Contents

Exam Review: Practice Hands-On CLI Skills

To do well on sim and simlet questions, you need to be comfortable with many Cisco router and switch commands, and how to use them from a Cisco CLI. As described in the introduction to this book, sim questions require you to decide what configuration commands need to be configured to fix a problem or to complete a working configuration. Simlet questions require you to answer multiple-choice questions by first using the CLI to issue **show** commands to look at the status of routers and switches in a small network.

To be ready for the exam, you need to know the following kinds of information:

CLI navigation: Basic CLI mechanics of moving into and out of user, enable, and configuration modes

Individual configuration: The meaning of the parameters of each configuration command

Feature configuration: The set of configuration commands, both required and optional, for each feature

Verification of configuration: The **show** commands that directly identify the configuration settings

Verification of status: The **show** commands that list current status values and the ability to decide incorrect configuration or other problem causes of less-than-optimal status values

To help remember and review all this knowledge and skill, you can do the tasks listed in the next several pages.

CCNA Exam Topics with CLI Skill Requirements

Wondering about all the topics in CCNA 200-301 that specifically include configuration or verification skills? You can just scan the CCNA 200-301 exam topics. However, Table 20-3 and Table 20-4 summarize the topics for which you could consider practicing your CLI skills. The tables organize the topics into the same order used in the *CCNA 200-301 Official Cert Guides, Volume 1 and 2*, with chapter references.

Table 20-3 Topics with Configuration Skills in CCNA Volume 1

Topic	Volume 1 Chapter	Date You Finished Lab Review
Switch IPv4	6	
Verifying LAN switching	5	
Switch IPv4	6	
Switch passwords	6	
Switch interfaces	7	
VLANs	8	
VLAN trunking	8	
STP and RSTP	10	
Layer 2 EtherChannel	10	
Router interfaces	15	
Router IPv4 addresses and static routes	16	

Topic	Volume 1 Chapter	Date You Finished Lab Review
Router on a Stick	17	
Layer 3 switching with SVIs	17	
Layer 3 switching with routed interfaces and L3 EtherChannels	17	
OSPF fundamentals	20	
OSPF network types	21	
IPv6 addressing on routers	24	
IPv6 static routes	25	

Table 20-4 Topics with Configuration Skills in CCNA Volume 2

Topic	Volume 2 Chapter	Date You Finished Lab Review
Standard ACLs	2	
Extended ACLs	3	
Telnet and SSH Access ACLs	5	
Port Security	6	
DHCP client and DHCP relay	7	
DHCP snooping	8	
Dynamic ARP Inspection	8	
Syslog, NTP, CDP, and LLDP	9	
NAT, PAT	10	

Practice Exam Mindset

1. First and foremost, have the positive mindset of “what can I learn from this?”
2. Read question to get “the big picture” or overview of the scenario.
3. Read question a second time and “make it mine”. Put myself in the scenario, like it’s actually happening to me, a real-world business task that I need to complete.
4. Then before even looking at the multiple-choice options, start to think of “what options I could try to fix or complete this task”.
5. Then look at option A, go through that option as it relates to the scenario, and see if it’s the best solution.
    a. Repeat on all option b, c, d, etc…
6. Eliminate all the illogical answers.
7. Then out of the remaining options, choose the one you think is correct. The one that makes more sense.
8. Check your answer to see if you got it correct. If not, take the time to understand why the correct answer is the best option. Make sure you Connect and Associate the correct answer in your mind with this process.

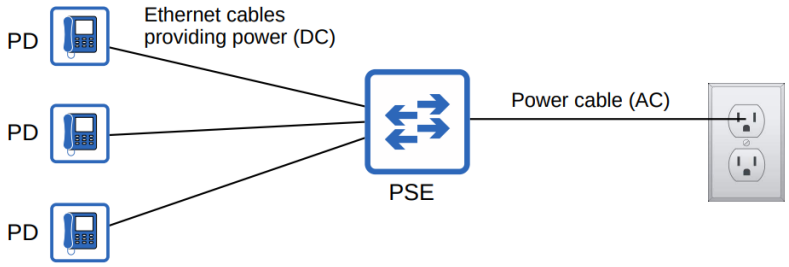
1 Network Fundamentals

1.1.h PoE

PoE allows Power Sourcing Equipment (PSE) to provide power to Powered Devices (PD) over an Ethernet cable.

Typically the PSE is a switch and the PDs are IP phones, IP cameras, wireless access points, etc.

The PSE receives AC power from the outlet, converts it to DC power, and supplies that DC power to the PDs.



Too much electrical current can damage electrical devices.
Power policing can be configured to prevent a PD from taking too much power:

PoE Policing Modes

PoE policing comprises two modes, which determine the action to take on the interface after a port shuts down because of an inline-power policing violation:

- Enable power policing: **(config-if)#power inline police** or **(config-if)#power inline police action errdisable**
- By default, power policy violation will put the port in **err-disable** restarted, or we can configure automatic recovery: **(config)#errdisable recover cause inline-power**
- Change the policy to log the event and restart the port immediately: **(config-if)#power inline police action log**
- Show power policy: **#show power inline police**

1. **Logging** - An error message is logged to the console and the interface restarts; the device powers up:

```
Switch(config-if)# power inline police action log
```

2. **Errdisable** (Default) - In addition to logging an error message to the console, the interface is placed in an errdisable state so that the device attached to the port does not receive inline-power until you restart the port or configure an errdisable autorecovery mechanism:

```
Switch(config-if)# power inline police
Switch(config-if)# power inline police action errdisable
```

Table 13-2 Power over Ethernet Standards

Name	Standard	Watts at PSE	Powered Wire Pairs
Cisco Inline Power	Cisco	7	2
PoE	802.3af	15	2
PoE+	802.3at	30	2
UPoE	802.3bt	60	4
UpoE+	802.3bt	100	4

* **UPoE** and **UPoe+** are only ones that use **4 powered wire pairs**

Question

You are tasked with installing and configuring a new PoE-supported IP camera with a power consumption of 20 W. After connecting it to a PoE-enabled switch, the camera does not turn on. What is the likely cause of the problem?

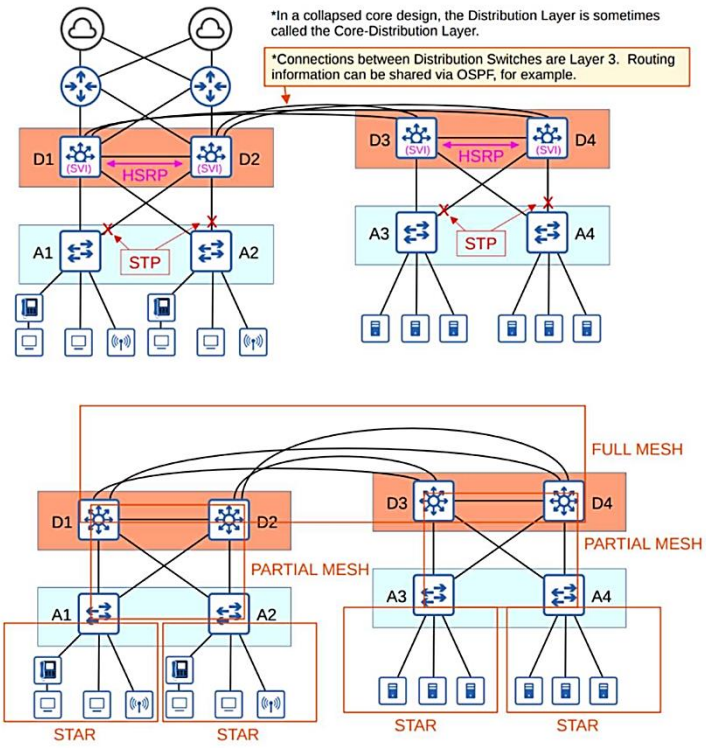
Answer

The correct answer is **The switch does not support the PoE Plus standard.** Normal PoE can only provide up to 15.4 W of power, while PoE Plus provides up to 30 W. Devices that support PoE do not need to be configured to use it, as they will power on when connected to the Ethernet. PoE is enabled on all ports. Supplied PoE power decreases with range, but the drop is minimal.

1.2.a 2,3 tier

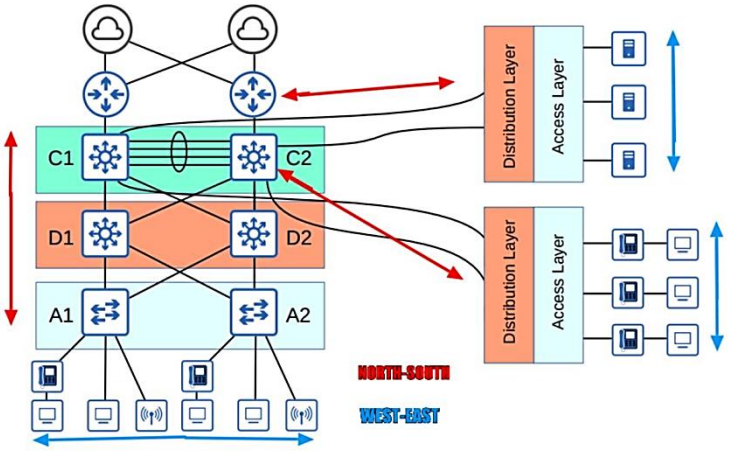
Two-Tier LAN:

- The two-tier LAN design consists of two hierarchical layers:
  - → **Access Layer**
  - → **Distribution Layer**
- Also called a 'Collapsed Core' design because it omits a layer that is found in the Three Tier design: the **Core Layer**
- **Access Layer:**
  - → the layer that end hosts connect to (PCs, printers, cameras, etc.)
  - → typically Access Layer Switches have lots of ports for end hosts to connect to
  - → QoS marking is typically done here
  - → Security services like port security, DAI, etc are typically performed here
  - → switchports might be PoE-enabled for wireless APs, IP phones, etc.
- **Distribution Layer:**
  - → aggregates connections from the Access Layer Switches
  - → typically is the border between Layer 2 and Layer 3
  - → connects to services such as Internet, WAN, etc.



