP and WLC.
- Data encryption is disabled by default and requires a DTLS license to be installed on the WLC before it can be enabled on the AP.




### *Flex Connect Aps*

FlexConnect enables the configuration and control of Aps over a WAN link.
There are two modes of option for the FlexConnect AP:

- **Connected mode** – The WLC is reachable. The FlexConnect AP has CAPWAP connectivity with the WLC through the CAPWAP tunnel. The WLC performs all CAPWAP functions.
- **Standalone mode** – The WLC is unreachable. The FlexConnect AP has lost CAPWAP connectivity with the WLC. The FlexConnect AP can assume some of the WLC functions such as switching client data traffic locally and performing client authentication locally.




## Channel Management

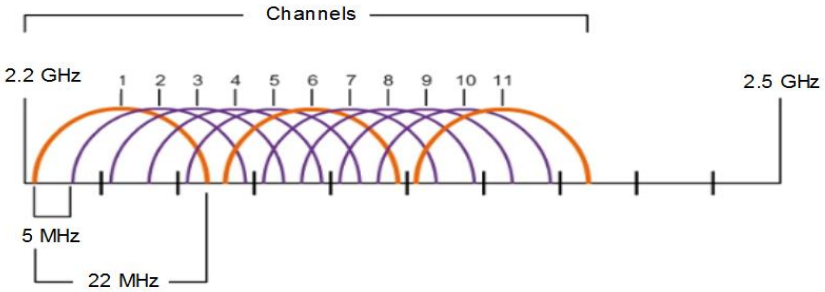
### *Frequency Channel Saturation*

If the demand for a specific wireless channel is too high, the channel may become oversaturated, degrading the quality of the communication.
Channel saturation can be mitigated using techniques that use the channels more efficiently.

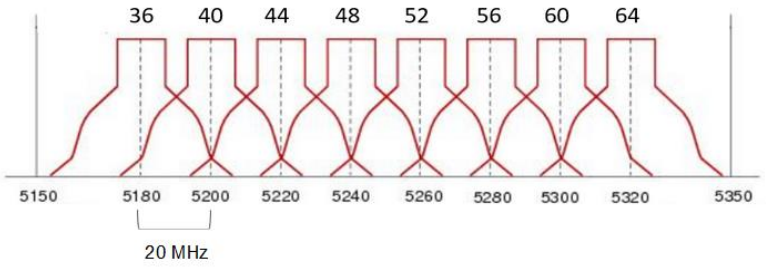
- **Direct-Sequence Spread Spectrum (DSSS)** - A modulation technique designed to spread a signal over a larger frequency band. Used by 802.11b devices to avoid interference from other devices using the same 2.4 GHz frequency.
- **Frequency-Hopping Spread Spectrum (FHSS)** - Transmits radio signals by rapidly switching a carrier signal among many frequency channels. Sender and receiver must be synchronized to “know” which channel to jump to. Used by the original 802.11 standard.
- **Orthogonal Frequency-Division Multiplexing (OFDM)** - A subset of frequency division multiplexing in which a single channel uses multiple sub-channels on adjacent frequencies. OFDM is used by a number of communication systems including 802.11a/g/n/ac.

***Channel Selection***

- The 2.4 GHz band is subdivided into multiple channels each allotted 22 MHz bandwidth and separated from the next channel by 5 MHz
- A best practice for 802.11b/g/n WLANs requiring multiple APs is to use non-overlapping channels such as 1, 6, and 11.




- For the 5GHz standards 802.11a/n/ac, there are 24 channels. Each channel is separated from the next channel by 20 MHz.
- Non-overlapping channels are 36, 48, and 60.

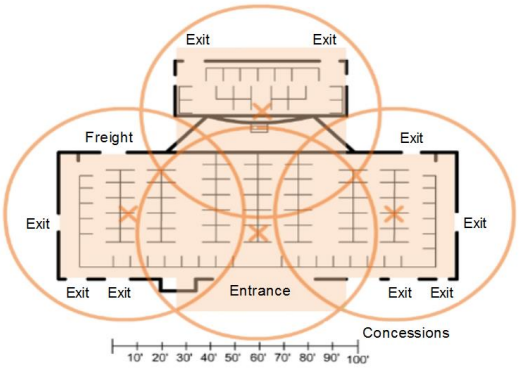



***Plan a WLAN Deployment***

The number of users supported by a WLAN depends on the following:

- The geographical layout of the facility
- The number of bodies and devices that can fit in a space
- The data rates users expect
- The use of non-overlapping channels by multiple APs and transmit power settings

When planning the location of APs, the approximate circular coverage area is important.



## WLAN Threats

### *Wireless Security Overview*

A WLAN is open to anyone within range of an AP and the appropriate credentials to associate to it.

Attacks can be generated by outsiders, disgruntled employees, and even unintentionally by employees. Wireless networks are specifically susceptible to several threats, including the following:

- Interception of data
- Wireless intruders
- Denial of Service (DoS) Attacks
- Rogue APs

### *DoS Attacks*

Wireless DoS attacks can be the result of the following:

- Improperly configured devices
- A malicious user intentionally interfering with the wireless communication
- Accidental interference

To minimize the risk of a DoS attack due to improperly configured devices and malicious attacks, harden all devices, keep passwords secure, create backups, and ensure that all configuration changes are incorporated off-hours.

### *Rogue Access Points*

- A rogue AP is an AP or wireless router that has been connected to a corporate network without explicit authorization and against corporate policy.
- Once connected, the rogue AP can be used by an attacker to capture MAC addresses, capture data packets, gain access to network resources, or launch a man-in-the-middle attack.
- A personal network hotspot could also be used as a rogue AP. For example, a user with secure network access enables their authorized Windows host to become a Wi-Fi AP.
- To prevent the installation of rogue APs, organizations must configure WLCs with rogue AP policies and use monitoring software to actively monitor the radio spectrum for unauthorized APs.

### *Man-in-the-Middle Attack*

In a man-in-the-middle (MITM) attack, the hacker is positioned in between two legitimate entities in order to read or modify the data that passes between the two parties. A popular wireless MITM attack is called the “evil twin AP” attack, where an attacker introduces a rogue AP and configures it with the same SSID as a legitimate AP.

Defeating a MITM attack begins with identifying legitimate devices on the WLAN. To do this, users must be authenticated. After all of the legitimate devices are known, the network can be monitored for abnormal devices or traffic.

## Secure WLANs

### *SSID Cloaking and MAC Address Filtering*

To address the threats of keeping wireless intruders out and protecting data, two early security features were used and are still available on most routers and APs:

**SSID Cloaking**

- APs and some wireless routers allow the SSID beacon frame to be disabled. Wireless clients must be manually configured with the SSID to connect to the network.

**MAC Address Filtering**

- An administrator can manually permit or deny clients wireless access based on their physical MAC hardware address. In the figure, the router is configured to permit two MAC addresses. Devices with different MAC addresses will not be able to join the 2.4GHz WLAN.

### *802.11 Original Authentication Methods*

The best way to secure a wireless network is to use authentication and encryption systems. Two types of authentication were introduced with the original 802.11 standard:

**Open system authentication**

- No password required. Typically used to provide free internet access in public areas like cafes, airports, and hotels.
- Client is responsible for providing security such as through a VPN.

**Shared key authentication**

- Provides mechanisms, such as WEP, WPA, WPA2, and WPA3 to authenticate and encrypt data between a wireless client and AP. However, the password must be pre-shared between both parties to connect.

### *Shared Key Authentication Methods*

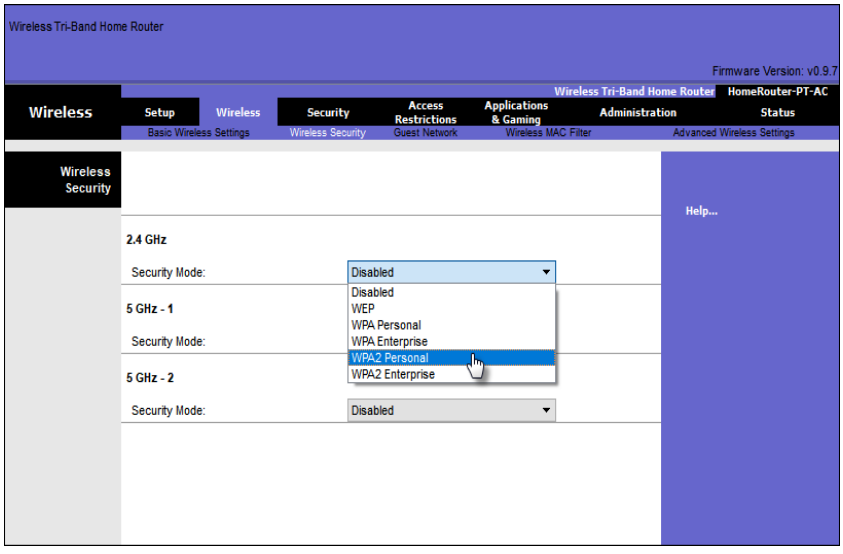
There are currently four shared key authentication techniques available, as shown in the table.

| Authentication Method | Description |
|---|---|
| Wired Equivalent Privacy (WEP) | The original 802.11 specification designed to secure the data using the Rivest Cipher 4 (RC4) encryption method with a static key. WEP is no longer recommended and should never be used. |
| Wi-Fi Protected Access (WPA) | A Wi-Fi Alliance standard that uses WEP but secures the data with the much stronger Temporal Key Integrity Protocol (TKIP) encryption algorithm. TKIP changes the key for each packet, making it much more difficult to hack. |
| WPA2 | It uses the Advanced Encryption Standard (AES) for encryption. AES is currently considered the strongest encryption protocol. |
| WPA3 | This is the next generation of Wi-Fi security. All WPA3-enabled devices use the latest security methods, disallow outdated legacy protocols, and require the use of Protected Management Frames (PMF). |

### *Authenticating a Home User*

Home routers typically have two choices for authentication: WPA and WPA2, with WPA 2 having two authentication methods.

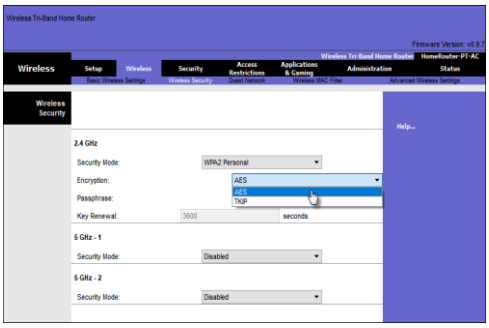
- **Personal** – Intended for home or small office networks, users authenticate using a pre-shared key (PSK). Wireless clients authenticate with the wireless router using a pre-shared password. No special authentication server is required.
- **Enterprise** – Intended for enterprise networks. Requires a Remote Authentication Dial-In User Service (RADIUS) authentication server. The device must be authenticated by the RADIUS server and then users must authenticate using 802.1X standard, which uses the Extensible Authentication Protocol (EAP) for authentication.



### *Encryption Methods*

WPA and WPA2 include two encryption protocols:

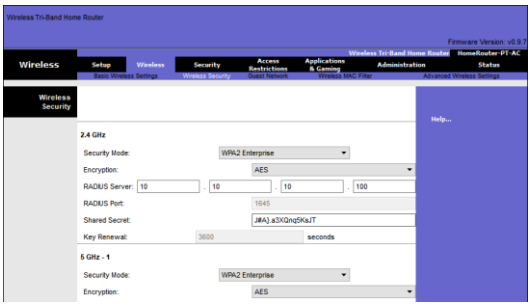
- **Temporal Key Integrity Protocol (TKIP)** – Used by WPA and provides support for legacy WLAN equipment. Makes use of WEP but encrypts the Layer 2 payload using TKIP.
- **Advanced Encryption Standard (AES)** – Used by WPA2 and uses the Counter Cipher Mode with Block Chaining Message Authentication Code Protocol (CCMP) that allows destination hosts to recognize if the encrypted and non-encrypted bits have been altered.




### *Authentication in the Enterprise*

Enterprise security mode choice requires an Authentication, Authorization, and Accounting (AAA) RADIUS server.
There pieces of information are required:

- **RADIUS server IP address** – IP address of the server.
- **UDP port numbers** –UDP ports 1812 for RADIUS Authentication, and 1813 for RADIUS Accounting, but can also operate using UDP ports 1645 and 1646.
- **Shared key** – Used to authenticate the AP with the RADIUS server.




**Note**: User authentication and authorization is handled by the 802.1X standard, which provides a centralized, server-based authentication of end users.

### *WPA 3*

Because WPA2 is no longer considered secure, WPA3 is recommended when available. WPA3 Includes four features:

- **WPA3 – Personal :** Thwarts brute force attacks by using Simultaneous Authentication of Equals (SAE).
- **WPA3 – Enterprise :** Uses 802.1X/EAP authentication. However, it requires the use of a 192-bit cryptographic suite and eliminates the mixing of security protocols for previous 802.11 standards.
- **Open Networks :** Does not use any authentication. However, uses Opportunistic Wireless Encryption (OWE) to encrypt all wireless traffic.
- **IoT Onboarding :** Uses Device Provisioning Protocol (DPP) to quickly onboard IoT devices.

## Module 13: WLAN Configuration

***Objectives:***

At the end of this module, the student should be able to:

- Configure a WLAN to support a remote site;
- Configure a WLC WLAN to use the management interface and WPA2 PSK authentication;
- Configure a WLC WLAN to use a VLAN interface, a DHCP server, and WPA2 Enterprise authentication;
- Troubleshoot common wireless configuration issues.