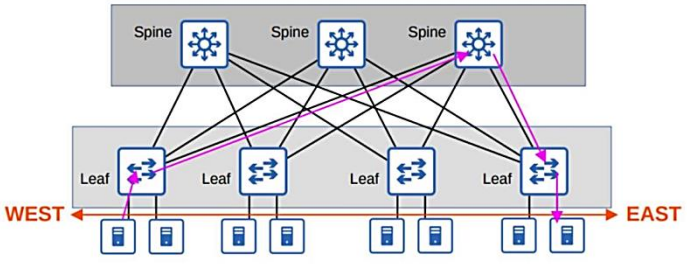
Three-Tier LAN:

- **Access Layer:**
  - → the layer that end hosts connect to (PCs, printers, cameras, etc.)
  - → typically Access Layer Switches have lots of ports for end hosts to connect to
  - → QoS marking is typically done here
  - → Security services like port security, DAI, etc are typically performed here
  - → switchports might be PoE-enabled for wireless APs, IP phones, etc.
- **Distribution Layer: *sometimes called Aggregation Layer**
  - → aggregates connections from the Access Layer Switches
  - → typically is the border between Layer 2 and Layer 3
  - → connects to services such as Internet, WAN, etc. *in a two-tier design
- **Core Layer:**
  - → Connects Distribution Layers together in large LAN networks
  - → The focus is speed ('fast transport')
  - → CPU-intensive operations such as security, QoS marking/classification, etc. should be avoided at this Layer
  - → Connections are all Layer 3. No spanning-tree!
  - → Should maintain connectivity throughout the LAN even if devices fail



Spine-Leaf Architecture:

- There are some rules about Spine-Leaf architecture:
  - → Every Leaf switch is connected to every Spine switch.
  - → Every Spine switch is connected to every Leaf switch.
  - → Leaf switches do not connect to other Leaf switches.
  - → Spine switches do not connect to other Spine switches.
  - → End hosts (servers etc.) only connect to Leaf switches.
- The path taken by traffic is randomly chosen to balance the traffic load among the Spine switches.
- Each server is separated by the same number of 'hops' (except those connected to the same Leaf), consistent latency for East-West traffic.



1.2.c Spine-leaf (aka Clos architecture)

- There are some rules about Spine-Leaf architecture:
  - → Every Leaf switch is connected to every Spine switch.
  - → Every Spine switch is connected to every Leaf switch.
  - → Leaf switches do not connect to other Leaf switches.
  - → Spine switches do not connect to other Spine switches.
  - → End hosts (servers etc.) only connect to Leaf switches.
- The path taken by traffic is randomly chosen to balance the traffic load among the Spine switches.
- Each server is separated by the same number of 'hops' (except those connected to the same Leaf), consistent latency for East-West traffic.

Cisco ACI architecture uses the Spine-leaf architecture.

1.2.d WAN

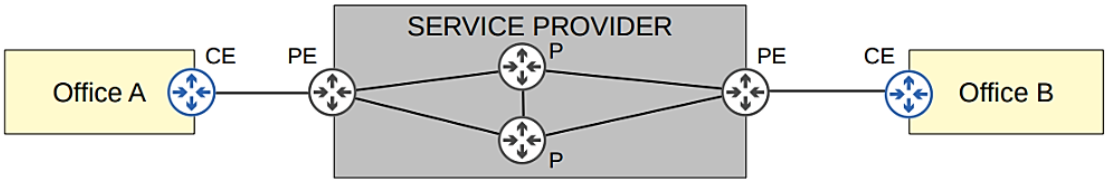
Wide Area Network (WAN) is a network that extends over a large geographic area and connects separate LANs.

The internet could be considered a WAN, WANs are more commonly used to enterprises' private connections between their sites.

There are two main types of WAN connection:

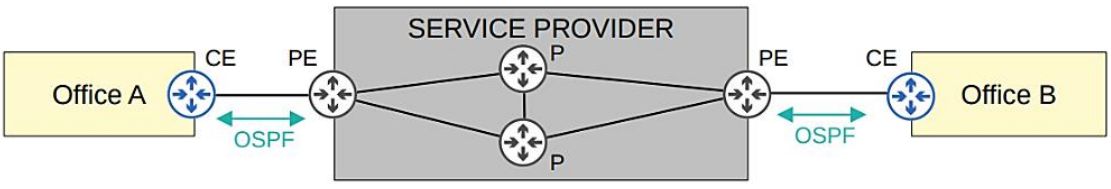
- **Leased Line: Point-to-point** serial link (High-level Data Link Control HDLC, or Point-to-Point PPP encapsulation) between offices over the Service Provider's network. There are standards provide different speeds:
  - In North America: T1 (1.544 Mbps), T2 (6.312 Mbps), T3 (44.736 Mbps).
  - In Europe: E1 (2.048 Mbps), E2 (8.448 Mbps), E3 (34.368 Mbps).

- Multiple Protocol Label Switching (**MPLS**): uses a **shared core infrastructure from the service provider**. Many different technologies can be used to connect to a service provider's MPLS network for WAN service (4G/5G, Cable TV, Serial, Ethernet/Fiber, etc.) When the Provider Edge (PE) router receives frames from the Customer Edge (CE) router, they add a label to the frame and these labels are used to make forwarding decisions over the service provider network, not the destination IP.
  - MPLS stands for 'Multi Protocol Label Switching'.
  - Similar to the Internet, service providers' MPLS networks are shared infrastructure because many customer enterprises connect to and share the same infrastructure to make WAN connections.
  - However, the *label switching* in the name of MPLS allows VPNs to be created over the MPLS infrastructure through the use of **labels**.
  - Some important terms: CE router = Customer Edge router
    PE router = Provider Edge router
    P router = Provider core router
  - When the PE routers receive frames from the CE routers, they add a label to the frame.
  - These labels are used to make forwarding decisions within the service provider network, not the destination IP.



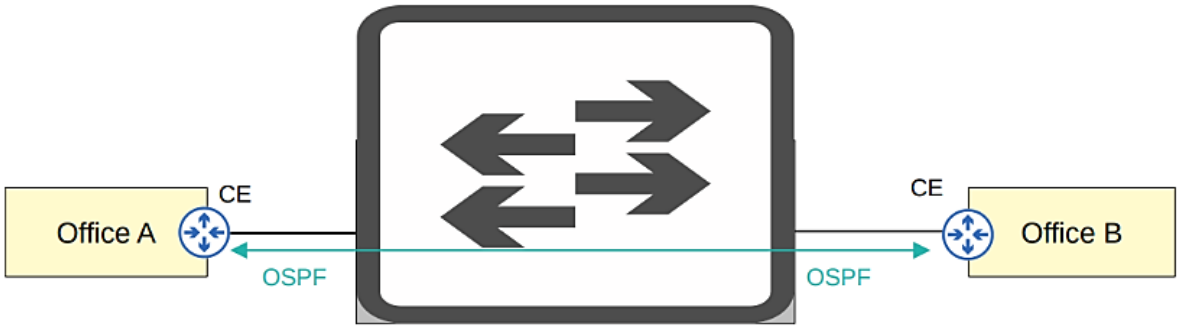
Virtual Private Network (VPN) are often used with MPLS:

- **Layer 3 MPLS VPN**: The CE and PE routers peer using OSPF to share routing information. The provider's core routers are transparent to the customer.
  - The CE routers do not use MPLS, it is only used by the PE/P routers.
  - When using a *Layer 3 MPLS VPN*, the CE and PE routers peer using OSPF, for example, to share routing information.
  - For example, in the diagram below Office A's CE will peer with one PE, and Office B's CE will peer with the other PE.
  - Office A's CE will learn about Office B's routes via this OSPF peering, and Office B's CE will learn about Office A's routes too.

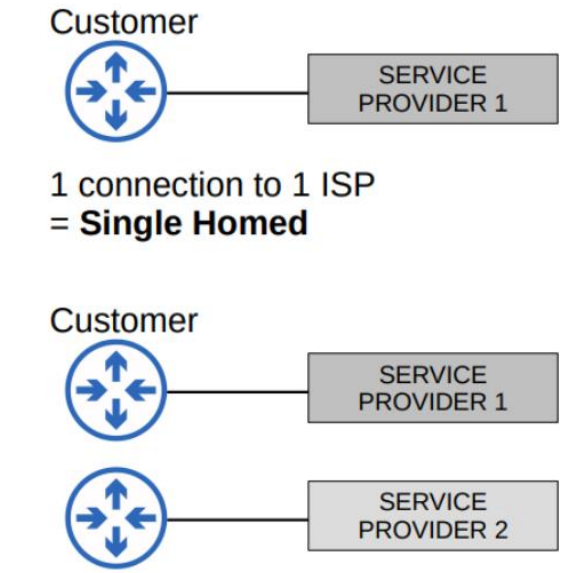


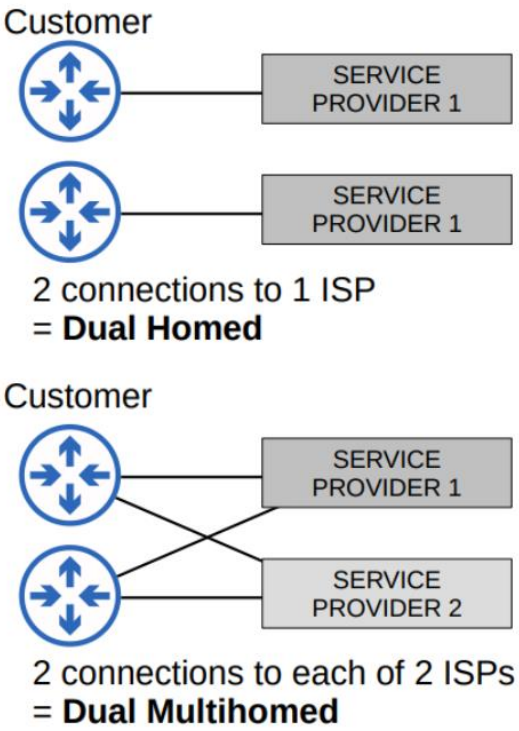
– **Layer 2 MPLS VPN**: The service provider is transparent to the CE routers (all their network acts like a switch).

- When using a *Layer 2 MPLS VPN*, the CE and PE routers do not form peerings.
- The service provider network is entirely *transparent* to the CE routers.
- In effect, it is like the two CE routers are directly connected.
  → Their WAN interfaces will be in the same subnet.
- If a routing protocol is used, the two CE routers will peer directly with each other.



Redundant Internet Connections





Copper UTP Ethernet Standards

SPEED	COMMON NAME	CAT	IEEE	INFORMAL NAME	MAX LENGTH	COPPER PAIRS
10 Mbps	Ethernet	3	802.3	10BASE-T	100 m	2
100 Mbps	Fast Ethernet	5	802.3u	100BASE-T	100 m	2
1 Gbps	Gigabit Ethernet	5e	802.3ab	1000BASE-T	100 m	4
10 Gbps	10 Gig Ethernet	6	802.3an	10GBASE-T	100 m	4

For 10BASE-T and 100BASE-T cables:

- Transmitter (Tx) pins: 1 and 2 on router, firewall, and PC. 3 and 6 on switch.
- Receiver (Rx) pins: 3 and 6 on router, firewall, and PC. 1 and 2 on switch.

Fiber Ethernet Standards

- Fiber: Multimode fiber allows multiple angles (modes) of light to enter the core, allow longer cables than UTP but shorter than single-mode fiber. Single-mode fiber allow only a single mode, more expensive than multimode fiber, but allow longer cables.

Speed	Cable type	IEEE standard	Informal name	Maximum lenth
1 Gbps	Multi/Single mode	802.3z	1000BASE-LX	550 m (multi) / 5 km (single)
10 Gbps	Multimode	802.3ae	10GBASE-SR	400 m
10 Gbps	Single-mode	802.3ae	10GBASE-LR	10 km
10 Gbps	Single-mode	802.3ae	10GBASE-ER	30 km

You are connecting two Catalyst 6500 switches with fiber-optic cable. When you boot SwitchA, you receive a SYS-3-TRANSCEIVER_NOTAPPROVED error.

Which of the following is most likely the cause of the problem? *(Select the best answer.)*

A. You have connected a cable to an incorrect port.

B. You have installed a third-party SFP module.

1.4 Identify Interface/Cable issues

The **show interfaces status** and **show interfaces** commands list both the speed and duplex settings on an interface:

```
S1# show interface status

Port  Name  Status      Vlan    Duplex  Speed  Type
Fa0/1       connected   trunk   full    100    10/100BaseTX
Fa0/2       connected   1       half    100    10/100BaseTX
Fa0/3       connected   1       a-full  a-100  10/100BaseTX
Fa0/4       disabled    1       auto    auto   10/100BaseTX
Fa0/5       disabled    1       auto    auto   10/100BaseTX
Fa0/6       notconnect  1       auto    auto   10/100BaseTX
```

```
SW1# show interfaces fa0/13
! lines omitted for brevity
     Received 284 broadcasts (0 multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 281 multicast, 0 pause input
     0 input packets with dribble condition detected
     95226 packets output, 10849674 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 unknown protocol drops
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 PAUSE output
     0 output buffer failures, 0 output buffers swapped out
```

- **Runts**: Frames that are smaller than the minimum frame size (64 bytes)
- **Giants**: Frames that are larger than the maximum frame size (1518 bytes)
- **CRC**: Frames that failed the CRC check (in the Ethernet FCS trailer)
- **Frame**: Frames that have an incorrect format (due to an error)
- **Input errors**: Total of various counters, such as the above four
- **Output errors**: Frames the switch tried to send, but failed due to an error

Runts: Frames that did not meet the minimum frame size requirement (64 bytes, including the 18-byte destination MAC, source MAC, type, and FCS). Can be caused by collisions.

Giants: Frames that exceed the maximum frame size requirement (1518 bytes, including the 18-byte destination MAC, source MAC, type, and FCS).

Input Errors: A total of many counters, including runts, giants, no buffer, CRC, frame, overrun, and ignored counts.

CRC: Received frames that did not pass the FCS math; can be caused by collisions.

Frame: Received frames that have an illegal format, for example, ending with a partial byte; can be caused by collisions.

Packets Output: Total number of packets (frames) forwarded out the interface.

Output Errors: Total number of packets (frames) that the switch port tried to transmit, but for which some problem occurred.

Collisions: Counter of all collisions that occur when the interface is transmitting a frame.

Late Collisions: The subset of all collisions that happen after the 64th byte of the frame has been transmitted. (In a properly working Ethernet LAN, collisions should occur within the first 64 bytes; late collisions today often point to a duplex mismatch.)

baby giants do not indicate a duplex mismatch on an Ethernet local area network (LAN). A baby giant is an Ethernet frame that is up to 1,600 bytes in length. The default maximum transmission unit (MT U) size for Ethernet frames is 1,500 bytes, not including the Ethernet header and the cyclic redundancy check (CRC) trailer, which add 18 bytes to the frame. Baby giant frames are slightly larger than an Ethernet frame.

Table 29-7 Common LAN Layer 1 Problem Indicators

Type of Problem	Counter Values Indicating This Problem	Common Root Causes
Excessive noise	Many input errors, few collisions	Wrong cable category (Cat5, Cat5E, Cat6), damaged cables, EMI
Collisions	More than roughly 0.1% of all frames are collisions	Duplex mismatch (seen on the half-duplex side), jabber, DoS attack
Late collisions	Increasing late collisions	Collision domain or single cable too long, duplex mismatch

1.5 Compare TCP to UDP

Overall, TCP provides more features than UDP, but at the cost of additional overhead.

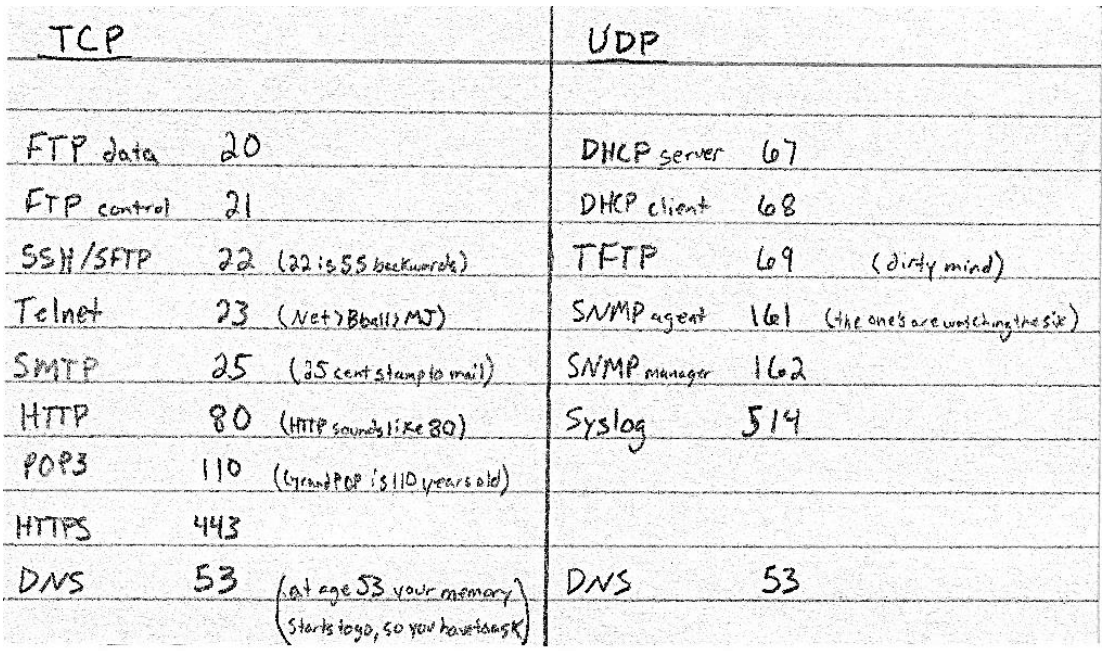
TCP	UDP
Connection-oriented	Connectionless
Reliable	Unreliable
Sequencing	No sequencing
Flow control	No flow control
Used for downloads, file sharing, etc.	Used for VoIP, live video, etc.

Where:

- Connection-oriented: Before actually sending data to the destination host, the two hosts communicate to establish a connection. Once the connection is established, the data exchange begins.
- Reliable: The destination host must acknowledge that it received each TCP segment. If a segment is not acknowledged, it is sent again.
- Sequencing: Sequence numbers in TCP header allow destination hosts to put segments in correct order.
- Flow control: The destination host can tell the source host to increase/decrease the data transmission rate.

TCP	UDP
transport layer	transport layer
connection	connectionless
flow control	no flow control
error recovery	error check / discard
slower	faster
TCP window size	no windowing
guaranteed delivery	best effort
ordered data	unordered data
retransmission	no retransmission
HTTP, Telnet, SSH, FTP	DHCP, SNMP, VoIP, Video

TCP			UDP		
FTP data	20		DHCP server	67	
FTP control	21		DHCP client	68	
SSH	22		TFTP	69	Dirty mind
Telnet	23		SNMP agent	161	The 1's are watching the 6
SMTP	25	25 cents to mail	SNMP manager	162	
HTTP	80	HTTP sounds like 80	Syslog	514	
POP3	110	GrandPOP is 110 years old			
HTTPS	443				
DNS	53	Lose your memory at 53	DNS	53	Lose your memory at 53



Which of the following Application layer protocols use UDP for unsynchronized, connectionless data transfer? *(Select 2 choices.)*

- ☐ A. FTP
- ☑ **B. SNMP**
- ☐ C. SMTP
- ☐ D. HTTP
- ☑ **E. TFTP**

Explanation:

Simple Network Management Protocol (SNMP) and Trivial File Transfer Protocol (TFTP) use User Datagram Protocol (UDP) for unsynchronized, connectionless data transfer. UDP is a Transport layer protocol that does not use sequence numbers or establish synchronized connections. Because of UDP's connectionless nature, transmitted datagrams can appear out of sequence or can be dropped without notice; thus it is the responsibility of the Application layer protocol to reorder packets or request the transmission of lost datagrams. SNMP is used to monitor and manage network devices. TFTP uses UDP port 69 to transfer files unreliably and without authentication over a network. Administrators can use TFTP to transfer Cisco IOS images from a server to a device to perform firmware upgrades. Other common Application layer protocols that use UDP include Dynamic Host Configuration Protocol (DHCP), which is used to assign Internet Protocol (IP) addressing information to clients, Network Time Protocol (NTP), which is used to coordinate time on a network, and Remote Authentication Dial-In User Service (RADIUS), which is used to authenticate users.