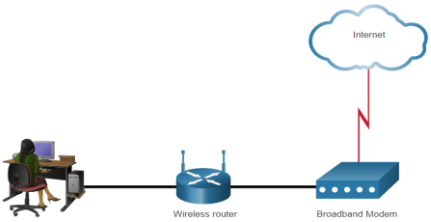
### Remote Site WLAN Configuration

***The Wireless Router***

Remote workers, small branch offices, and home networks often use a small office and home router.

- These "integrated" routers typically include a switch for wired clients, a port for an internet connection (sometimes labeled "WAN"), and wireless components for wireless client access.
- These wireless routers typically provide WLAN security, DHCP services, integrated Name Address Translation (NAT), quality of service (QoS), as well as a variety of other features.
- The feature set will vary based on the router model.

**Note**: Cable or DSL modem configuration is usually done by the service provider's representative either on-site or remotely.




***Log in to the Wireless Router***

Most wireless routers are preconfigured to be connected to the network and provide services.

- Wireless router default IP addresses, usernames, and passwords can easily be found on the internet.
- Therefore, your first priority should be to change these defaults for security reasons.

To gain access to the wireless router's configuration GUI

- Open a web browser and enter the default IP address for your wireless router.
- The default IP address can be found in the documentation that came with the wireless router or you can search the internet.
- The word **admin** is commonly used as the default username and password.

### *Basic Network Setup*

Basic network setup includes the following steps:

- Log in to the router from a web browser.
- Change the default administrative password.
- Log in with the new administrative password.
- Change the default DHCP IPv4 addresses.
- Renew the IP address.
- Log in to the router with the new IP address.

### *Basic Wireless Setup*

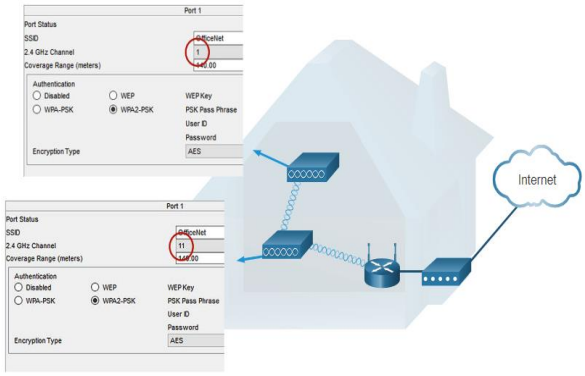
Basic wireless setup includes the following steps:

- View the WLAN defaults.
- Change the network mode, identifying which 802.11 standard is to be implemented.
- Configure the SSID.
- Configure the channel, ensuring there are no overlapping channels in use.
- Configure the security mode, selecting from Open, WPA, WPA2 Personal, WPA2 Enterprise, etc..
- Configure the passphrase, as required for the selected security mode.

### *Configure a Wireless Mesh Network*

In a small office or home network, one wireless router may suffice to provide wireless access to all the clients.

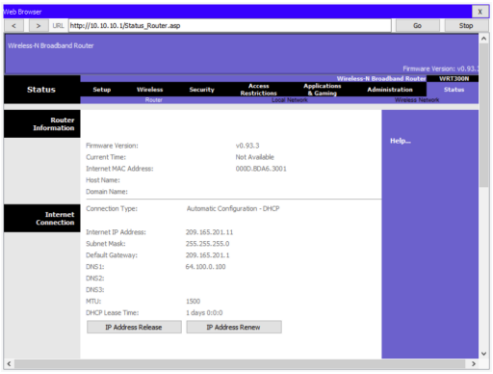
- If you want to extend the range beyond approximately 45 meters indoors and 90 meters outdoors, you create a wireless mesh.
- Create the mesh by adding access points with the same settings, except using different channels to prevent interference.
- Extending a WLAN in a small office or home has become increasingly easier.
- Manufacturers have made creating a wireless mesh network (WMN) simple through smartphone apps.



### *NAT for IPv4*

Typically, the wireless router is assigned a publicly routable address by the ISP and uses a private network address for addressing on the LAN.

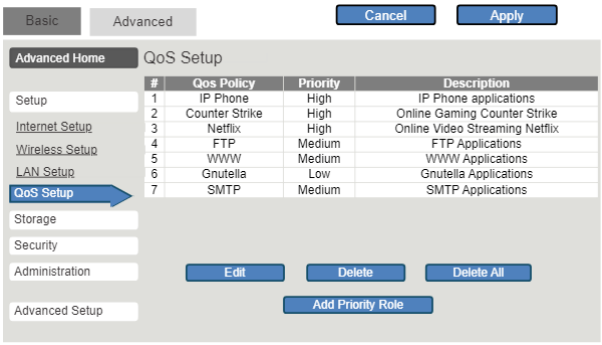
- To allow hosts on the LAN to communicate with the outside world, the router will use a process called Network Address Translation (NAT).
- NAT translates a private (local) source IPv4 address to a public (global) address (the process is reversed for incoming packets).
- NAT makes sharing one public IPv4 address possible by tracking the source port numbers for every session established by a device.
- If your ISP has IPv6 enabled, you will see a unique IPv6 address for each device.




### *Quality of Service*

Many wireless routers have an option for configuring Quality of Service (QoS).

- By configuring QoS, you can guarantee that certain traffic types, such as voice and video, are prioritized over traffic that is not as time-sensitive, such as email and web browsing.
- On some wireless routers, traffic can also be prioritized on specific ports.



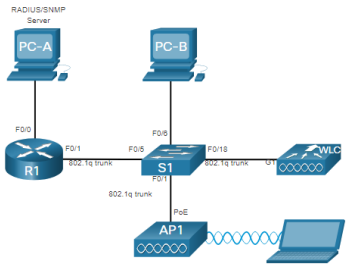

### *Port Forwarding*

Wireless routers typically block TCP and UDP ports to prevent unauthorized access in and out of a LAN.

- However, there are situations when specific ports must be opened so that certain programs and applications can communicate with devices on different networks.
- Port forwarding is a rule-based method of directing traffic between devices on separate networks.
- Port triggering allows the router to temporarily forward data through inbound ports to a specific device.
- You can use port triggering to forward data to a computer only when a designated port range is used to make an outbound request.

## Configure a Basic WLAN on the WLC

### *WLC Topology*




The topology and addressing scheme used for this topic are shown in the figure and the table.

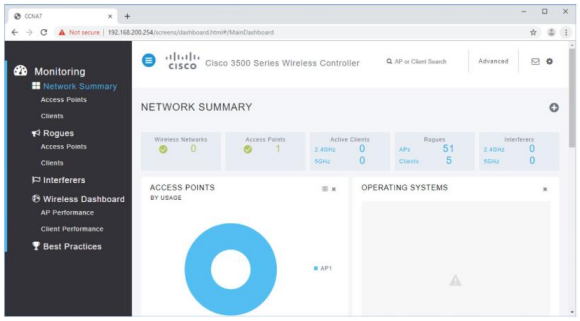
- The access point (AP) is a controller-based AP as opposed to an autonomous AP, so it requires no initial configuration and is often called lightweight APs (LAPs).
- LAPs use the Lightweight Access Point Protocol (LWAPP) to communicate with a WLAN controller (WLC).
- Controller-based APs are useful in situations where many APs are required in the network.
- As more APs are added, each AP is automatically configured and managed by the WLC.

| Device | Interface | IP Address | Subnet Mask |
|---|---|---|---|
| R1 | F0/0 | 172.16.1.1 | 255.255.255.0 |
| R1 | F0/1.1 | 192.168.200.1 | 255.255.255.0 |
| S1 | VLAN 1 | DHCP | |
| WLC | Management | 192.168.200.254 | 255.255.255.0 |
| AP1 | Wired 0 | 192.168.200.3 | 255.255.255.0 |
| PC-A | NIC | 172.16.1.254 | 255.255.255.0 |
| PC-B | NIC | DHCP | |
| Wireless Laptop | NIC | DHCP | |

### *Lon in to the WLC*

Configuring a wireless LAN controller (WLC) is not that much different from configuring a wireless router. The WLC controls APs and provides more services and management capabilities.

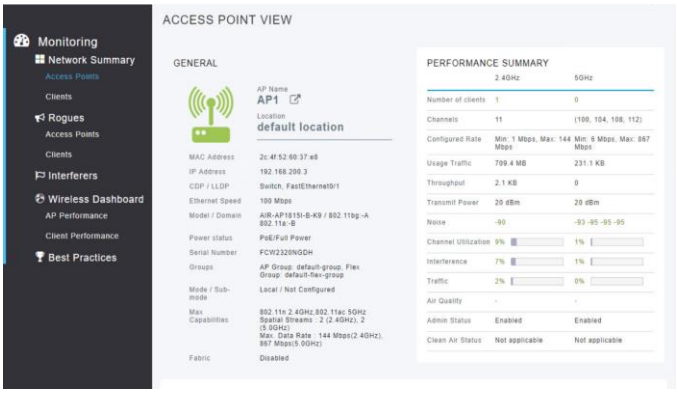
- The user logs into the WLC using credentials that were configured during initial setup.
- The **Network Summary** page is a dashboard that provides a quick overview of configured wireless networks, associated access points (APs), and active clients.
- You can also see the number of rogue access points and clients.



### *View AP Information*

Click **Access Points** from the left menu to view an overall picture of the AP's system information and performance.

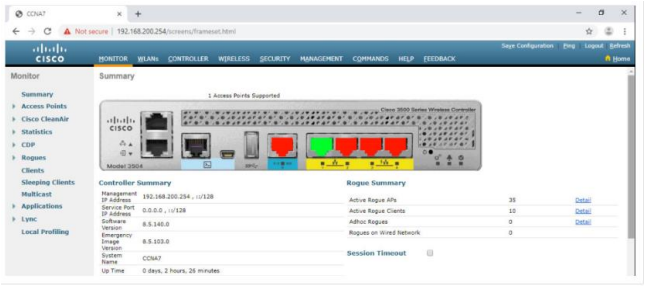
- The AP is using IP address 192.168.200.3.
- Because Cisco Discovery Protocol (CDP) is active on this network, the WLC knows that the AP is connected to the FastEthernet 0/1 port on the switch.
- This AP in the topology is a Cisco Aironet 1815i which means you can use the command-line and a limited set of familiar IOS commands.




### *Advanced Settings*

Most WLC will come with some basic settings and menus that users can quickly access to implement a variety of common configurations.

- However, as a network administrator, you will typically access the advanced settings.
- For the Cisco 3504 Wireless Controller, click **Advanced** in the upper right-hand corner to access the advanced **Summary** page.
- From here, you can access all the features of the WLC.



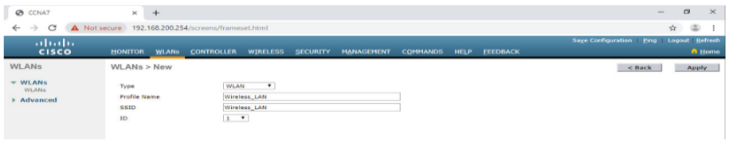

### *Configure a WLAN*

Wireless LAN Controllers have Layer 2 switch ports and virtual interfaces that are created in software and are very similar to VLAN interfaces.

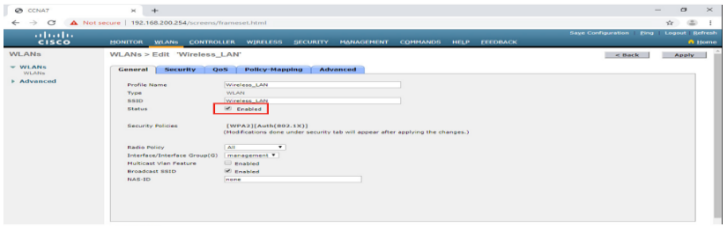
- Each physical port can support many APs and WLANs.
- The ports on the WLC are essentially trunk ports that can carry traffic from multiple VLANs to a switch for distribution to multiple APs.
- Each AP can support multiple WLANs.

Basic WLAN configuration on the WLC includes the following steps:

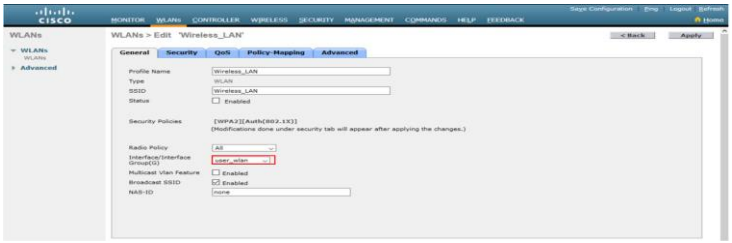
1. **Create the WLAN:** In the figure, a new WLAN with an SSID name **Wireless_LAN** is created.



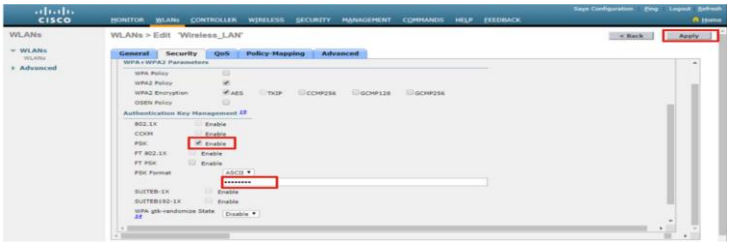
2. **Apply and Enable the WLAN:** Next the WLAN is enabled the WLAN settings are configured.



6. **Select the Interface:** The interface that will carry the WLAN traffic must be selected.

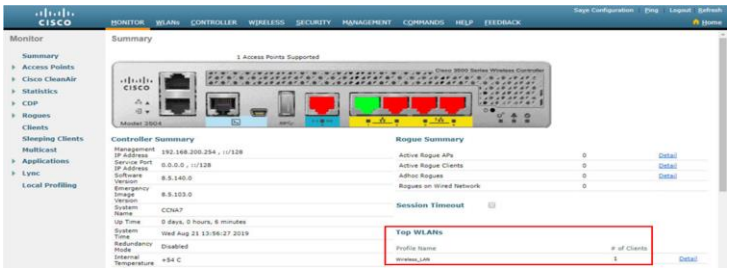


5. **Secure the WLAN:** The Security tab is used to access all the available options for securing the LAN.