1.7 Describe the need for private IPv4 addressing

Binary	Decimal
00000000	0
10000000	128
11000000	192
11100000	224
11110000	240
11111000	248
11111100	252
11111110	254
11111111	255

Class	Leading Bits	Network Bits	Remaining Bits	Number of Networks	Hosts per Network	Default Subnet Mask
Class A	0xxx (1-127)	8	24	128	16,777,214	255.0.0.0
Class B	10xx (128-191)	16	16	16,384	65,534	255.255.0.0
Class C	110x (192-223)	24	8	2,097,152	254	255.255.255.0
Class D (multicast)	1110 (224-239)	Not defined	Not defined	Not defined	Not defined	Not defined
Class E (reserved)	1111 (240-254)	Not defined	Not defined	Not defined	Not defined	Not defined

RFC 1918 Private IPv4 Addresses

IP address range	Number of addresses	Classful description	Largest CIDR block (subnet mask)	Host ID size
10.0.0.0 – 10.255.255.255	16,777,216	single class A	10.0.0.0/8 (255.0.0.0)	24 bits
172.16.0.0 – 172.31.255.255	1,048,576	16 contiguous class Bs	172.16.0.0/12 (255.240.0.0)	20 bits
192.168.0.0 – 192.168.255.255	65,536	256 contiguous class Cs	192.168.0.0/16 (255.255.0.0)	16 bits

RFC 1918 Private IPv4 Addresses
All of 10.x.x.x
Between 172.16.x.x – 172.31.x.x
All of 192.168.x.x

1.8 Configure and verify IPv6 addressing and prefix

Enable IPv6 routing: **(config)#ipv6 unicast-routing**

Setting up IPv6 address on interface: **(config-if)#ipv6 address [IPv6]/[prefix length]**

global unicast	internet routable with global routing prefix 2000::/3
multicast address	prefix FF00::/8 (send to group members)
unique local address	private global address, not internet routable, starts with FD00::/8
link-local address	mandatory, auto-configured, local subnet only, used for routing adjacency, prefix FE80::/10
loopback address	universal address, assigned to every interface, prefix ::1/128
modified eui-64	IPv6 host portion identifier, derived from MAC address
unspecified address	source address for initializing host, :1/128

For IPv6 addressing the only loopback address that exists is ::1

1.9 Describe IPv6 address types

1.9a Unicast (global, unique local and link local)

First understand that: Global = Public, Local = Private

- *Global Unicast* = prefix **2000::** Public address, routable over internet.
  - "In the year 2000, unicorns cast global spells"

- *Link Local* = prefix **fe80::** auto-generated, not routable.
  - "For excellent time, call 1-800-800-8000"

- *Unique Local* = prefix **fd00::** private IP address, not routable

1.9.c Multicast (Fast Fast multicast)

Multicast addresses = prefix **ff00::/8** one-to-many, one source to multiple destinations.
It uses the address range of ff00::/8 (FF00:: to FFFF:FFFF:...).

There are multiple multicast scopes, which determine how far the package should be forwarded:

- *Interface-local (FF01):* The packet does not leave local device.
- *Link-local (FF02):* The packet does not leave local subnet.
- *Site-local (FF05):* The packet does not leave the physical location (not forwarded over WAN).
- *Organization-local (FF08):* The packet does not leave WAN.
- *Global (FF0E):* No boundary, could be routed over the internet.

For link-local addresses (FF02::/16), there are multiple conventions:

Purpose	IPv6	IPv4
All nodes/hosts (broadcast)	FF02::1	224.0.0.1
All routers	FF02::2	224.0.0.2
All OSPF routers	FF02::5	224.0.0.5
All OSPF DRs/BDRs	FF02::6	224.0.0.6
All RIP routers	FF02::9	224.0.0.9
All EIGRP routers	FF02::A	224.0.0.10

- FF02:1 is all hosts on a link, FF02::2 is all routers on a link
- Router Advertisements are sent to FF02::1 and the unicast address of the RS (Router Solicitation) host
- The All-IPv6-routers local-scope multicast address is FF02::2
- EIGRPv6 uses FF02::A as it's MULTICAST address

Which of the following statements about unique local unicast IPv6 addresses are true? *(Select 2 choices.)*

- ☐ **A. The first 7 bits of the prefix are always 1111110.**
- ☑ **B. They are unique only within an organization.**

1.9.d Modified EUI-64

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

HEXADECIMAL CONVERSION

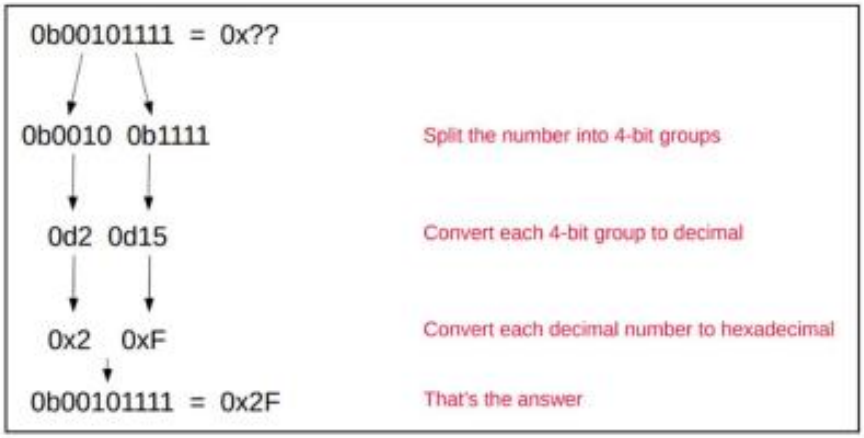
- Binary / Base 2 / 0b
  0, 1

10 ← Is that decimal 10? Or binary 10 (=decimal 2)? Or hexadecimal 10 (=decimal 16)?

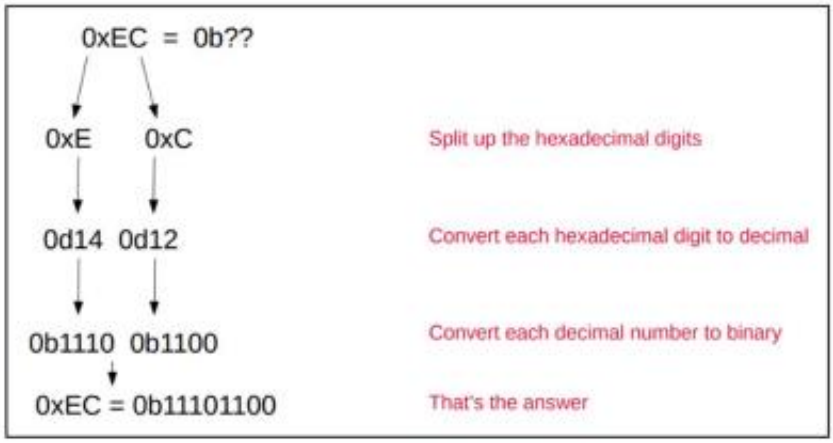
0b10

- Decimal / Base 10 / 0d
  0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Hexadecimal / Base 16 / 0x
  0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

Binary → Hexadecimal



Hexadecimal → Binary



FIND THE EUI-64 IPv6 ADDRESS

R1's G0/1 interface has a MAC address of `0D2A.4FA3.00B1`.
What will G0/1's IPv6 address be after issuing the following command?
R1(config-if)# **ipv6 address 2001:db8:0:1::/64 eui-64**

a) `2001:db8:0:1:0B2A:4FFF:FFA3:B1`

b) `2001:db8:0:1:C2A:4FFF:FEA3:B1`

c) `2001:db8:0:1:0F2A:4FFF:FFA3:B1`

d) `2001:db8:0:1:F2A:4FFF:FEA3:B1`

1. Divide the MAC:

   0D2A 4F | A3 00B1

2. Insert FFEE in middle:

   0D2A 4F**FF FE**A3 00B1

3. Invert 7th bit:

   0**D**2A 4FFF FEA3 00B1 ⟶ 0**F**2A 4FFFF FEA3 00B1

   1101 → 1111 = F (hex)

4. Insert new EUI-64 MAC as the last 64-bits of the IPv6 address:

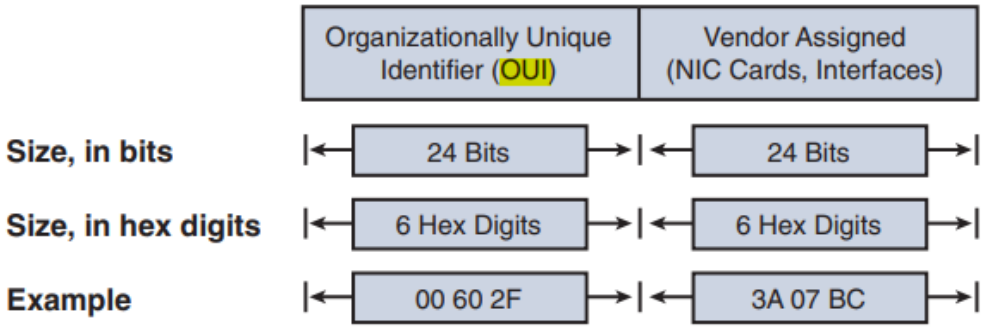
   2001:db8:0:1:: ⟶ 2001:db8:0:1:**F2A:4FFF:FEA3:B1**

When 'inverting the 7th bit':

- First convert the second hex digit (that contains the 7th bit) of the MAC address into binary.
- Then while in binary form, invert the 3rd binary digit (which is the 7th bit of the MAC).
- Then convert the inverted binary back into hex.
- Finally, replace the 2nd digit of the MAC with the new hex digit.

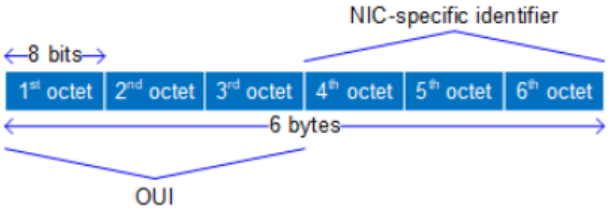
MAC Address

Figure 30-6 Structure of a Unicast Ethernet Address



The first three octets of a Media Access Control (MAC) address represent the organizationally unique identifier (OUI), which is assigned by the Institute of Electrical and Electronics Engineers (IEEE) to identify the manufacturer of the device. The last three octets make up the unique network interface card (NIC)-specific identifier assigned to the device by the manufacturer.

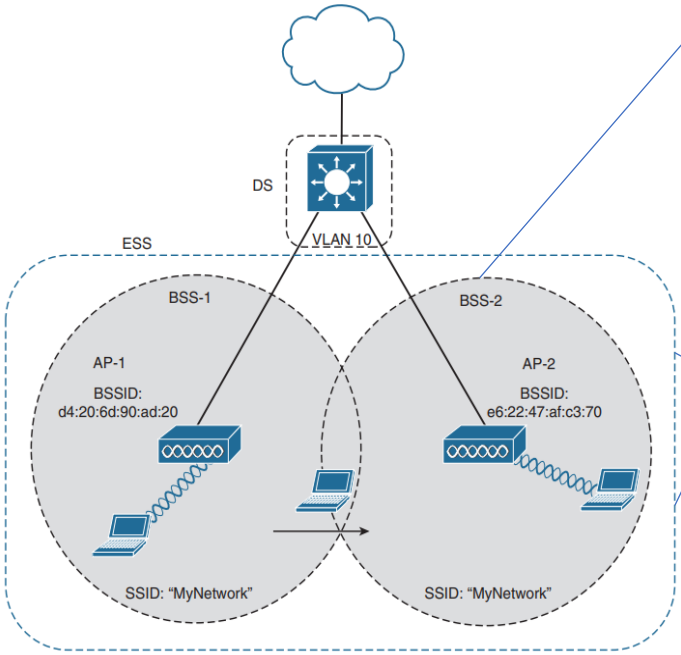
A MAC address, also known as a physical address, is a 48-bit address that is permanently encoded on a NIC. MAC addresses are written in hexadecimal format and are composed of six 8-bit octets for a total of 6 bytes of data in the entire address, as shown in the following diagram:



1.11 Describe wireless principles

- Wireless LANs are defined in 802.11.
- Operate in half duplex using CSMA/CA
- Wireless signals can be affected by **absorption, reflection, refraction, diffraction, and scattering.**
- Various aspects of waves can be measured, such as **amplitude, frequency,** and **period.**
- Frequency is measured in **hertz** (Hz).
- Wireless LANs use two frequency ranges: the **2.4 GHz** band and **5 GHz** band.
  → Wi-Fi 6 (802.11ax) can use the **6 GHz** range too.
- Bands are divided into channels.
- 5 GHz band consists of non-overlapping channels.
- 2.4 GHz band channels overlap. To avoid overlapping, use channels 1, 6, and 11 (in North America).
- 802.11 standards (802.11b, 802.11a, etc) and their frequencies/theoretical max data rates.
- Service sets are groups of wireless devices. Three types:
  → Independent (**IBSS**, also called **ad hoc**)
  → Infrastructure (**BSS**, **ESS**)
  *passing between APs in an ESS is called **roaming**.
  → Mesh (**MBSS**)
- Service sets are identified by an **SSID** (non-unique, human-readable) and **BSSID** (unique, MAC address of AP)
- The area around an AP where its signal is usable is called a **BSA**.
- The upstream wired network is called the **DS**.
- When multiple WLANs are used, each is mapped to a separate VLAN on the wired network.
- APs can also operate as a **repeater**, **workgroup bridge**, or **outdoor bridge**.

Standard	Frequencies	Max Data Rate (theoretical)	Alternate Name
802.11	2.4 GHz	2 Mbps	
802.11b	2.4 GHz	11 Mbps	
802.11a	5 GHz	54 Mbps	
802.11g	2.4 GHz	54 Mbps	
802.11n	2.4 / 5 GHz	600 Mbps	'Wi-Fi 4' (HT)
802.11ac	5 GHz	6.93 Gbps	'Wi-Fi 5'(VHT)
802.11ax	2.4 / 5 / 6 GHz	4*802.11ac	Wi-Fi 6'



BSS (Basic Service Set) is a kind of Infrastructure Service Set in which clients connect to each other via an AP, but not directly to each other.

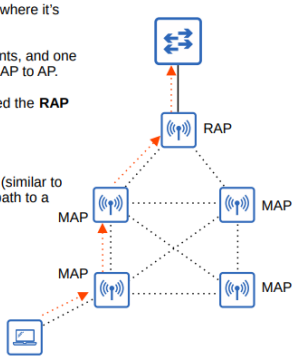
- A **BSSID** (**Basic Service Set ID**) is used to uniquely identify the AP.
  → Other APs can use the same SSID, but not the same BSSID
  → The BSSID is the MAC address of the AP's radio
- Wireless devices request to *associate* with the BSS.
- Wireless devices that have associated with the BSS are called 'clients' or 'stations'.
- *The area around an AP where its signal is usable is called a **BSA** (**Basic Service Area**).
- *Clients must communicate via the AP, not directly with each other.

ESS (Extended Service Set)

- To create larger wireless LANs beyond the range of a single AP, we use an **ESS** (**Extended Service Set**).
- APs with their own BSSs are connected by a wired network.
  → Each BSS uses the same SSID.
  → Each BSS has a unique BSSID.
  → Each BSS uses a different channel to avoid interference.
- Clients can pass between APs without having to reconnect, providing a seamless Wi-Fi experience when moving between APs.
  → This is called **roaming**.
- The BSAs should overlap about 10-15%.

MBSS (Mesh Basic Service Set)

- An **MBSS** (**Mesh Basic Service Set**) can be used in situations where it's difficult to run an Ethernet connection to every AP.
- Mesh APs use two radios: one to provide a BSS to wireless clients, and one to form a 'backhaul network' which is used to bridge traffic from AP to AP.
- At least one AP is connected to the wired network, and it is called the **RAP** (**Root Access Point**).
- The other APs are called **MAPs** (**Mesh Access Points**).
- A protocol is used to determine the best path through the mesh (similar to how dynamic routing protocols are used to determine the best path to a destination).

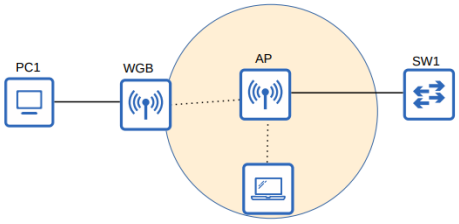


Workgroup bridge (WGB)

- A **workgroup bridge** (WGB) operates as a wireless client of another AP, and can be used to connect wired devices to the wireless network.
- In the example below, PC1 does not have wireless capabilities, and also does not have access to a wired connection to SW1.
- PC1 has a wired connection to the WGB, which has a wireless connection to the AP.

There are two kinds of WGBs:
Universal WGB (uWGB) is an 802.11 standard that allows one device to be bridged to the wireless network.

WGB is a Cisco-proprietary version of the 802.11 standard that allows multiple wired clients to be bridged to the wireless network.



An AP in repeater mode can be used to extend the range of a BSS.

The repeater will simply retransmit any signal it receives from the AP

- Single radio repeater must operate on same channel as the AP, but can drastically reduce the overall throughput on the Ch by 50%.
- Two radio repeater can receive on one channel, and then retransmit on another channel.

- An **outdoor bridge** can be used to connect networks over long distances without a physical cable connecting them.
- The APs will use specialized antennas that focus most of the signal power in one direction, which allows the wireless connection to be made over longer distances than normally possible.
- The connection can be point-to-point as in the diagram below, or point-to-multipoint in which multiple sites connect to one central site.

Standard	Frequency	Data Rate	Mnemonic	WiFi
802.11	2.4 ghz	2 mbps		
802.11b	2.4 ghz	11 mbps	**b**	
802.11a	5 ghz	54 mbps	**A**	
802.11g	2.4 ghz	54 mbps	**g**	
802.11n	2.4/5 ghz	600 mbps	**nN**	4
802.11ac	5 ghz	6.93 gbps		5
802.11ax	2.4/5/6 ghz	4 x 802.11ac		6

Mnemonic Key

b	A	g	nN	lowercase = 2.4 ghz UPPERCASE = 5 ghz
11	54		600	mbps

So just remember Bilbo Baggin's address:

Bilbo **bAgnN**
1154 600th st

What percentage of wireless coverage overlap is considered appropriate to ensure that wireless clients do not lose connectivity when roaming from one AP to another? *(Select the best answer.)*

A. 10 to 15 percent

1.12 Explain virtualization fundamentals (server virtualization, containers, and VRFs)

Modern servers are strong, using one physical server to host one application server is inefficient. Therefore virtualization is used.

Virtualization breaks the one-to-one relationship between the hardware and OS, allowing multiple OS's to run on a single physical server. Each divided instance is called a Virtual Machine (VM).