4. **Verify the WLAN is Operational:** The WLANs menu on the left is used to view the newly configured WLAN and its settings.

3. **Monitor the WLAN:** The Monitor tab is used to access the advanced Summary page and confirm that the Wireless_LAN now has one client using its services.



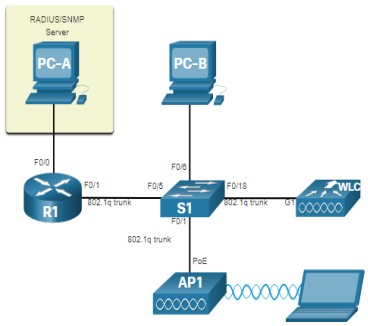
7. **View Wireless Client Details:** Click Clients in the left menu to view more information about the clients connected to the WLAN.

## Configure a WPA2 Enterprise WLAN on the WLC

### *SNMP and RADIUS*

PC-A is running Simple Network Management Protocol (SNMP) and Remote Authentication Dial-In User Service (RADIUS) server software.

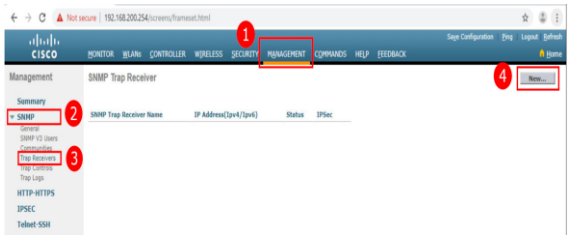
- The network administrator wants the WLC to forward all SNMP log messages (i.e., traps) to the SNMP server.
- The network administrator wants to use a RADIUS server for authentication, authorization, and accounting (AAA) services.
- Users will enter their username and password credentials which will be verified by the RADIUS server.
- The RADIUS server is required for WLANs that are using WPA2 Enterprise authentication.




### *Configure SNMP Server Information*

To enable SNMP and configure settings:

1. Click the **MANAGEMENT** tab to access a variety of management features.
2. Click **SNMP** to expand the sub-menus.
3. Click **Trap Receivers**.
4. Click **New...** to configure a new SNMP trap receiver.



- Enter the SNMP Community name and the IP address (IPv4 or IPv6) for the SNMP server and then click **Apply**.
- The WLC will now forward SNMP log messages to the SNMP server.

### *Configure RADIUS Server Information*

To configure the WLC with the RADIUS server information:

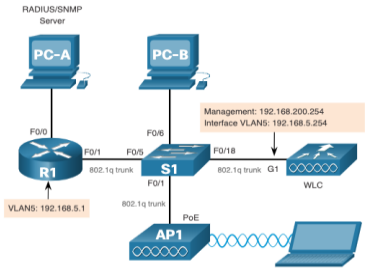
1. Click **SECURITY.**
2. Click **RADIUS**
3. Click **Authentication**
4. Click **New...** to add PC-A as the RADIUS server.
   Enter the IPv4 address for PC-A and the shared secret that will be used between the WLC and the RADIUS server and then click Apply.

After clicking **Apply**, the list of configured **RADIUS Authentication Servers** refreshes with the new server listed.

### *Topology with VLAN 5 Addressing*

Each WLAN configured on the WLC needs its own virtual interface.

- The WLC has five physical data ports that can be configured to support multiple WLANs and virtual interface.
- The new WLAN will use interface VLAN 5 and network 192.168.5.0/24 and therefore R1 has been configured for VLAN 5 as shown in the topology and **show ip interface brief** output.




```
R1# show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
FastEthernet0/0            172.16.1.1      YES manual up                    up
FastEthernet0/1            unassigned      YES unset  up                    up
FastEthernet0/1.1          192.168.200.1   YES manual up                    up
FastEthernet0/1.5          192.168.5.254   YES manual up                    up
(output omitted)
R1#
```

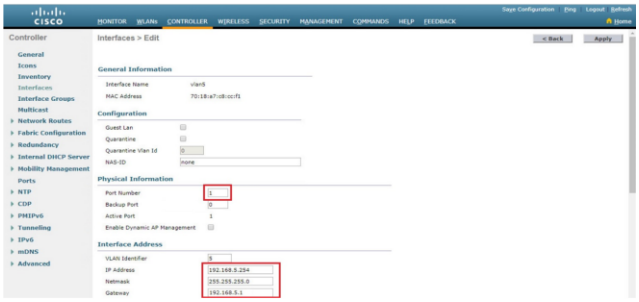
### *Configure a New Interface*

VLAN interface configuration on the WLC includes the following steps:

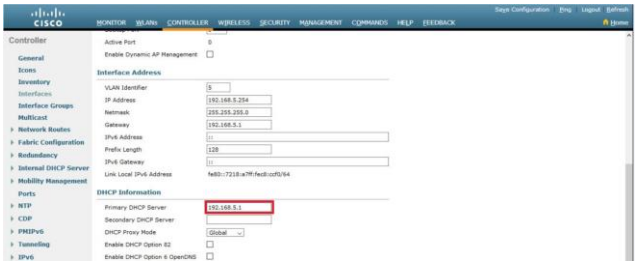
1. **Create a new interface:** Click **CONTROLLER > Interfaces > New...**


2. **Configure the VLAN name and ID:** In the example, the new interface is named **vlan5**, the VLAN ID is **5**, and applied.


3. **Configure the port and interface address:** On the interface Edit page, configure the physical port number (i.e., the WLC G1 interface is Port Number 1 on the WLC), the VLAN 5 interface addressing (i.e., 192.168.5.254/24), and the default gateway (i.e., 192.168.5.1)




4. **Configure the DHCP server address:** The example configures a primary DHCP server at IPv4 address 192.168.5.1 which is the default gateway router address which is enabled as a DHCP server.



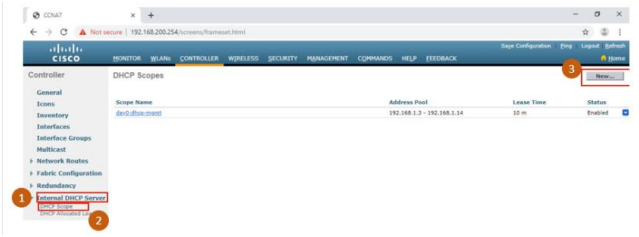
5. **Apply and Confirm:** Scroll to the top and click Apply and then click OK for the warning message

6. **Verify Interfaces:** Click Interfaces to verify that the new vlan5 interface is shown in the list of interfaces with its IPv4 address.

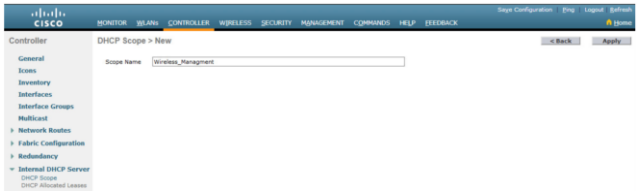
### *Configure a DHCP Scope*

DHCP scope configuration includes the following steps:

1. **Create a new DHCP scope:** To configure a new DHCP scope, click **Internal DHCP Server > DHCP Scope > New....**

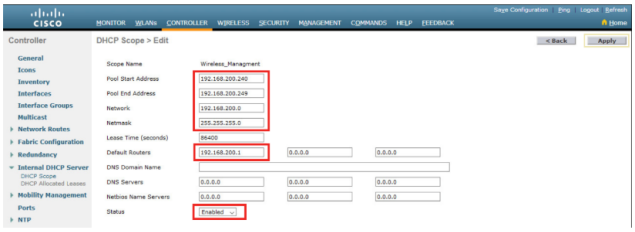


2. **Name the DHCP scope:** The scope is named **Wireless_Management** and then applied.

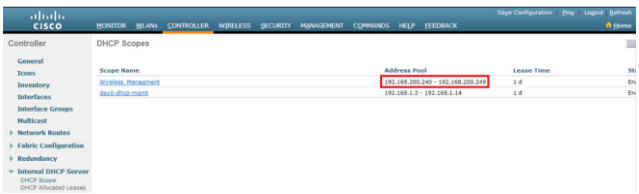


3. **Verify the new DHCP scope:** In the DHCP Scopes page click the new Scope Name to configure the DHCP scope.

4. **Configure and enable the new DHCP scope:** On the Edit screen for the Wireless_Management scope, configure a pool of addresses (i.e., 192.168.200.240/24 to .249), the default router IPv4 address (i.e., 192.168.200.1), then **Enabled** and **Apply.**



5. **Verify the enable DHCP scope:** The network administrator is returned to the **DHCP Scopes** page and can verify the scope is ready to be allocated to a new WLAN.



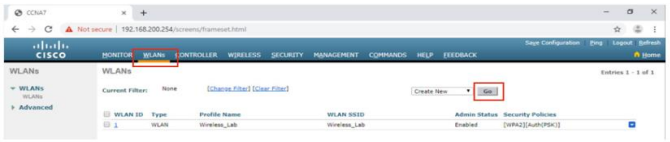
### *Configure a WPA2 Enterprise WLAN*

By default, all newly created WLANs on the WLC will use WPA2 with Advanced Encryption System (AES).

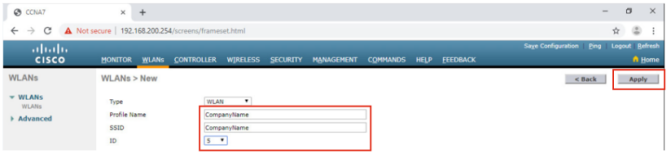
- 802.1X is the default key management protocol used to communicate with the RADIUS server.
- Next, create a new WLAN to use interface **vlan5**.

Configuring a new WLAN on the WLC includes the following steps:

1. **Create a new WLAN:** Click the **WLANs** tab and then **Go** to create a new WLAN.

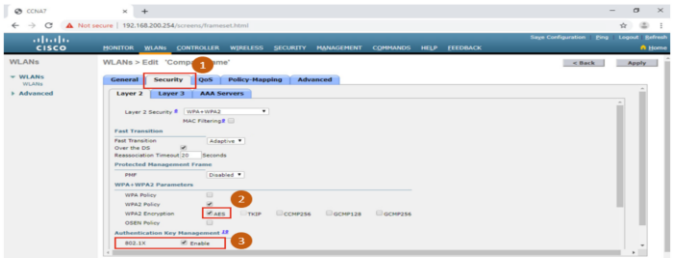


2. **Configure the WLAN name and SSID:** Enter the profile name and SSID, choose an ID of **5**, and then click **Apply** to create the new WLAN.

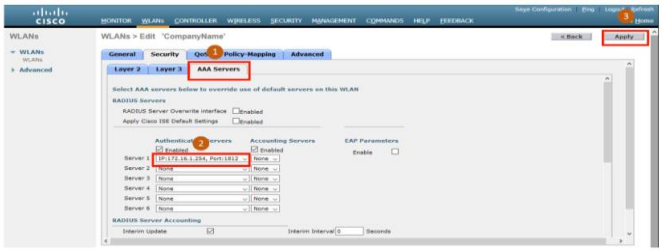


3. **Enable the WLAN for VLAN 5:** Once the WLAN, change the status to **Enabled**, choose **vlan5** from the Interface/Interface Group(G) dropdown list, and then click **Apply** and click **OK** to accept the popup message.

4. **Verify AES and 802.1X defaults:** Click the **Security** tab to view the default security configuration for the new WLAN.



5. **Configure the RADIUS server:** To select the RADIUS server that will be used to authenticate WLAN users, click the **AAA Servers** tab and in the dropdown box, select the RADIUS server that was configured on the WLC previously, and then **Apply** your changes.



6. **Verify that the new WLAN is available:** To verify that the new WLAN is listed and enabled click on the **WLANs** submenu.

## Troubleshoot WLAN Issues

### *Troubleshooting Approaches*

Network problems can be simple or complex, and can result from a combination of hardware, software, and connectivity issues.

- Technicians must be able to analyze the problem and determine the cause of the error before they can resolve the network issue.
- This process is called troubleshooting.

Troubleshooting any sort of network problem should follow a systematic approach.

A common and efficient troubleshooting methodology is based on the scientific method and can be broken into the six main steps shown in the table on the next slide.

| Step | Title | Description |
|---|---|---|
| 1 | Identify the Problem | The first step in the troubleshooting process is to identify the problem. While tools can be used in this step, a conversation with the user is often very helpful. |
| 2 | Establish a Theory of Probable Causes | After you have talked to the user and identified the problem, you can try and establish a theory of probable causes. This step often yields more than a few probable causes to the problem. |
| 3 | Test the Theory to Determine Cause | Based on the probable causes, test your theories to determine which one is the cause of the problem. A technician will often apply a quick procedure to test and see if it solves the problem. If a quick procedure does not correct the problem, you might need to research the problem further to establish the exact cause. |
| 4 | Establish a Plan of Action to Resolve the Problem and Implement the Solution | After you have determined the exact cause of the problem, establish a plan of action to resolve the problem and implement the solution. |
| 5 | Verify Full System Functionality and Implement Preventive Measures | After you have corrected the problem, verify full functionality and, if applicable, implement preventive measures. |
| 6 | Document Findings, Actions, and Outcomes | In the final step of the troubleshooting process, document your findings, actions, and outcomes. This is very important for future reference. |

### *Wireless Client Not Connecting*

If there is no connectivity, check the following:

- Confirm the network configuration on the PC using the **ipconfig** command.
- Confirm that the device can connect to the wired network. Ping a known IP address.
- If needed, reload drivers as appropriate for the client or try a different wireless NIC.
- If the wireless NIC of the client is working, check the security mode and encryption settings on the client.