A hypervisor is used to manage and allocate the hardware resources to each VM. There are 3 main types of hypervisor:

- Type 1 hypervisor: Run directly on the hardware, also called bare-metal hypervisor. Examples are *VMware ESXi, Microsoft Hyper-V*. Often used in data centers.
- Type 2 hypervisor: Run as a program on an OS, also called hosted hypervisor. The OS run on the hardware is called Host OS, the OS run on VM is called Guest OS. Examples are *VMware Workstation, Oracle Virtualbox*. Often used on personal devices.

  Benefits of virtualization are:

  - Partitioning: Run multiple OS on a physical machine with allocated system resources.
  - Isolation: Provide fault and security isolation.
  - Encapsulation: An entire VM could be save as a file.
  - Hardware independence: Independence to the type of hardware.

  VMs are connected to each other and to the external network via a virtual switch running on the hypervisor. Virtual switch could (but not must) use VLAN to seperate VMs at layer 2.

A **virtual machine monitor (VMM)** is another name for a hypervisor

VRFs

VRFs create multiple virtual routers inside a single router or Layer 3 switch. The router configuration associates interfaces and routing protocol neighbors to VRFs, with a separate routing table per VRF.

Overlapping subnets causes problems for a router (or Layer 3 switch) when using traditional conventions. VRFs solve this problem

VRF Summary of Critical Points:

- The VRF must be created via configuration in each device that performs routing (router or Layer 3 switch).
- Each router will have a separate IP routing table for each VRF, each holding routes for only that VRF.
- The configuration assigns each interface to a VRF so that the router places the associated connected route into that VRF's routing table.
- The routing protocol configuration defines VRFs to associate neighbor relationships and VRFs. Routes learned by a given routing protocol neighbor result in new routes in that VRF's routing table.
- The router keeps its original routing table, called the global routing table. The global routing table holds routes related to interfaces and routing protocol neighbors not associated with any VRF.

Overlapping subnets causes problems for a router (or Layer 3 switch) when using traditional conventions. Working through some of the key points:

- Typically, one router has one IP routing table.
- Typically, a router allows only one interface connected to the same subnet.
- If an engineer attempts to connect a second interface to the same subnet, the router will not bring up the interface.
- Data Center virtualization software can locate and move VMs and reprogram networking, so VMs from multiple customers can exist on one physical server—creating a case of overlapping subnets within that physical server.

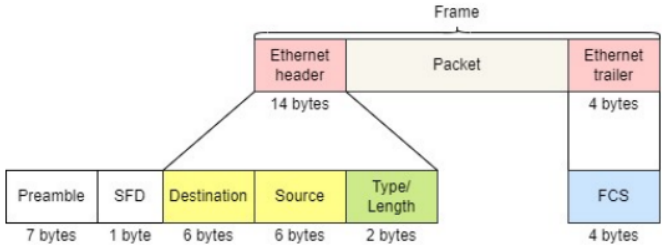
1.13 Describe switching concepts

1.13.b Frame switching

Frame switching is the activity of a switch to read the Frame header and send the frame to appropriate ports based on its destination.

ETHERNET HEADER AND TRAILER

Figure 6.2 The contents of the Ethernet header and trailer. They are split up into multiple fields, each serving a different purpose. The fields of the header are the Destination, Source, and Type/Length. The trailer consists of a single field: the Frame Check Sequence (FCS). Although not considered part of the Ethernet frame, the Preamble and Start Frame Delimiter (SFD) are sent with each frame.



= **26 bytes** (header + trailer)

Please **S**top **D**ropping **S**ushi, **T**hey're **L**ovely **D**elicate **P**ure and **F**resh

Ethernet Field	PREAMBLE	SFD (START FRAME DELIMITER)	DESTINATION	SOURCE	TYPE/LENGTH (ONLY ONE)	DATA/PAD	FCS (FRAME CHECK SEQUENCE)
Mnemonic	**P**lease	**S**top	**D**ropping	**S**ushi	**T**hey're **L**ovely	**D**elicate **P**ure	**F**resh
Bytes	7	1	6	6	2	46-1500	4
Bits	56	8	48	48	16		32
Keywords	synchronize		MAC	MAC	IPv4 = 0x0800 (hex) IPv6 = 0x86DD (hex)		CRC

Let's break it down:

- **Preamble**: The initial sequence of *alternating 1s and 0s* (10101010) in an Ethernet frame.
  Allows devices to synchronize their receiver clocks.
- **SFD (Start Frame Delimiter)**: *Marks the end of the preamble and indicates the start of the frame.*
  Breaks the Preamble sequence of 1s and 0s by putting a 1 at end (10101011).
- **Destination**: Represents the MAC address of the intended recipient of the frame.
- **Source**: Indicates the MAC address of the sender of the frame.
- **Type**: Specifies the protocol type or EtherType of the frame (*value of 1536 or greater indicates Type*).
- **Length**: Indicates the length of the frame payload (*value of 1500 or less indicates Length*).
- **Data**: The actual data or payload of the frame (L3 network packet)
- **Pad**: Padding used to ensure the frame meets the minimum length requirements.
- **FCS** (Frame Check Sequence): A checksum value used for error detection. Uses *Cyclic Redundancy Check (CRC)*.

By using the mnemonic **P**lease **S**top **D**ropping **S**ushi, **T**hey're **L**ovely **D**elicate **P**ure and **F**resh, you can remember the order of these fields in an Ethernet frame.

- **Preamble** field of an Ethernet frame provides receiver clock synchronization.
- **SFD** field marks the end of the preamble, and the beginning of the rest of the frame
- **FCS** field detects corrupted data by running a 'CRC' algorithm over the received data
- MAC address is 48 bits (first 24bits is OUI)

A switch uses the Source MAC Address field to populate its MAC address table. It associates the source MAC address with the interface on which the frame was received. This allows the switch to learn how to reach other devices on the network.

If a switch doesn't know how to reach a destination, it uses an **Unknown Unicast frame** which is flooded out all interfaces except the one it was received on.

A switch will use destination Media Access Control (MAC) addresses to make forwarding decisions. Switches make forwarding decisions based on the destination MAC address contained in a frame's header. The switch first searches the Content Addressable Memory (CAM) table for an entry that matches the frame's destination MAC address. The CAM table, which is also called the switching table, is used by a switch to discover the relationship between the Layer 2 address of a device and the physical port used to reach the device. If the frame's destination MAC address is not found in the table, the switch forwards the frame to all its ports, except the port from which it received the frame. If the destination MAC address is found in the table, the switch forwards the frame to the appropriate port. The source MAC address is also recorded if it did not previously exist in the CAM table.

1.13.a MAC learning and aging

ARP (Address Resolution Protocol) is used to discover the MAC address (L2 address) of a known IP address (L3 address). ARP consists of two messages:

- **ARP Request**: broadcast to the network (to the address FFFF.FFFF.FFFF).
- **ARP Reply**: unicast to the requested host.

By default, learned MAC will age to conserve memory. The default value is **300 seconds (5 minutes),** value of 0 disables aging

1.13.d MAC address table

- Consists of Vlan, Mac Address, Type (Static/Dynamic) and Ports.
- Show by: **#show mac address-table**
- Clear dynamic MAC addresses by: **#clear mac address-table dynamic**
- Clear a specific dynamic MAC address:
  #clear mac address-table dynamic address [mac-address]
  or **#clear mac address-table dynamic interface [interface-id]**
- Manually adding MAC address: **(config)#mac address-table static [mac-address] vlan [vlan-id] interface [interface-id]**
- Config MAC aging time: **(config)#mac address-table aging-time [time-in-seconds] (vlan [vlan-id])**

The Content Addressable Memory (CAM) table is used by a switch to discover the relationship between the Open Systems Interconnection (OSI) Layer 2 address of a device and the physical port used to reach the device. Switches make forwarding decisions based on the destination MAC address contained in a frame's header. The switch first searches the CAM table for an entry that matches the frame's destination MAC address. If the frame's destination MAC address is not found in the table, the switch forwards the frame to all its ports, except the port from which it received the frame. If the destination MAC address is found in the table, the switch forwards the frame to the appropriate port. The source MAC address is also recorded if it did not previously exist in the CAM table.

MAC addr = IP multicast

The Media Access Control (MAC) address **01-00-5E-0F-0F-0F** represents an Internet Protocol (IP) multicast address. The Ethernet multicast range of **01-00-5E-00-00-00** through **01-00-5E-7F-FF-FF** has been allocated for IP multicast use.

2 Network Access

2.1.a Access ports (data and voice)

- Force a port into access mode (i.e. do not autonegotiate, do not accept trunking):
  (config-if)#switchport mode access
- Associate a port to a VLAN:
  (config-if)#switchport access vlan [vlan-id]
- Add a voice VLAN to an interface:
  (config-if)#switchport voice vlan [vlan-id]
  Although the interface sends/receives traffic from two VLANs in this case, the port is still considered an access port.

A **voice port** is considered an **access port**.

A voice port needs to be in a separate vlan.

Normal/Extended VLANs

VLAN Range	Description
VLAN 1-1005	normal VLAN range
VLAN 1006-4094	extended VLAN range
VLAN 1, 1002-1005	auto-created / can't delete
VLAN 1006-4094	no pruning from trunk

2.2.a Trunk ports

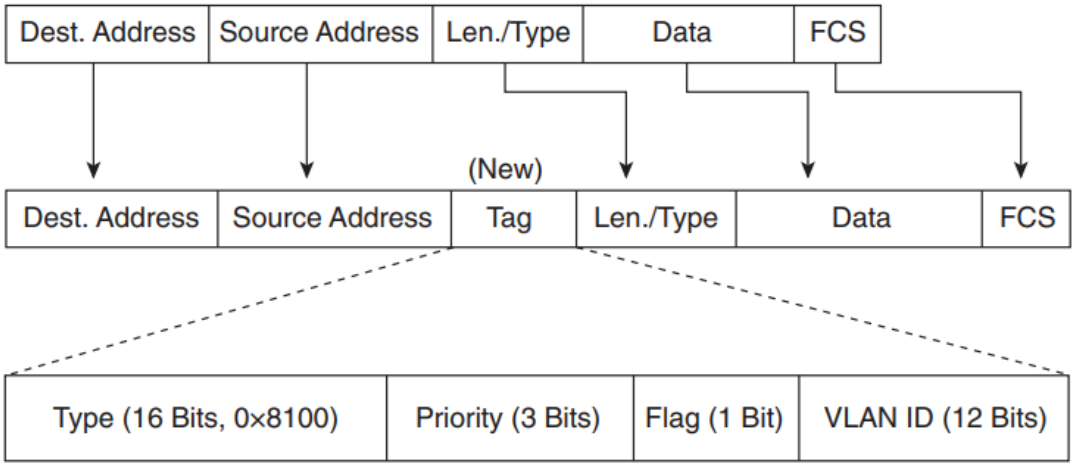
Dynamic Trunking Protocol (DTP)

DTP Mode	Result
auto - auto (default)	access port
auto – desirable	trunk negotiated
desirable – desirable	trunk negotiated
nonegotiate	access port

*static trunking (DTP turned off)

2.2.b 802.1Q

Figure 26-3 Fields of the 802.1Q Tag Inside an Ethernet Frame



- Normal Ethernet header:
  Preamble – STD – Destination – Source – Type
- Ethernet header with 802.1q tag:
  Preamble – STD – Destination – Source – **802.1q tag** – Type

801.1q tag format is TPID + TCI, or in more detail:
TPID (16 bits) – PCP (3 bits) – DEI (1 bit) – VID (12 bits)
Where:

- TPID (Tag Protocol ID): always set to 0x8100 to indicate 802.1q.
- PCP (Priority Code Point): used cor Class of Service (CoS), which give priority to important traffic in congested networks.
- DEI (Drop Eligible Indicator): indicate if the frame can be dropped in congested networks.
- VID (VLAN ID): indicate the VLAN ID.

Since VID has 12 bits, we can have up to 4094 VLANs, which are divided into:

- Normal VLANs: 1-1005
- Extended VLANs: 1006-4094, only supported in modern switches.

802.1Q tag format

16 bits	3 bits	1 bit	12 bits
TPID	TCI		
	PCP	DEI	VID

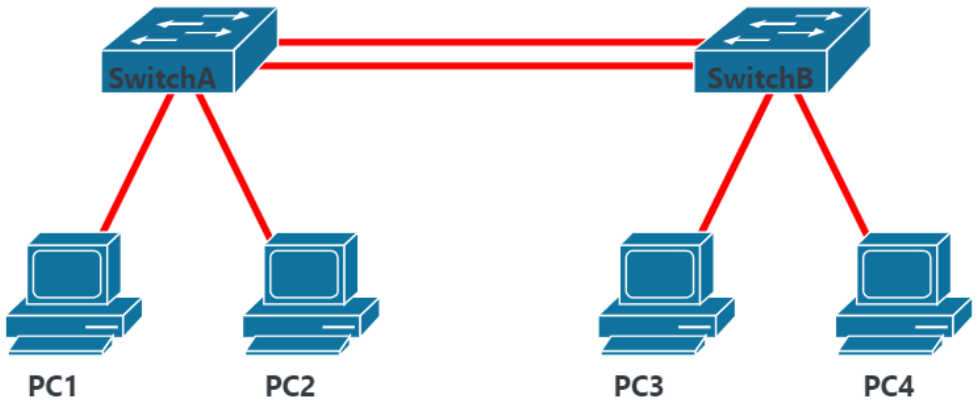
Server	Client	Transparent
creates/modifies/deletes VLANs	synchronizes VTP information	creates/modifies/deletes VLANs
synchronizes VTP information	originates VTP advertisements	forwards VTP advertisements
originates VTP advertisements	forwards VTP advertisements	stores VLAN information in NVRAM
forwards VTP advertisements		
stores VLAN information in NVRAM		

VTP client mode One of three VTP operational modes for a switch with which switches learn about VLAN numbers and names from other switches, but which does not allow the switch to be directly configured with VLAN information.

VTP server mode One of three VTP operational modes. Switches in server mode can configure VLANs, tell other switches about the changes, and learn about VLAN changes from other switches.

VTP transparent mode One of three VTP operational modes. Switches in transparent mode can configure VLANs, but they do not tell other switches about the changes, and they do not learn about VLAN changes from other switches.

BOSON LAB – VTP



Your company's network includes two Cisco switches, SwitchA and SwitchB.

SwitchB has been configured to participate in a VTP domain named **boson** with the password **NetSimX**. SwitchB has also been configured with the correct trunk and EtherChannel settings.

SwitchA is being repurposed from a previous installation and must be reconfigured.

Configure SwitchA with the following parameters:

- Configure SwitchA to participate in the same VTP domain as SwitchB, and ensure that changes can be made to the VLAN configuration of SwitchA directly from the command line.
- Configure switch ports FastEthernet 0/1 and FastEthernet 0/2 to use a Cisco-proprietary EtherChannel negotiation protocol.
- Configure switch ports FastEthernet 0/1 and FastEthernet 0/2 as members of EtherChannel port-group 1, and ensure that they actively negotiate EtherChannel links.
- Configure the EtherChannel virtual interface to establish a trunk link with SwitchB, and ensure that the link uses a standards-based encapsulation protocol.
- Ensure that DTP is disabled on the EtherChannel virtual interface.

Incorrect

Answer:

Using the console, you should connect to SwitchA and execute the following commands:

```
SwitchA>enable
SwitchA#configure terminal
SwitchA(config)#vtp mode server
SwitchA(config)#vtp domain boson
SwitchA(config)#vtp password NetSimX
SwitchA(config)#interface range fastethernet 0/1 - 2
SwitchA(config-if-range)#channel-protocol pagp
SwitchA(config-if-range)#channel-group 1 mode desirable
SwitchA(config-if-range)#exit
SwitchA(config)#interface port-channel 1
SwitchA(config-if)#switchport trunk encapsulation dot1q
SwitchA(config-if)#switchport mode trunk
SwitchA(config-if)#switchport nonegotiate
SwitchA(config-if)#end
```

2.3 Configure and verify Layer 2 discovery protocols (Cisco Discovery Protocol and LLDP)

CDP	LLDP
has a 60-second update frequency	has a 30-second update frequency
has a 180-second hold timer	has a 120-second hold timer
multicast 0100.0CCC.CCCC	multicast 0180.C200.000E
is enabled by default	is disabled by default
is a Layer 2 protocol	is a Layer 2 protocol
is a proprietary protocol	is an open-standard protocol
can convey VTP information	

Cisco Discovery Protocol (CDP)

CDP is a Layer 2 Cisco proprietary neighbor discovery protocol. Cisco IP phone appears to CDP as a unique neighbor device with an IP address. During bootup, the IP phone receives voice VLAN configuration from the access switch port.

- Cisco-proprietary, enabled on Cisco devices by default.
- CDP messages are periodically sent to multicast MAC address **0100.0CCC.CCCC**.
- By default, CDP messages are sent once every **60 seconds**, and the holdtime of CDP neighbor table is **180 seconds**.
- Related commands:
    * Enable/disable CDP globally: **(config)#(no) cdp run**
    * Enable/disable CDP on an interface: **(config-if)#(no) cdp enable**

* Set CDP send timer: **(config)#cdp timer [time-in-second]**
* Set CDP hold time: **(config)#cdp holdtime [time-in-second]**
* Show cdp configuration (timers, version): **#show cdp**
* Show how many CDP messages have been sent/received: **#show cdp traffic**
* Display which interfaces CDP is enabled on: **#show cdp interface**
* List basic info (without IP addresses) on all neighbors, or neighbors on a given interface: **#show cdp neighbors ([interface-name])**
* Show detailed information on all neighbors: **show cdp neighbors detail**
* Display info of one specific neighbor: **#show cdp entry ([name-of-neighbor])**

CDP can assist in network discovery and troubleshooting. CDP advertises the following helpful information:

- **Device identifier:** Typically the host name
- **Address list:** Network and data-link addresses
- **Port identifier:** The interface on the remote router or switch on the other end of the link that sent the CDP advertisement
- **Capabilities list:** Information on what type of device it is (for example, a router or a switch)
- **Platform:** The model and OS level running on the device

Table 9-3 show cdp Commands That List Information About Neighbors

Command	Description
show cdp neighbors [*type number*]	Lists one summary line of information about each neighbor or just the neighbor found on a specific interface if an interface was listed
show cdp neighbors detail	Lists one large set (approximately 15 lines) of information, one set for every neighbor
show cdp entry *name*	Lists the same information as the **show cdp neighbors detail** command, but only for the named neighbor (case sensitive)

With IP Phones, CDP messages containing *device voice VLAN ID* info, is sent **from the Switch** to the IP Phone. Not the other way around. The IP phone doesn't send the VLAN ID to the switch.

LLDP

- Industry standard (IEEE 802.1AB).
- Has same syntax to CDP, but with **lldp replacing cdp** with some differences:
  * Enable transmission on an interface: **(config-if)#lldp transmit**
  * Enable receipt on an interface: **(config-if)#lldp receive**
  * With LLDP, **Tx** (transmission) /**Rx** (reception) can be controlled independently on each interface with the cmds above **lldp [transmit / receive]**
- By default, LLDP timer is **30 seconds** and the holdtime is **120 seconds**.
- LLDP messages are sent to MAC address **0180.C200.000E**

Note that when reading LLDP/CDP output, "Local interface" means the interface on the host, while "Port ID" means the interface on the neighbor device.