If the PC is operational but the wireless connection is performing poorly, check the following:

- Is the PC out of the planned coverage area (BSA)?
- Check the channel settings on the wireless client.
- Check for interference with the 2.4 GHz band.

Next, ensure that all the devices are actually in place.

- Consider a possible physical security issue.
- Is there power to all devices and are they powered on?

Finally, inspect links between cabled devices looking for bad connectors or damaged or missing cables.

- If the physical plant is in place, verify the wired LAN by pinging devices, including the AP.
- If connectivity still fails at this point, perhaps something is wrong with the AP or its configuration.
- When the user PC is eliminated as the source of the problem, and the physical status of devices is confirmed, begin investigating the performance of the AP.
- Check the power status of the AP.

***Troubleshooting When the Network is Slow***

To optimize and increase the bandwidth of 802.11 dual-band routers and APs, either:

- **Upgrade your wireless clients** - Older 802.11b, 802.11g, and even 802.11n devices can slow the entire WLAN. For the best performance, all wireless devices should support the same highest acceptable standard.
- **Split the traffic** - The easiest way to improve wireless performance is to split the wireless traffic between the 802.11n 2.4 GHz band and the 5 GHz band. Therefore, 802.11n (or better) can use the two bands as two separate wireless networks to help manage the traffic.

There are several reasons for using a split-the-traffic approach:

- The 2.4 GHz band may be suitable for basic Internet traffic that is not time-sensitive.
- The bandwidth may still be shared with other nearby WLANs.
- The 5 GHz band is much less crowded than the 2.4 GHz band; ideal for streaming multimedia.
- The 5 GHz band has more channels; therefore, the channel chosen is likely interference-free.

By default, dual-band routers and APs use the same network name on both the 2.4 GHz band and the 5 GHz band.

- It may be useful to segment the traffic.
- The simplest way to segment traffic is to rename one of the wireless networks.

To improve the range of a wireless network, ensure the wireless router or AP location is free of obstructions, such as furniture, fixtures, and tall appliances.

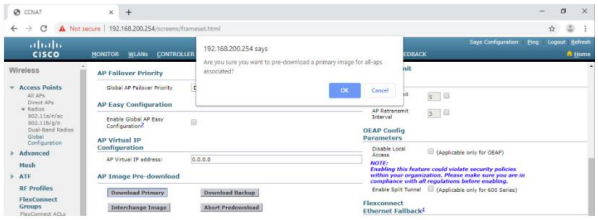
- These block the signal, which shortens the range of the WLAN.
- If this still does not solve the problem, then a Wi-Fi Range Extender or deploying the Powerline wireless technology may be used.

### *Updating Firmware*

Most wireless routers and APs offer upgradable firmware that should be periodically verified.

On a WLC, there will most likely be the ability to upgrade the firmware on all APs that the WLC controls.

- In the figure, the firmware image that will be used to upgrade all the APs is downloaded.
- On a Cisco 3504 Wireless Controller, click **WIRELESS > Access Points** > **Global Configuration** and then scroll to the bottom of the page for the AP Image Pre-download section.




**Laboratory Exercise: WLAN Configuration**

**Online Chapter Exam**

**Practical Exam**

**Module 14: Routing Concepts**

***Objectives:***

At the end of this module, the student should be able to:

- Explain how routers determine the best path;
- Explain how routers forward packets to the destination;
- Configure basic settings on a router;
- Describe the structure of a routing table;
- Compare static and dynamic routing concepts.

**Path Determination**

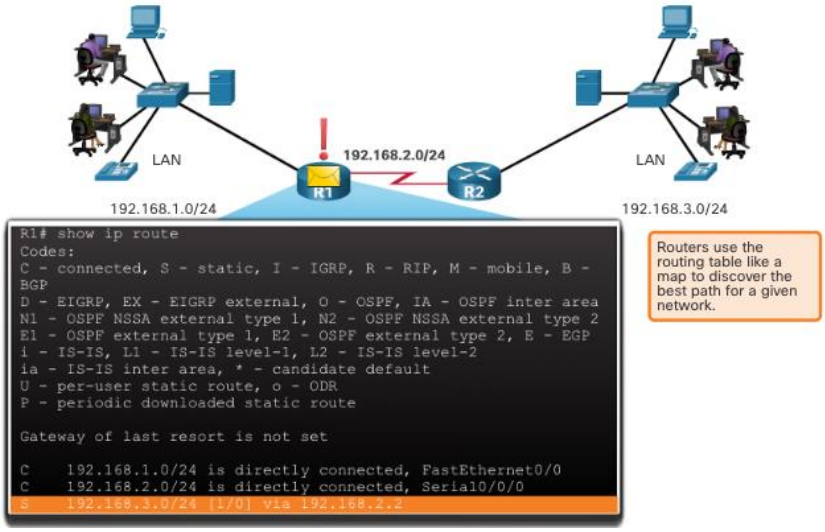
***Two Functions of a Router***

When a router receives an IP packet on one interface, it determines which interface to use to forward the packet to the destination. This is known as routing. The interface that the router uses to forward the packet may be the final destination, or it may be a network connected to another router that is used to reach the destination network. Each network that a router connects to typically requires a separate interface, but this may not always be the case.

The primary functions of a router are to determine the best path to forward packets based on the information in its routing table, and to forward packets toward their destination.

***Router Functions Example***

The router uses its IP routing table to determine which path (route) to use to forward a packet. R1 and R2 will use their respective IP routing tables to first determine the best path, and then forward the packet.



### *Best Path Equals Longest Match*

- The best path in the routing table is also known as the longest match.
- The routing table contains route entries consisting of a prefix (network address) and prefix length. For there to be a match between the destination IP address of a packet and a route in the routing table, a minimum number of far-left bits must match between the IP address of the packet and the route in the routing table. The prefix length of the route in the routing table is used to determine the minimum number of far-left bits that must match.
- The longest match is the route in the routing table that has the greatest number of far-left matching bits with the destination IP address of the packet. The longest match is always the preferred route.

**Note**: The term prefix length will be used to refer to the network portion of both IPv4 and IPv6 addresses.

### *IPv4 Longest Match Example*

In the table, an IPv4 packet has the destination IPv4 address 172.16.0.10. The router has three route entries in its IPv4 routing table that match this packet: 172.16.0.0/12, 172.16.0.0/18, and 172.16.0.0/26. Of the three routes, 172.16.0.0/26 has the longest match and would be chosen to forward the packet. For any of these routes to be considered a match there must be at least the number of matching bits indicated by the subnet mask of the route.

| **Destination IPv4 Address** | | **Address in Binary** |
|---|---|---|
| 172.16.0.10 | | **10101100.00010000.00000000.00**001010 |
| Route Entry | Prefix/Prefix Length | Address in Binary |
| 1 | 172.16.0.0**/12** | **10101100.0001**0000.00000000.00001010 |
| 2 | 172.16.0.0**/18** | **10101100.00010000.00**000000.00001010 |
| 3 | 172.16.0.0**/26** | **10101100.00010000.00000000.00**001010 |

### *IPv6 Longest Match Example*

An IPv6 packet has the destination IPv6 address 2001:db8:c000::99. This example shows three route entries, but only two of them are a valid match, with one of those being the longest match. The first two route entries have prefix lengths that have the required number of matching bits as indicated by the prefix length. The third route entry is not a match because its /64 prefix requires 64 matching bits.

| **Destination** | **2001:db8:c000::99/48** | |
|---|---|---|
| Route Entry | Prefix/Prefix Length | Does it match? |
| 1 | **2001:db8:c0**00::**/40** | Match of 40 bits |
| 2 | **2001:db8:c000**::**/48** | Match of 48 bits (longest match) |
| 3 | **2001:db8:c000**:5555::**/64** | Does not match 64 bits |

### *Build the Routing Table*

**Directly Connected Networks:** Added to the routing table when a local interface is configured with an IP address and subnet mask (prefix length) and is active (up and up).

**Remote Networks:** Networks that are not directly connected to the router. Routers learn about remote networks in two ways:

- **Static routes** - Added to the routing table when a route is manually configured.
- **Dynamic routing protocols** - Added to the routing table when routing protocols dynamically learn about the remote network.

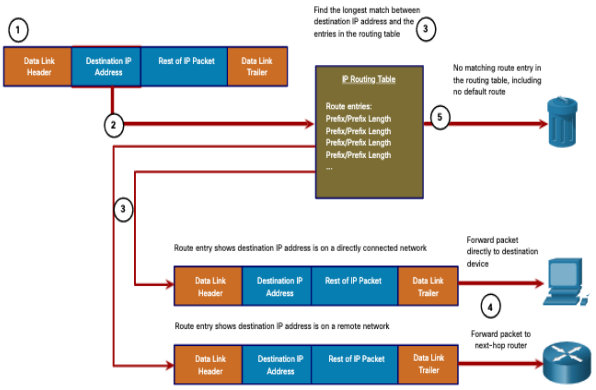
**Default Route:** Specifies a next-hop router to use when the routing table does not contain a specific route that matches the destination IP address. The default route can be entered manually as a static route, or learned automatically from a dynamic routing protocol.

- A default route has a /0 prefix length. This means that no bits need to match the destination IP address for this route entry to be used. If there are no routes with a match longer than 0 bits, the default route is used to forward the packet. The default route is sometimes referred to as a gateway of last resort.

## Packet Forwarding

### *Packet Forwarding Decision Process*

1. The data link frame with an encapsulated IP packet arrives on the ingress interface.
2. The router examines the destination IP address in the packet header and consults its IP routing table.
3. The router finds the longest matching prefix in the routing table.
4. The router encapsulates the packet in a data link frame and forwards it out the egress interface. The destination could be a device connected to the network or a next-hop router.
5. However, if there is no matching route entry the packet is dropped.




After a router has determined the best path, it could do the following:

### Forward the Packet to a Device on a Directly Connected Network

- If the route entry indicates that the egress interface is a directly connected network, the packet can be forwarded directly to the destination device. Typically this is an Ethernet LAN.
- To encapsulate the packet in the Ethernet frame, the router needs to determine the destination MAC address associated with the destination IP address of the packet. The process varies based on whether the packet is an IPv4 or IPv6 packet.

**Forward the Packet to a Next-Hop Router**

- If the route entry indicates that the destination IP address is on a remote network, meaning a device on network that is not directly connected. The packet must be forwarded to the next-hop router. The next-hop address is indicated in the route entry.
- If the forwarding router and the next-hop router are on an Ethernet network, a similar process (ARP and ICMPv6 Neighbor Discovery) will occur for determining the destination MAC address of the packet as described previously. The difference is that the router will search for the IP address of the next-hop router in its ARP table or neighbor cache, instead of the destination IP address of the packet.

**Note**: This process will vary for other types of Layer 2 networks.

**Drop the Packet - No Match in Routing Table**

- If there is no match between the destination IP address and a prefix in the routing table, and if there is no default route, the packet will be dropped.

***End-to-End Packet Forwarding***

The primary responsibility of the packet forwarding function is to encapsulate packets in the appropriate data link frame type for the outgoing interface. For example, the data link frame format for a serial link could be Point-to-Point (PPP) protocol, High-Level Data Link Control (HDLC) protocol, or some other Layer 2 protocol.

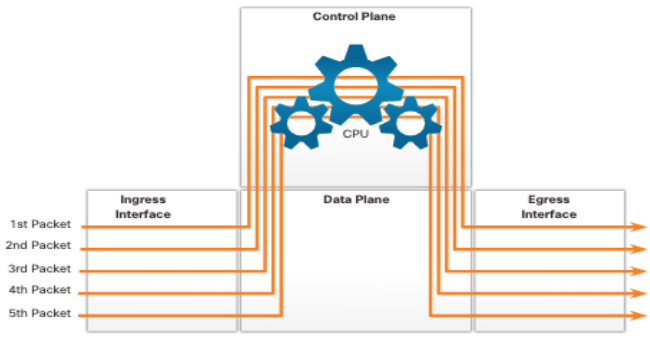
***Packet Forwarding Mechanisms***

The primary responsibility of the packet forwarding function is to encapsulate packets in the appropriate data link frame type for the outgoing interface. The more efficiently a router can perform this task, the faster packets can be forwarded by the router.

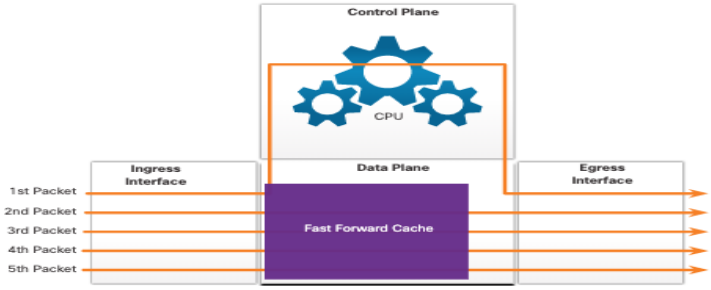
Routers support the following three packet forwarding mechanisms:

- Process switching
- Fast switching
- Cisco Express Forwarding (CEF)

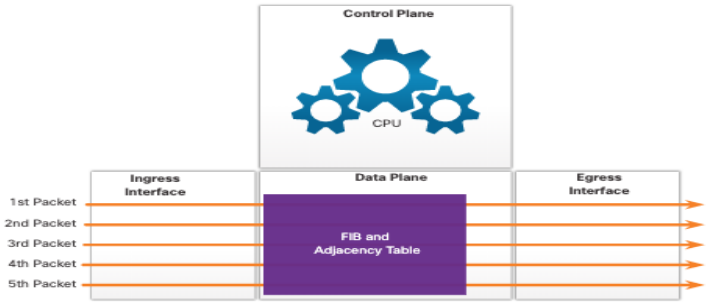
**Process Switching:** An older packet forwarding mechanism still available for Cisco routers. When a packet arrives on an interface, it is forwarded to the control plane where the CPU matches the destination address with an entry in its routing table, and then determines the exit interface and forwards the packet. It is important to understand that the router does this for every packet, even if the destination is the same for a stream of packets.



**Fast Switching:** Another, older packet forwarding mechanism which was the successor to process switching. Fast switching uses a fast-switching cache to store next-hop information. When a packet arrives on an interface, it is forwarded to the control plane where the CPU searches for a match in the fast-switching cache. If it is not there, it is process-switched and forwarded to the exit interface. The flow information for the packet is then stored in the fast-switching cache. If another packet going to the same destination arrives on an interface, the next-hop information in the cache is re-used without CPU intervention.



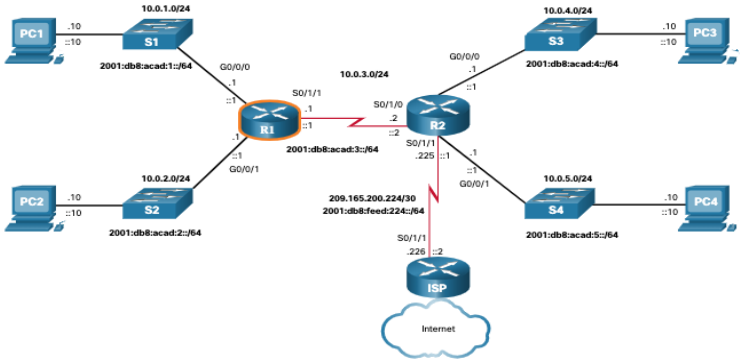

**Cisco Express Forwarding (CEF):** The most recent and default Cisco IOS packet-forwarding mechanism. CEF builds a Forwarding Information Base (FIB), and an adjacency table. The table entries are not packet-triggered like fast switching but change-triggered, such as when something changes in the network topology. When a network has converged, the FIB and adjacency tables contain all the information that a router would have to consider when forwarding a packet.




## Basic Router Configuration Review

### *Topology*

The topology in the figure will be used for configuration and verification examples. It will also be used in the next topic to discuss the IP routing table.



### *Configuration Commands*

```
Router> enable
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# hostname R1
R1(config)# enable secret class
R1(config)# line console 0
R1(config-line)# logging synchronous
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# transport input ssh telnet
R1(config-line)# exit
R1(config)# service password-encryption
R1(config)# banner motd #
Enter TEXT message. End with a new line and the #
********************************************** WARNING:
Unauthorized access is prohibited!
**********************************************
#
R1(config)# ipv6 unicast-routing
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# description Link to LAN 1
R1(config-if)# ip address 10.0.1.1 255.255.255.0
R1(config-if)# ipv6 address 2001:db8:acad:1::1/64
R1(config-if)# ipv6 address fe80::1:a link-local R1(config-
if)# no shutdown
R1(config-if)# exit
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# description Link to LAN 2
R1(config-if)# ip address 10.0.2.1 255.255.255.0 R1(config-
if)# ipv6 address 2001:db8:acad:2::1/64 R1(config-if)# ipv6
address fe80::1:b link-local R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface serial 0/1/1
R1(config-if)# description Link to R2
R1(config-if)# ip address 10.0.3.1 255.255.255.0 R1(config-
if)# ipv6 address 2001:db8:acad:3::1/64 R1(config-if)# ipv6
address fe80::1:c link-local R1(config-if)# no shutdown
R1(config-if)# exit
R1# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
```

### *Verification Commands*

Common verification commands include the following:

- **show ip interface brief**
- **show running-config interface** *interface-type number*
- **show interfaces**
- **show ip interface**
- **show ip route**
- **ping**

In each case, replace **ip** with **ipv6** for the IPv6 version of the command.

Filter Command Output

Filtering commands can be used to display specific sections of output. To enable the filtering command, enter a pipe (**|**) character after the **show** command and then enter a filtering parameter and a filtering expression.
The filtering parameters that can be configured after the pipe include:

- **section** - This displays the entire section that starts with the filtering expression.
- **include** - This includes all output lines that match the filtering expression.
- **exclude** - This excludes all output lines that match the filtering expression.
- **begin** - This displays all the output lines from a certain point, starting with the line that matches the filtering expression.

**Note**: Output filters can be used in combination with any **show** command.

## IP Routing Table

### *Route Sources*

A routing table contains a list of routes to known networks (prefixes and prefix lengths). The source of this information is derived from the following:

- Directly connected networks
- Static routes
- Dynamic routing protocols

The source for each route in the routing table is identified by a code. Common codes include the following:

- **L** - Identifies the address assigned to a router interface.
- **C** - Identifies a directly connected network.
- **S** - Identifies a static route created to reach a specific network.
- **O** - Identifies a dynamically learned network from another router using the OSPF routing protocol.

***** - This route is a candidate for a default route.

### *Routing Table Principles*

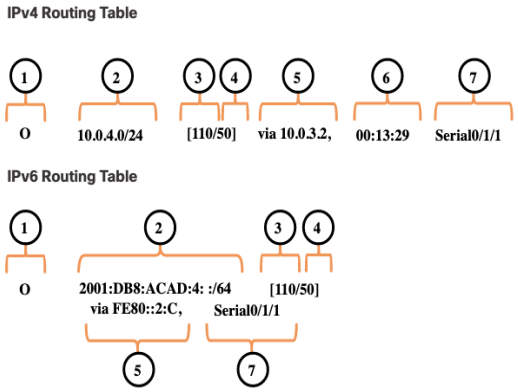
There are three routing table principles as described in the table. These are issues that are addressed by the proper configuration of dynamic routing protocols or static routes on all the routers between the source and destination devices.

| Routing Table Principle | Example |
|---|---|
| Every router makes its decision alone, based on the information it has in its own routing table. | • R1 can only forward packets using its own routing table.<br>• R1 does not know what routes are in the routing tables of other routers (e.g., R2). |
| The information in a routing table of one router does not necessarily match the routing table of another router. | Just because R1 has route in its routing table to a network in the internet via R2, that does not mean that R2 knows about that same network. |
| Routing information about a path does not provide return routing information. | R1 receives a packet with the destination IP address of PC1 and the source IP address of PC3. Just because R1 knows to forward the packet out its G0/0/0 interface, doesn't necessarily mean that it knows how to forward packets originating from PC1 back to the remote network of PC3 |

### *Routing Table Entries*

In the figure, the numbers identify the following information:

- **Route source** - This identifies how the route was learned.
- **Destination network (prefix and prefix length)** - This identifies the address of the remote network.
- **Administrative distance** - This identifies the trustworthiness of the route source. Lower values indicate preferred route source.
- **Metric** - This identifies the value assigned to reach the remote network. Lower values indicate preferred routes.
- **Next-hop** - This identifies the IP address of the next router to which the packet would be forwarded.
- **Route timestamp** - This identifies how much time has passed since the route was learned.
- **Exit interface** - This identifies the egress interface to use for outgoing packets to reach their final destination.




**Note**: The prefix length of the destination network specifies the minimum number of far-left bits that must match between the IP address of the packet and the destination network (prefix) for this route to be used.

### *Directly Connected Networks*

To learn about any remote networks, the router must have at least one active interface configured with an IP address and subnet mask (prefix length). This is known as a directly connected network or a directly connected route. Routers add a directly connected route to its routing table when an interface is configured with an IP address and is activated.

- A directly connected network is denoted by a status code of **C** in the routing table. The route contains a network prefix and prefix length.
- The routing table also contains a local route for each of its directly connected networks, indicated by the status code of **L**.
- For IPv4 local routes the prefix length is /32 and for IPv6 local routes the prefix length is /128. This means the destination IP address of the packet must match all the bits in the local route for this route to be a match. The purpose of the local route is to efficiently determine when it receives a packet for the interface instead of a packet that needs to be forwarded.

### *Static Routes*

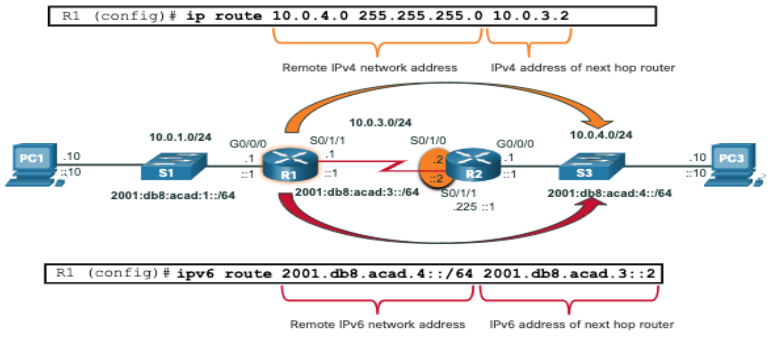
After directly connected interfaces are configured and added to the routing table, static or dynamic routing can be implemented for accessing remote networks. Static routes are manually configured. They define an explicit path between two networking devices. They are not automatically updated and must be manually reconfigured if the network topology changes.

Static routing has three primary uses:

- It provides ease of routing table maintenance in smaller networks that are not expected to grow significantly.
- It uses a single default route to represent a path to any network that does not have a more specific match with another route in the routing table. Default routes are used to send traffic to any destination beyond the next upstream router.
- It routes to and from stub networks. A stub network is a network accessed by a single route, and the router has only one neighbor.

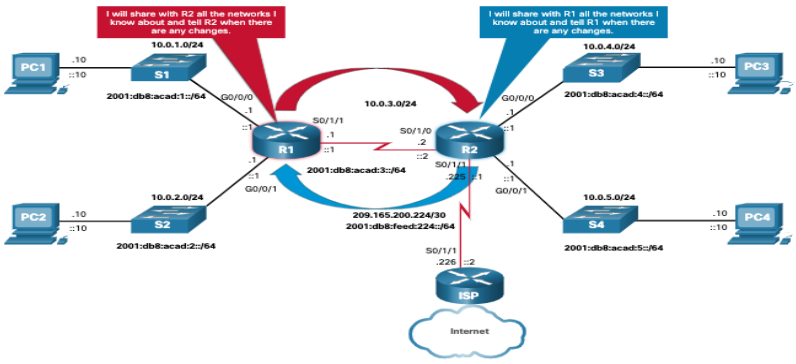
### *Static Routes in the IP Routing Table*

The topology in the figure is simplified to show only one LAN attached to each router. The figure shows IPv4 and IPv6 static routes configured on R1 to reach the 10.0.4.0/24 and 2001:db8:acad:4::/64 networks on R2.




### *Dynamic Routing Protocols*

Dynamic routing protocols are used by routers to automatically share information about the reachability and status of remote networks. Dynamic routing protocols perform several activities, including network discovery and maintaining routing tables.



### *Dynamic Routes in the Routing Table*

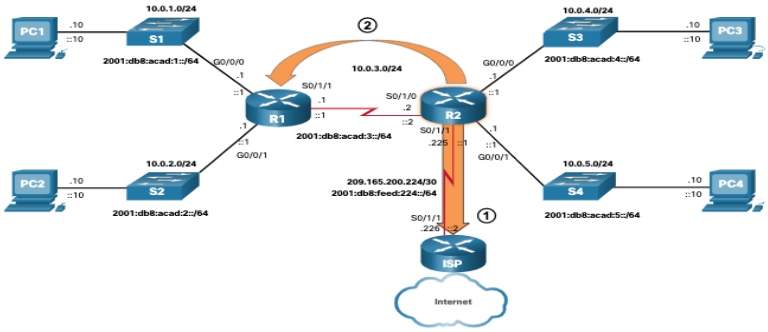
OSPF is now being used in our sample topology to dynamically learn all the networks connected to R1 and R2. The routing table entries use the status code of **O** to indicate the route was learned by the OSPF routing protocol. Both entries also include the IP address of the next-hop router, via *ip-address*.
**Note**: IPv6 routing protocols use the link-local address of the next-hop router.

**Note**: OSPF routing configuration for IPv4 and IPv6 is beyond the scope of this course.

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP D
- EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
(output omitted for brevity)
O 10.0.4.0/24 [110/50] via 10.0.3.2, 00:24:22, Serial0/1/1
O 10.0.5.0/24 [110/50] via 10.0.3.2, 00:24:15, Serial0/1/1
R1# show ipv6 route
IPv6 Routing Table - default - 10 entries
(Output omitted)
NDr - Redirect, RL - RPL, O - OSPF Intra, OI - OSPF Inter
O 2001:DB8:ACAD:4::/64 [110/50]
   via FE80::2:C, Serial0/1/1
O 2001:DB8:ACAD:5::/64 [110/50]
   via FE80::2:C, Serial0/1/1
```

### *Default Route*

The default route specifies a next-hop router to use when the routing table does not contain a specific route that matches the destination IP address. A default route can be either a static route or learned automatically from a dynamic routing protocol. A default route has an IPv4 route entry of 0.0.0.0/0 or an IPv6 route entry of ::/0. This means that zero or no bits need to match between the destination IP address and the default route.




### *Structure of an IPv4 Routing Table*

IPv4 was standardized using the now obsolete classful addressing architecture. The IPv4 routing table is organized using this same classful structure. Although the lookup process no longer uses classes, the structure of the IPv4 routing table still retains in this format.

An indented entry is known as a child route. A route entry is indented if it is the subnet of a classful address (class A, B or C network). Directly connected networks will always be indented (child routes) because the local address of the interface is always entered in the routing table as a /32. The child route will include the route source and all the forwarding information such as the next-hop address. The classful network address of this subnet will be shown above the route entry, less indented, and without a source code. That route is known as a parent route.

- An indented entry is known as a **child route**. A route entry is indented if it is the subnet of a classful address (class A, B or C network).
- Directly connected networks will always be indented (child routes) because the local address of the interface is always entered in the routing table as a /32.
- The child route will include the route source and all the forwarding information such as the next-hop address.
- The classful network address of this subnet will be shown above the route entry, less indented, and without a source code. That route is known as a **parent route.**

```
Router# show ip route
(Output omitted)
   192.168.1.0/24 is variably..
C     192.168.1.0/24 is direct..
L     192.168.1.1/32 is direct..
O     192.168.2.0/24 [110/65]..
O     192.168.3.0/24 [110/65]..
   192.168.12.0/24 is variab..
C     192.168.12.0/30 is direct..
L     192.168.12.1/32 is direct..
   192.168.13.0/24 is variably..
C     192.168.13.0/30 is direct..
L     192.168.13.1/32 is direct..
   192.168.23.0/30 is subnette..
O     192.168.23.0/30 [110/128]..
Router#
```

### *Structure of an IPv6 Routing Table*

The concept of classful addressing was never part of IPv6, so the structure of an IPv6 routing table is very straight forward. Every IPv6 route entry is formatted and aligned the same way.

```
R1# show ipv6 route
(output omitted for brevity)
OE2 ::/0 [110/1], tag 2
   via FE80::2:C, Serial0/0/1
C 2001:DB8:ACAD:1::/64 [0/0]
   via GigabitEthernet0/0/0, directly connected
L 2001:DB8:ACAD:1::1/128 [0/0]
   via GigabitEthernet0/0/0, receive
C 2001:DB8:ACAD:2::/64 [0/0]
  via GigabitEthernet0/0/1, directly connected
L 2001:DB8:ACAD:2::1/128 [0/0]
   via GigabitEthernet0/0/1, receive
C 2001:DB8:ACAD:3::/64 [0/0]
   via Serial0/1/1, directly connected
L 2001:DB8:ACAD:3::1/128 [0/0]
   via Serial0/1/1, receive
O 2001:DB8:ACAD:4::/64 [110/50]
   via FE80::2:C, Serial0/1/1
O 2001:DB8:ACAD:5::/64 [110/50]
   via FE80::2:C, Serial0/1/1
L FF00::/8 [0/0]
   via Null0, receive
R1#
```

### *Administrative Distance*

A route entry for a specific network address (prefix and prefix length) can only appear once in the routing table. However, it is possible that the routing table learns about the same network address from more than one routing source. Except for very specific circumstances, only one dynamic routing protocol should be implemented on a router. Each routing protocol may decide on a different path to reach the destination based on the metric of that routing protocol.

This raises a few questions, such as the following:
- How does the router know which source to use?
- Which route should it install in the routing table?

Cisco IOS uses what is known as the administrative distance (AD) to determine the route to install into the IP routing table. The AD represents the "trustworthiness" of the route. The lower the AD, the more trustworthy the route source.

The table lists various routing protocols and their associated ADs.

| Route Source | Administrative Distance |
|---|---|
| Directly connected | 0 |
| Static route | 1 |
| EIGRP summary route | 5 |
| External BGP | 20 |
| Internal EIGRP | 90 |
| OSPF | 110 |
| IS-IS | 115 |
| RIP | 120 |
| External EIGRP | 170 |
| Internal BGP | 200 |

**Static and Dynamic Routing**

***Static or Dynamic?***

Static and dynamic routing are not mutually exclusive. Rather, most networks use a combination of dynamic routing protocols and static routes.

Static routes are commonly used in the following scenarios:

- As a default route forwarding packets to a service provider
- For routes outside the routing domain and not learned by the dynamic routing protocol
- When the network administrator wants to explicitly define the path for a specific network
- For routing between stub networks

Static routes are useful for smaller networks with only one path to an outside network. They also provide security in a larger network for certain types of traffic, or links to other networks that need more control.

Dynamic routing protocols are implemented in any type of network consisting of more than just a few routers. Dynamic routing protocols are scalable and automatically determine better routes if there is a change in the topology.

Dynamic routing protocols are commonly used in the following scenarios:

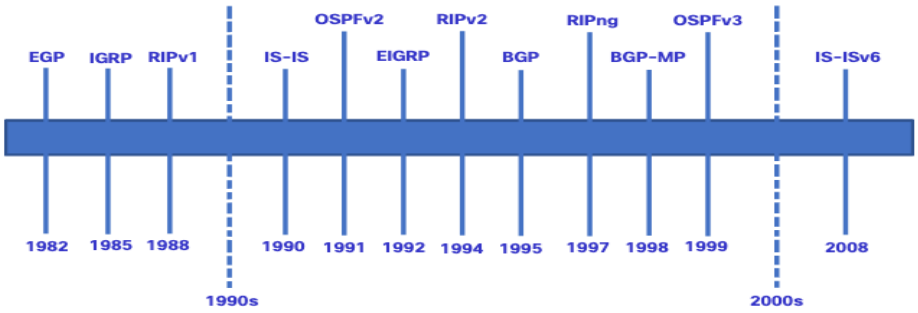
- In networks consisting of more than just a few routers
- When a change in the network topology requires the network to automatically determine another path
- For scalability. As the network grows, the dynamic routing protocol automatically learns about any new networks.

The table shows a comparison of some the differences between dynamic and static routing.

| Feature | Dynamic Routing | Static Routing |
|---|---|---|
| Configuration complexity | Independent of network size | Increases with network size |
| Topology changes | Automatically adapts to topology changes | Administrator intervention required |
| Scalability | Suitable for simple to complex network topologies | Suitable for simple topologies |
| Security | Security must be configured | Security is inherent |
| Resource Usage | Uses CPU, memory, and link bandwidth | No additional resources needed |
| Path Predictability | Route depends on topology and routing protocol used | Explicitly defined by the administrator |

Dynamic Routing Evolution

Dynamic routing protocols have been used in networks since the late 1980s. One of the first routing protocols was RIP. RIPv1 was released in 1988, but some of the basic algorithms within the protocol were used on the Advanced Research Projects Agency Network (ARPANET) as early as 1969. As networks evolved and became more complex, new routing protocols emerged.




The table classifies the current routing protocols. Interior Gateway Protocols (IGPs) are routing protocols used to exchange routing information within a routing domain administered by a single organization. There is only one EGP and it is BGP. BGP is used to exchange routing information between different organizations, known as autonomous systems (AS). BGP is used by ISPs to route packets over the internet. Distance vector, link-state, and path vector routing protocols refer to the type of routing algorithm used to determine best path.

| | Interior Gateway Protocols | | | | Exterior Gateway Protocols |
|---|---|---|---|---|---|
| | Distance Vector | | Link-State | | Path Vector |
| IPv4 | RIPv2 | EIGRP | OSPFv2 | IS-IS | BGP-4 |
| IPv6 | RIPng | EIGRP for IPv6 | OSPFv3 | IS-IS for IPv6 | BGP-MP |

### *Dynamic Routing Protocol Concepts*

A routing protocol is a set of processes, algorithms, and messages that are used to exchange routing information and populate the routing table with the choice of best paths. The purpose of dynamic routing protocols includes the following:

- Discovery of remote networks
- Maintaining up-to-date routing information
- Choosing the best path to destination networks
- Ability to find a new best path if the current path is no longer available

The main components of dynamic routing protocols include the following:

- **Data structures -** Routing protocols typically use tables or databases for their operations. This information is kept in RAM.
- **Routing protocol messages -** Routing protocols use various types of messages to discover neighboring routers, exchange routing information, and other tasks to learn and maintain accurate information about the network.
- **Algorithm -** An algorithm is a finite list of steps used to accomplish a task. Routing protocols use algorithms for facilitating routing information and for the best path determination.

Routing protocols determine the best path, or route, to each network. That route is then offered to the routing table. The route will be installed in the routing table if there is not another routing source with a lower AD.

### *Best Path*

The best path is selected by a routing protocol based on the value or metric it uses to determine the distance to reach a network. A metric is the quantitative value used to measure the distance to a given network. The best path to a network is the path with the lowest metric.

Dynamic routing protocols typically use their own rules and metrics to build and update routing tables. The following table lists common dynamic protocols and their metrics.

| Routing Protocol | Metric |
|---|---|
| **Routing Information Protocol (RIP)** | • The metric is “hop count”.<br>• Each router along a path adds a hop to the hop count.<br>• A maximum of 15 hops allowed. |
| **Open Shortest Path First (OSPF)** | • The metric is “cost” which is the based on the cumulative bandwidth from source to destination.<br>• Faster links are assigned lower costs compared to slower (higher cost) links. |
| **Enhanced Interior Gateway Routing Protocol (EIGRP)** | • It calculates a metric based on the slowest bandwidth and delay values.<br>• It could also include load and reliability into the metric calculation. |

***Load Balancing***

When a router has two or more paths to a destination with equal cost metrics, then the router forwards the packets using both paths equally. This is called equal cost load balancing.

- The routing table contains the single destination network, but has multiple exit interfaces, one for each equal cost path. The router forwards packets using the multiple exit interfaces listed in the routing table.
- If configured correctly, load balancing can increase the effectiveness and performance of the network.
- Equal cost load balancing is implemented automatically by dynamic routing protocols. It is enabled with static routes when there are multiple static routes to the same destination network using different next-hop routers.

**Note**: Only EIGRP supports unequal cost load balancing.

**Laboratory Exercise: Static and Dynamic Routing Simulation**

## Module 15: IP Static Routing

***Objectives:***

At the end of this module, the student should be able to:

- Describe the command syntax for static routes;
- Configure IPv4 and IPv6 static routes;
- Configure IPv4 and IPv6 default static routes;
- Configure a floating static route to provide a backup connection;
- Configure IPv4 and IPv6 static host routes that direct traffic to a specific host.

### Static Rotes

***Types of Static Routes***

Static routes are commonly implemented on a network. This is true even when there is a dynamic routing protocol configured.

Static routes can be configured for IPv4 and IPv6. Both protocols support the following types of static routes:

- Standard static route
- Default static route
- Floating static route
- Summary static route

Static routes are configured using the **ip route** and **ipv6 route** global configuration commands.

***Next-Hop Options***

When configuring a static route, the next hop can be identified by an IP address, exit interface, or both. How the destination is specified creates one of the three following types of static route:

- **Next-hop route** - Only the next-hop IP address is specified
- **Directly connected static route** - Only the router exit interface is specified
- **Fully specified static route** - The next-hop IP address and exit interface are specified

***IPv4 Static Route Command***

IPv4 static routes are configured using the following global configuration command:

**Router(config)# ip route** *network-address subnet-mask* { *ip-address* | *exit-intf* [*ip-address*]} [distance]

**Note:** Either the *ip-address*, *exit-intf,* or the *ip-address* and *exit-intf* parameters must be configured.

### *IPv6 Static Route Command*

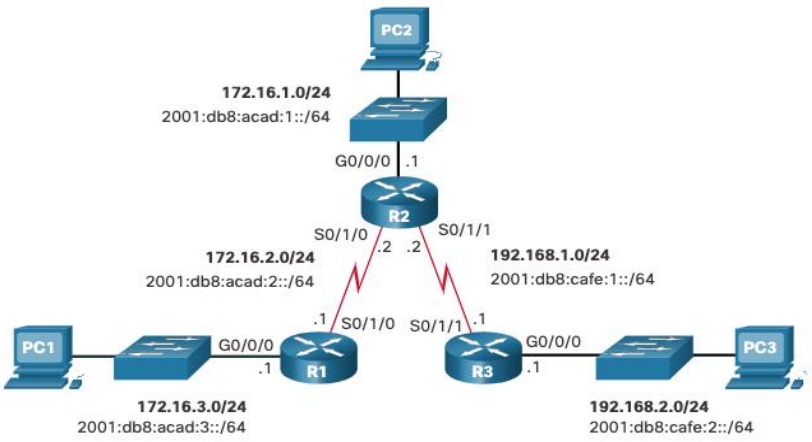
IPv6 static routes are configured using the following global configuration command:

**Router(config)# ipv6 route** *ipv6-prefix/prefix-length* {*ipv6-address* | *exit-intf* [*ipv6-address*]} [*distance*]

Most of parameters are identical to the IPv4 version of the command.

### *Dual-Stack Topology*

The figure shows a dual-stack network topology. Currently, no static routes are configured for either IPv4 or IPv6.




### *IPv4 Starting Routing Tables*

- Each router has entries only for directly connected networks and associated local addresses.
- R1 can ping R2, but cannot ping the R3 LAN

```
R1# show ip route | begin Gateway
Gateway of last resort is not set
     172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C       172.16.2.0/24 is directly connected, Serial0/1/0
L       172.16.2.1/32 is directly connected, Serial0/1/0
C       172.16.3.0/24 is directly connected, GigabitEthernet0/0/0
L       172.16.3.1/32 is directly connected, GigabitEthernet0/0/0
R1#
R1# ping 172.16.2.2
Type escape sequence to abort. Sending 5, 100-byte ICMP Echos to 172.16.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5)
R1# ping 192.168.2.1
Type escape sequence to abort. Sending 5, 100-byte ICMP Echos to 192.168.2.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

### *IPv6 Starting Routing Tables*

- Each router has entries only for directly connected networks and associated local addresses.
- R1 can ping R2, but cannot ping the R3 LAN.

```
R1# show ipv6 route | begin C
C 2001:DB8:ACAD:2::/64 [0/0]
        via Serial0/1/0, directly connected
L 2001:DB8:ACAD:2::1/128 [0/0]
        via Serial0/1/0, receive
C 2001:DB8:ACAD:3::/64 [0/0]
        via GigabitEthernet0/0/0, directly connected
L 2001:DB8:ACAD:3::1/128 [0/0]
        via GigabitEthernet0/0/0, receive
L FF00::/8 [0/0]
        via Null0, receive
R1#
R1# ping 2001:db8:acad:2::2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:ACAD:2::2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms)
R1# ping 2001:DB8:cafe:2::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:CAFE:2::1, timeout is 2 seconds:
% No valid route for destination
Success rate is 0 percent (0/1)
```

## Configure IP Static Routes

### *IPv4 Next-Hop Static Rote*

In a next-hop static route, only the next-hop IP address is specified. The exit interface is derived from the next hop. For example, three next-hop IPv4 static routes are configured on R1 using the IP address of the next hop, R2.

R1(config)# **ip route 172.16.1.0 255.255.255.0 172.16.2.2**

R1(config)# **ip route 192.168.1.0 255.255.255.0 172.16.2.2**

R1(config)# **ip route 192.168.2.0 255.255.255.0 172.16.2.2**

The resulting routing table entries on R1:

```
R1# show ip route | begin Gateway
Gateway of last resort is not set
      172.16.0.0/16 is variably subnetted, 5 subnets, 2 masks
S        172.16.1.0/24 [1/0] via 172.16.2.2
C        172.16.2.0/24 is directly connected, Serial0/1/0
L        172.16.2.1/32 is directly connected, Serial0/1/0
C        172.16.3.0/24 is directly connected, GigabitEthernet0/0/0
L        172.16.3.1/32 is directly connected, GigabitEthernet0/0/0
S     192.168.1.0/24 [1/0] via 172.16.2.2
S     192.168.2.0/24 [1/0] via 172.16.2.2
```

### *IPv6 Next-Hop Static Route*

The commands to configure R1 with the IPv6 static routes to the three remote networks are as follows:

R1(config)# **ipv6 unicast-routing**
R1(config)# **ipv6 route 2001:db8:acad:1::/64 2001:db8:acad:2::2**
R1(config)# **ipv6 route 2001:db8:cafe:1::/64 2001:db8:acad:2::2**
R1(config)# **ipv6 route 2001:db8:cafe:2::/64 2001:db8:acad:2::2**

The routing table for R1 now has routes to the three remote IPv6 networks.

```
R1# show ipv6 route
IPv6 Routing Table - default - 8 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
       NDr - Redirect, RL - RPL, O - OSPF Intra, OI - OSPF Inter
       OE1 - OSPF ext 1, OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1
       ON2 - OSPF NSSA ext 2, la - LISP alt, lr - LISP site-registrations
       ld - LISP dyn-eid, lA - LISP away, le - LISP extranet-policy
       a - Application
S   2001:DB8:ACAD:1::/64 [1/0]
     via 2001:DB8:ACAD:2::2
C   2001:DB8:ACAD:2::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:DB8:ACAD:2::1/128 [0/0]
     via Serial0/1/0, receive
C   2001:DB8:ACAD:3::/64 [0/0]
     via GigabitEthernet0/0/0, directly connected
L   2001:DB8:ACAD:3::1/128 [0/0]
     via GigabitEthernet0/0/0, receive
S   2001:DB8:CAFE:1::/64 [1/0]
     via 2001:DB8:ACAD:2::2
S   2001:DB8:CAFE:2::/64 [1/0]
     via 2001:DB8:ACAD:2::2
L   FF00::/8 [0/0]
     via Null0, receive
```

### *IPv4 Directly Connected Static Route*

When configuring a static route, another option is to use the exit interface to specify the next-hop address. Three directly connected IPv4 static routes are configured on R1 using the exit interface.

**Note**: Using a next-hop address is generally recommended. Directly connected static routes should only be used with point-to-point serial interfaces.

**R1(config)# ip route 172.16.1.0 255.255.255.0 s0/1/0**
**R1(config)# ip route 192.168.1.0 255.255.255.0 s0/1/0**
**R1(config)# ip route 192.168.2.0 255.255.255.0 s0/1/0**

```
R1# show ip route | begin Gateway
Gateway of last resort is not set
      172.16.0.0/16 is variably subnetted, 5 subnets, 2 masks
S        172.16.1.0/24 is directly connected, Serial0/1/0
C        172.16.2.0/24 is directly connected, Serial0/1/0
L        172.16.2.1/32 is directly connected, Serial0/1/0
C        172.16.3.0/24 is directly connected, GigabitEthernet0/0/0
L        172.16.3.1/32 is directly connected, GigabitEthernet0/0/0
S     192.168.1.0/24 is directly connected, Serial0/1/0
S     192.168.2.0/24 is directly connected, Serial0/1/0
```

### *IPv6 Directly Connected Static Route*

In the example, three directly connected IPv6 static routes are configured on R1 using the exit interface.

**Note**: Using a next-hop address is generally recommended. Directly connected static routes should only be used with point-to-point serial interfaces.

**R1(config)# ipv6 route 2001:db8:acad:1::/64 s0/1/0**
**R1(config)# ipv6 route 2001:db8:cafe:1::/64 s0/1/0**
**R1(config)# ipv6 route 2001:db8:cafe:2::/64 s0/1/0**

```
R1# show ipv6 route
IPv6 Routing Table - default - 8 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
       NDr - Redirect, RL - RPL, O - OSPF Intra, OI - OSPF Inter
       OE1 - OSPF ext 1, OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1
       ON2 - OSPF NSSA ext 2, la - LISP alt, lr - LISP site-registrations
       ld - LISP dyn-eid, lA - LISP away, le - LISP extranet-policy
       a - Application
S   2001:DB8:ACAD:1::/64 [1/0]
     via Serial0/1/0, directly connected
C   2001:DB8:ACAD:2::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:DB8:ACAD:2::1/128 [0/0]
     via Serial0/1/0, receive
C   2001:DB8:ACAD:3::/64 [0/0]
     via GigabitEthernet0/0/0, directly connected
L   2001:DB8:ACAD:3::1/128 [0/0]
     via GigabitEthernet0/0/0, receive
S   2001:DB8:CAFE:1::/64 [1/0]
     via Serial0/1/0, directly connected
S   2001:DB8:CAFE:2::/64 [1/0]
     via Serial0/1/0, directly connected
L   FF00::/8 [0/0]
     via Null0, receiveIPv6 Routing Table - default - 8 entries
R1#
```

### *IPv4 Fully Specified Static Route*

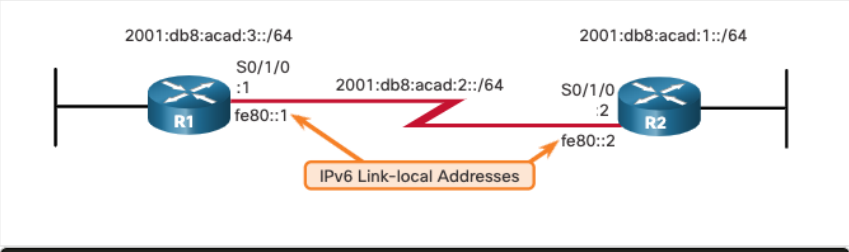
- In a fully specified static route, both the exit interface and the next-hop IP address are specified. This form of static route is used when the exit interface is a multi-access interface and it is necessary to explicitly identify the next hop. The next hop must be directly connected to the specified exit interface. Using an exit interface is optional, however it is necessary to use a next-hop address.
- It is recommended that when the exit interface is an Ethernet network, that the static route includes a next-hop address. You can also use a fully specified static route that includes both the exit interface and the next-hop address.

```
R1(config)# ip route 172.16.1.0 255.255.255.0 GigabitEthernet 0/0/1 172.16.2.2
R1(config)# ip route 192.168.1.0 255.255.255.0 GigabitEthernet 0/0/1 172.16.2.2
R1(config)# ip route 192.168.2.0 255.255.255.0 GigabitEthernet 0/0/1 172.16.2.2
```

```
R1# show ip route | begin Gateway
Gateway of last resort is not set
      172.16.0.0/16 is variably subnetted, 5 subnets, 2 masks
S        172.16.1.0/24 [1/0] via 172.16.2.2, GigabitEthernet0/0/1
C        172.16.2.0/24 is directly connected, GigabitEthernet0/0/1
L        172.16.2.1/32 is directly connected, GigabitEthernet0/0/1
C        172.16.3.0/24 is directly connected, GigabitEthernet0/0/0
L        172.16.3.1/32 is directly connected, GigabitEthernet0/0/0
S     192.168.1.0/24 [1/0] via 172.16.2.2, GigabitEthernet0/0/1
S     192.168.2.0/24 [1/0] via 172.16.2.2, GigabitEthernet0/0/1
```

### *IPv6 Fully Specified Static Route*

In a fully specified static route, both the exit interface and the next-hop IPV6 address are specified.
There is a situation in IPv6 when a fully specified static route must be used. If the IPv6 static route uses an IPv6 link-local address as the next-hop address, use a fully specified static route. The figure shows an example of a fully specified IPv6 static route using an IPv6 link-local address as the next-hop address.




```
R1(config)# ipv6 route 2001:db8:acad:1::/64 fe80::2
%Interface has to be specified for a link-local nexthop
R1(config)# ipv6 route 2001:db8:acad:1::/64 s0/1/0 fe80::2
```

The reason a fully specified static route must be used is because IPv6 link-local addresses are not contained in the IPv6 routing table. Link-local addresses are only unique on a given link or network. The next-hop link-local address may be a valid address on multiple networks connected to the router. Therefore, it is necessary that the exit interface be included.
The following example shows the IPv6 routing table entry for this route. Notice that both the next-hop link-local address and the exit interface are included.

```
R1# show ipv6 route static | begin 2001:db8:acad:1::/64
S   2001:DB8:ACAD:1::/64 [1/0]
    via FE80::2, Seria0/1/0
```

***Verify a Static Route***

Along with **show ip route**, **show ipv6 route**, **ping** and **traceroute**, other useful commands to verify static routes include the following:

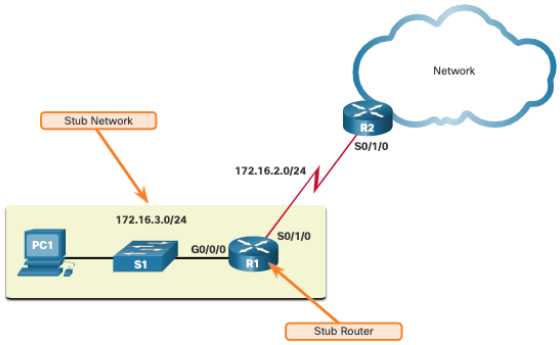
- **show ip route static**
- **show ip route** *network*
- **show running-config | section ip route**

Replace **ip** with **ipv6** for the IPv6 versions of the command.

**Configure IP Default Static Routes**

***Default Static Route***

- A default route is a static route that matches all packets. A single default route represents any network that is not in the routing table.
- Routers commonly use default routes that are either configured locally or learned from another router. The default route is used as the Gateway of Last Resort.
- Default static routes are commonly used when connecting an edge router to a service provider network, or a stub router (a router with only one upstream neighbor router).
- The figure shows a typical default static route scenario.




**IPv4 Default Static Route:** The command syntax for an IPv4 default static route is similar to any other IPv4 static route, except that the network address is **0.0.0.0** and the subnet mask is **0.0.0.0**. The 0.0.0.0 0.0.0.0 in the route will match any network address.

**Note**: An IPv4 default static route is commonly referred to as a quad-zero route.

The basic command syntax for an IPv4 default static route is as follows:
Router(config)# **ip route 0.0.0.0 0.0.0.0 {ip-address | exit-intf}**

**IPv6 Default Static Route:** The command syntax for an IPv6 default static route is similar to any other IPv6 static route, except that the ipv6-prefix/prefix-length is **::/0**, which matches all routes.

The basic command syntax for an IPv6 default static route is as follows:

Router(config)# **ipv6 route ::/0 {ipv6-address | exit-intf}**

### *Configure a Default Static Route*

The example shows an IPv4 default static route configured on R1. With the configuration shown in the example, any packets not matching more specific route entries are forwarded to R2 at 172.16.2.2.
R1(config)# **ip route 0.0.0.0 0.0.0.0 172.16.2.2**

An IPv6 default static route is configured in similar fashion. With this configuration any packets not matching more specific IPv6 route entries are forwarded to R2 at 2001:db8:acad:2::2
R1(config)# **ipv6 route ::/0 2001:db8:acad:2::2**

### *Verify a Default Static Route*

The **show ip route static** command output from R1 displays the contents of the static routes in the routing table. Note the asterisk (*) next to the route with code 'S'. The asterisk indicates that this static route is a candidate default route, which is why it is selected as the Gateway of Last Resort.

Notice that the static default route configuration uses the /0 mask for IPv4 default routes. Remember that the IPv4 subnet mask in a routing table determines how many bits must match between the destination IP address of the packet and the route in the routing table. A /0 mask indicates that none of the bits are required to match. As long as a more specific match does not exist, the default static route matches all packets.

```
R1# show ip route static
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
+ - replicated route, % - next hop override

Gateway of last resort is 172.16.2.2 to network 0.0.0.0

S* 0.0.0.0/0 [1/0] via 172.16.2.2
```

This example shows the **show ipv6 route static** command output to display the contents of the routing table.

Notice that the static default route configuration uses the ::/0 prefix for IPv6 default routes. Remember that the IPv6 prefix-length in a routing table determines how many bits must match between the destination IP address of the packet and the route in the routing table. A ::/0 prefix indicates that none of the bits are required to match. As long as a more specific match does not exist, the default static route matches all packets.

```
R1# show ipv6 route static
IPv6 Routing Table - default - 8 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE - Destination
       NDr - Redirect, RL - RPL, O - OSPF Intra, OI - OSPF Inter
       OE1 - OSPF ext 1, OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1
       ON2 - OSPF NSSA ext 2, la - LISP alt, lr - LISP site-registrations
       ld - LISP dyn-eid, lA - LISP away, le - LISP extranet-policy
       a - Application
S   ::/0 [1/0]
     via 2001:DB8:ACAD:2::2
```

## Configure Floating Static Routes

### *Floating Static Routes*

- Another type of static route is a floating static route. Floating static routes are static routes that are used to provide a backup path to a primary static or dynamic route. The floating static route is only used when the primary route is not available.
- To accomplish this, the floating static route is configured with a higher administrative distance than the primary route. The administrative distance represents the trustworthiness of a route. If multiple paths to the destination exist, the router will choose the path with the lowest administrative distance.
- By default, static routes have an administrative distance of 1, making them preferable to routes learned from dynamic routing protocols.
- The administrative distance of a static route can be increased to make the route less desirable than that of another static route or a route learned through a dynamic routing protocol. In this way, the static route "floats" and is not used when the route with the better administrative distance is active.

### *Configure IPv4 and IPv6 Floating Static Routes*

The commands to configure default and floating IP default routes are as follows:

```
R1(config)# ip route 0.0.0.0 0.0.0.0 172.16.2.2
R1(config)# ip route 0.0.0.0 0.0.0.0 10.10.10.2 5
R1(config)# ipv6 route ::/0 2001:db8:acad:2::2
R1(config)# ipv6 route ::/0 2001:db8:feed:10::2 5
```

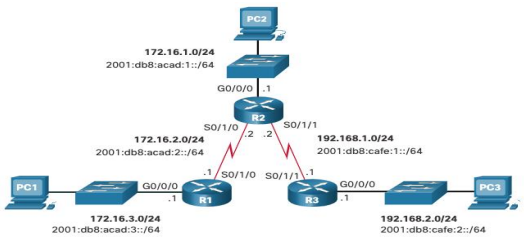
The **show ip route** and **show ipv6 route** output verifies that the default routes to R2 are installed in the routing table. Note that the IPv4 floating static route to R3 is not present in the routing table.

```
R1# show ip route static | begin Gateway
Gateway of last resort is 172.16.2.2 to network 0.0.0.0

S*   0.0.0.0/0 [1/0] via 172.16.2.2
R1# show ipv6 route static | begin S :
S   ::/0 [1/0]
     via 2001:DB8:ACAD:2::2
R1#
```

### *Test the Floating Static Routes*

- What would happen if R2 failed? To simulate this, R2 shuts down both of its serial interfaces.
- R1 automatically generates syslog messages for the link going down.
- A look at R1's routing table would show the secondary route being used.




```
R1# show ip route static | begin Gateway
Gateway of last resort is 10.10.10.2 to network 0.0.0.0
S*    0.0.0.0/0 [5/0] via 10.10.10.2
R1# show ipv6 route static | begin ::
S   ::/0 [5/0]
     via 2001:DB8:FEED:10::2
R1#
```

## Configure Static Host Routes

### *Host Routes*

A host route is an IPv4 address with a 32-bit mask, or an IPv6 address with a 128-bit mask. The following shows the three ways a host route can be added to the routing table:

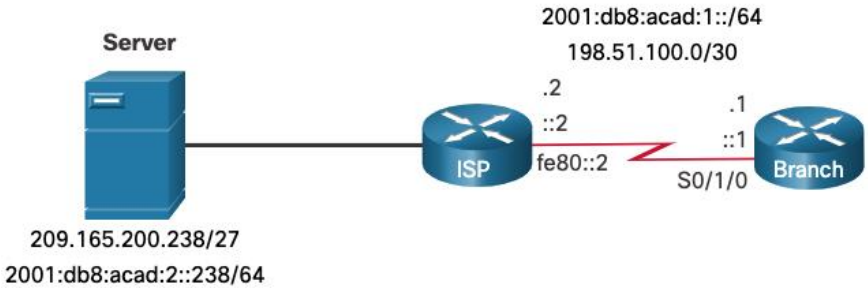
- Automatically installed when an IP address is configured on the router
- Configured as a static host route
- Host route automatically obtained through other methods (discussed in later courses)

### *Automatically Installed Host Routes*

- Cisco IOS automatically installs a host route, also known as a local host route, when an interface address is configured on the router. A host route allows for a more efficient process for packets that are directed to the router itself, rather than for packet forwarding.
- This is in addition to the connected route, designated with a **C** in the routing table for the network address of the interface.
- The local routes are marked with **L** in the output of the routing table.

### *Static Host Routes*

A host route can be a manually configured static route to direct traffic to a specific destination device, such as the server shown in the figure. The static route uses a destination IP address and a 255.255.255.255 (/32) mask for IPv4 host routes, and a /128 prefix length for IPv6 host routes.




### *Configure Static Host Routes*

The example shows the IPv4 and IPv6 static host route configuration on the Branch router to access the server.

```
Branch(config)# ip route 209.165.200.238 255.255.255.255 198.51.100.2
Branch(config)# ipv6 route 2001:db8:acad:2::238/128 2001:db8:acad:1::2
Branch(config)# exit
Branch#
```

### *Verify Static Host Routes*

A review of both the IPv4 and IPv6 route tables verifies that the routes are active.

```
Branch# show ip route | begin Gateway
Gateway of last resort is not set
      198.51.100.0/24 is variably subnetted, 2 subnets, 2 masks
C        198.51.100.0/30 is directly connected, Serial0/1/0
L        198.51.100.1/32 is directly connected, Serial0/1/0
      209.165.200.0/32 is subnetted, 1 subnets
S        209.165.200.238 [1/0] via 198.51.100.2
Branch# show ipv6 route
(Output omitted)
C   2001:DB8:ACAD:1::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:DB8:ACAD:1::1/128 [0/0]
     via Serial0/1/0, receive
S   2001:DB8:ACAD:2::238/128 [1/0]
     via 2001:DB8:ACAD:1::2
Branch#
```

Configure IPv6 Static Host Route with Link-Local Next-Hop

For IPv6 static routes, the next-hop address can be the link-local address of the adjacent router. However, you must specify an interface type and an interface number when using a link-local address as the next hop, as shown in the example. First, the original IPv6 static host route is removed, then a fully specified route configured with the IPv6 address of the server and the IPv6 link-local address of the ISP router.

```
Branch(config)# no ipv6 route 2001:db8:acad:2::238/128 2001:db8:acad:1::2
Branch(config)# ipv6 route 2001:db8:acad:2::238/128 serial 0/1/0 fe80::2
Branch# show ipv6 route | begin ::
C   2001:DB8:ACAD:1::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:DB8:ACAD:1::1/128 [0/0]
     via Serial0/1/0, receive
S   2001:DB8:ACAD:2::238/128 [1/0]
     via FE80::2, Serial0/1/0
Branch#
```

**Laboratory Exercise: Static Routing Configuration**

## Module 16: Troubleshoot Static and Default Routes

***Objectives:***

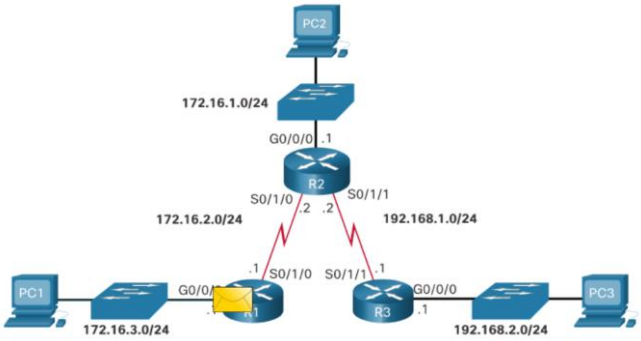
At the end of this module, the student should be able to:

- Explain how a router processes packets when a static route is configured;
- Troubleshoot common static and default route configuration issues.

### Packet Processing with Static Routes

***Static Routes and Packet Forwarding***

- PC1 addresses a packet to PC3 and sends it to the default gateway address.
- When the packet arrives on the R1 G0/0/0 interface, R1 decapsulates the packet and searches the routing table for a matching destination network entry.

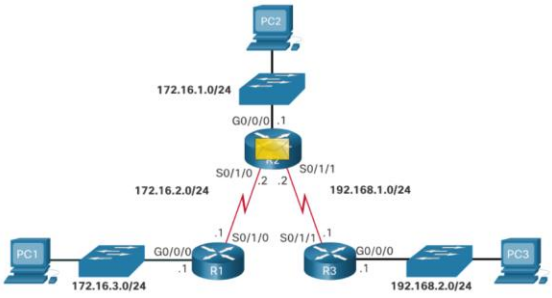



If the destination IP address:

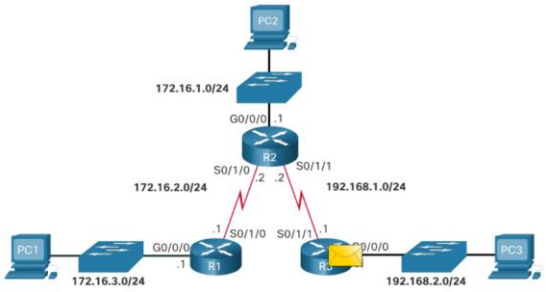
- Matches a static route entry, R1 will use the static route to identify the next-hop IP address or exit interface.
- Does not match a specific route to the destination network, then R1 will use the default static route (if configured).
- Does not match a route table entry, then R1 will drop the packet and send an ICMP message back to the source (i.e., PC1).

Assuming R1 matched a routing table entry, it encapsulates the packet in a new frame and forwards it out of interface S0/1/0 to R2.

- R2 receives the packet on its S0/1/0 interface.
- It decapsulates and processes the packet the same way R1 did.
- When R2 finds a match in the routing table, it uses the identified next-hop IP address or exit interface and sends the packet out of its interface S0/1/1 towards R3.



- R3 receives the packet, decapsulates it, and searches the routing table for a match.
- The destination IP address of PC3 matches the directly connected G0/0/0 interface. Therefore, R3 searches the ARP table for the Layer 2 MAC address of PC3.
- If no ARP entry exists, then R3 sends an ARP request out of the G0/0/0 interface.




- PC3 responds with an ARP reply containing its MAC address.
- R3 encapsulates the packet in a new frame and uses the PC3 MAC address as the destination MAC address and the G0/0/0 MAC address as the source MAC address.
- The frame is forwarded out of interface G0/0/0 and PC3 receives and processes it accordingly.

## Troubleshoot IPv4 Static and Default Route Configuration

### *Network Changes*

Networks fail for a number of reasons:

- An interface can fail
- A service provider drops a connection
- Links can become oversaturated
- An administrator may enter a wrong configuration.

Network administrators are responsible for pinpointing and solving the problem.

To efficiently find and solve these issues, it is advantageous to be intimately familiar with tools to help isolate routing problems quickly.

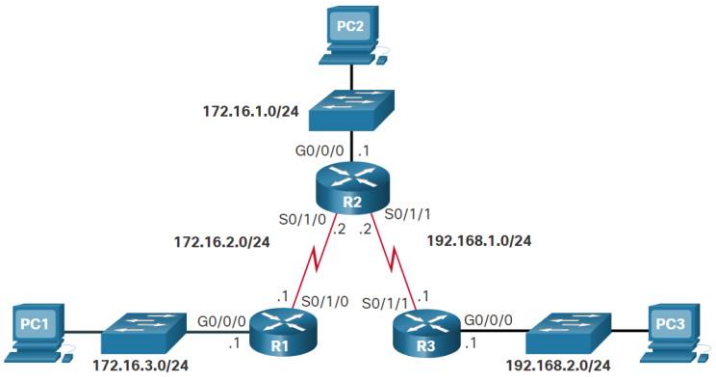
### *Common Troubleshooting Commands*

| Command | Description |
|---|---|
| **ping** | • Verify Layer 3 connectivity to destination.<br>• Extended pings provide additional options. |
| **traceroute** | • Verify path to destination network.<br>• It uses ICMP echo reply messages to determine the hops to the destination. |
| **show ip route** | • Displays the routing table.<br>• Used to verify route entries for destination IP addresses. |
| **show ip interface brief** | • Displays the status of device interfaces.<br>• Used to verify the operational status and IP address of an interface. |
| **show cdp neighbors** | • Displays a list of directly connected Cisco devices.<br>• Also used to validate Layer 1 and 2 connectivity. |

***Solve a Connectivity Problem***

Connectivity from PC1 to PC3 fails.

- Extended pings from the R1 G0/0/0 interface to PC3 fail.
- Pings from R1 (i.e., S0/1/0 interface) to R2 are successful.
- Pings from R1 (i.e., S0/1/0 interface) to R3 are successful.




- R2 routing table reveals the problem and the incorrect static route is removed.
- A new static route solves the problem.
    - **ip route 172.16.3.0 255.255.255.0 172.16.2.1**

```
R2# show ip route | begin Gateway
Gateway of last resort is not set
      172.16.0.0/16 is variably subnetted, 5 subnets, 2 masks
C           172.16.1.0/24 is directly connected,
GigabitEthernet0/0/0
L        172.16.1.1/32 is directly connected,
GigabitEthernet0/0/0
C        172.16.2.0/24 is directly connected, Serial0/1/0
L        172.16.2.2/32 is directly connected, Serial0/1/0
S        172.16.3.0/24 [1/0] via 192.168.1.1
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, Serial0/1/1
L        192.168.1.2/32 is directly connected, Serial0/1/1
S     192.168.2.0/24 [1/0] via 192.168.1.1
R2#
```

**Online Chapter Exam**

**Final Practical Exam**

**Final Examination**