Use the **no lldp run** command to disable LLDP globally, just like **no cdp run**

To disable lldp or cdp on an interface use **no [lldp/cdp] enable**

To restore LLDP hold timer back to default use **no lldp holdtime**

LLDP shows the following about neighbors:

- Management address
  - the IP address used to access the device for management (configuring and verifying the device)
- System capabilities
  - different hardware and software specifications of the device, OS
- System name
  - the host name that was configured on that device
- LLDP can be used to learn the OS version of a neighboring device.

However, the LLDP output in the example does differ from CDP in a few important ways:

- LLDP uses **B** as the capability code for switching, referring to **bridge**, a term for the device type that existed before switches that performed the same basic functions.
- LLDP does not identify IGMP as a capability, while CDP does (**I**).
- CDP lists the neighbor's **platform**, a code that defines the device type, while LLDP does not.
- LLDP lists capabilities with different conventions (see upcoming Example 9-19).

2.4 Configure and verify (Layer 2/Layer 3) EtherChannel (LACP)

LACP	PAgP
open standard (multivendor)	Cisco proprietary
bundle = 8 ports + 8 standby	bundle = 8 ports
passive mode (default)	auto mode (default)
active mode	desirable mode
any port active mode = etherchannel	any port desirable mode = etherchannel

Question:

If LAG is enabled on a WLC that contains eight distribution system ports. All eight distribution system ports are connected to a single switch that is correctly configured to unconditionally bundle its ports. If seven of the eight links fail, what will happen with the traffic?

Answer:

The WLC will pass all wireless client traffic to the switch. The Cisco wireless LAN controller (WLC) will pass all wireless client traffic to the switch, even though seven of the eight links in the link aggregation (LAG) bundle have failed in this scenario. LAG enables multiple distribution system ports on a WLC to operate as one logical group when connected to a switch. Thus, LAG enables load balancing across links between devices and redundancy. If one link fails, the other links in the LAG bundle will continue to function. By default, LAG is enabled on all distribution system ports when it is enabled. However, LAG requires only one functional physical port in order to pass wireless client traffic to the switch.

Guidelines for Configuring EtherChannel

- PAgP is Cisco proprietary.
- LACP is defined in 802.3ad.
- You can combine from two to eight parallel links.
- All ports must be identical:
  - Same speed and duplex
  - Cannot mix Fast Ethernet and Gigabit Ethernet
  - Cannot mix PAgP and LACP
  - Must all be VLAN trunk or nontrunk operational status
- All links must be either Layer 2 or Layer 3 in a single channel group.
- To create a channel in PAgP, sides must be set to
  - Auto-Desirable
  - Desirable-Desirable
- To create a channel in LACP, sides must be set to
  - Active-Active
  - Active-Passive
- To create a channel without using PAgP or LACP, sides must be set to On-On.
- Do *not* configure a GigaStack gigabit interface converter (GBIC) as part of an EtherChannel.
- An interface that is already configured to be a Switched Port Analyzer (SPAN) destination port will not join an EtherChannel group until SPAN is disabled.
- Do *not* configure a secure port as part of an EtherChannel.
- Interfaces with different native VLANs cannot form an EtherChannel.
- When using trunk links, ensure all trunks are in the same mode—Inter-Switch Link (ISL) or dot1q.

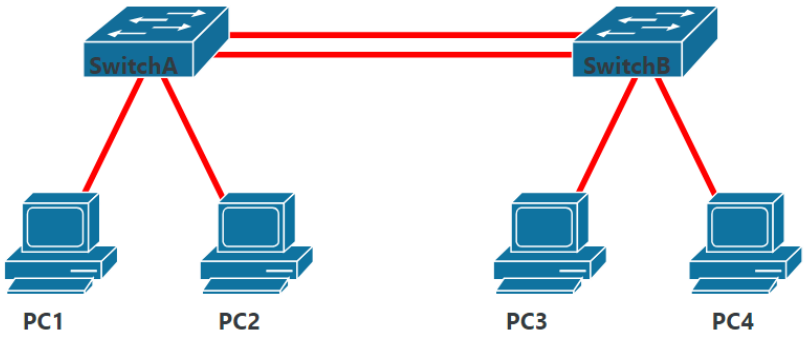
Configuring Layer 2 EtherChannel

`Switch(config)#interface range fastethernet 0/1 - 4`	Moves to interface range configuration mode.
`Switch(config-if-range)#channel-protocol pagp`	Specifies the PAgP protocol to be used in this channel.
Or	
`Switch(config-if-range)#channel-protocol lacp`	Specifies the LACP protocol to be used in this channel.
`Switch(config-if-range)#channel-group 1 mode {desirable | auto | on | passive | active }`	Creates channel group 1 and assigns interfaces 01–04 as part of it. Use whichever mode is necessary, depending on your choice of protocol.

The following table displays the channel-group configurations that will establish an EtherChannel:

SwitchA \ SwitchB	off	auto	desirable	passive	active	on
off	NO	NO	NO	NO	NO	NO
auto	NO	NO	PAgP	NO	NO	NO
desirable	NO	PAgP	PAgP	NO	NO	NO
passive	NO	NO	NO	NO	LACP	NO
active	NO	NO	NO	LACP	LACP	NO
on	NO	NO	NO	NO	NO	ON

Boson ExSIM PBQ (EtherChannel)



Your company's network includes two Cisco switches, SwitchA and SwitchB. SwitchB has been configured with the correct VLAN, trunk, and EtherChannel settings.

Configure SwitchA with the following parameters:

- Switch ports FastEthernet 0/3, FastEthernet 0/4, FastEthernet 0/8, and FastEthernet 0/9 should be explicitly configured as access ports.
- VLAN 10 should contain switch ports FastEthernet 0/3 and FastEthernet 0/4.
- VLAN 20 should contain switch ports FastEthernet 0/8 and FastEthernet 0/9.
- Switch ports FastEthernet 0/1 and FastEthernet 0/2 should be configured to use a standards-based EtherChannel negotiation protocol.
- Switch ports FastEthernet 0/1 and FastEthernet 0/2 should be configured as members of EtherChannel port-group 1 and should actively negotiate EtherChannel links.
- The EtherChannel virtual interface should be configured to use a standards-based encapsulation protocol and to always function as a trunk.

Answer:

Using the console, you should connect to SwitchA and execute the following commands:

```
SwitchA>enable
SwitchA#configure terminal
SwitchA(config)#interface range fastethernet 0/3 - 4
SwitchA(config-if-range)#switchport mode access
SwitchA(config-if-range)#switchport access vlan 10
SwitchA(config-if-range)#exit
SwitchA(config)#interface range fastethernet 0/8 - 9
SwitchA(config-if-range)#switchport mode access
SwitchA(config-if-range)#switchport access vlan 20
SwitchA(config-if-range)#exit
SwitchA(config)#interface range fastethernet 0/1 - 2
SwitchA(config-if-range)#channel-protocol lacp
SwitchA(config-if-range)#channel-group 1 mode active
SwitchA(config-if-range)#exit
SwitchA(config)#interface port-channel 1
SwitchA(config-if)#switchport trunk encapsulation dot1q
SwitchA(config-if)#switchport mode trunk
SwitchA(config-if)#end
```

Explanation:
A virtual LAN (VLAN) is used to create two or more separate, logical LANs on the same switch. By default, all switch ports are members of VLAN 1, which is called the default VLAN. A switch port can be manually assigned to another VLAN; however, it can only be a member of one VLAN at a time. Therefore, assigning a switch port to a VLAN other than the default VLAN removes the switch port from the default VLAN.

Before a switch port can operate in a designated VLAN, it must be in access mode. To place a switch port into access mode, you should issue the **switchport mode access** command in interface configuration mode. A switch port can be assigned a VLAN membership if the port is not in access mode; however, the VLAN membership will not apply to the port until the port is placed into access mode. To configure a switch port to operate within a particular VLAN, you should issue the **switchport access vlan** *vlan-id* command. This command uses the *vlan-id* keyword to specify the VLAN membership that is assigned to the switch port. For example, to configure switch port FastEthernet 0/3 to operate in VLAN 10, you should issue the following commands:

```
SwitchA(config)#interface fastethernet 0/3
SwitchA(config-if)#switchport mode access
SwitchA(config-if)#switchport access vlan 10
```

EtherChannel is used to bundle two or more identical, physical interfaces into a single logical link between networking devices, such as Cisco Catalyst switches. An EtherChannel can be permanently established between switches, or it can be negotiated by using one of two aggregation protocols: the Cisco-proprietary Port Aggregation Protocol (PAgP) or the open-standard Institute of Electrical and Electronics Engineers (IEEE) 802.3ad protocol, which is also known as Link Aggregation Control Protocol (LACP). To configure a switch port to use a particular aggregation protocol, you should use the **channel-protocol** {**lacp** | **pagp**} command. This command uses the **lacp** and **pagp** keywords to specify the aggregation protocol that is used by the switch port to dynamically establish an EtherChannel. For example, to configure switch port FastEthernet 0/1 to use LACP, you should issue the following commands:

```
SwitchA(config)#interface fastethernet 0/1
SwitchA(config-if)#channel-protocol lacp
```

An EtherChannel can have up to eight active switch ports in the bundle that forms the logical link between switches. Every switch port in the bundle, which is also referred to as a channel group, must be configured with the same speed and duplex settings. To configure a switch port to be a member of a particular channel group, you should issue the **channel-group** *number* **mode** {**active** | **auto** | **desirable** | **on** | **passive**} command. This command uses the *number* keyword to specify a particular channel group. The supported values for the *number* keyword vary depending on hardware platform and IOS revision; however, the values typically range between 1 and 256 with a maximum of 48 channel groups. The **on** keyword is used to configure a switch port as a permanent member of an EtherChannel link, the **active** and **passive** keywords are LACP parameters, and the **auto** and **desirable** keywords are PAgP parameters. The following table describes the function of each of these parameters:

Protocol	Mode	Description
PAgP	Auto	Places a port into a passive negotiating state in which the port responds to PAgP packets it receives but does not initiate PAgP packet negotiation.
PAgP	Desirable	Places a port into an active negotiating state in which the port initiates negotiations with other ports by sending PAgP packets.
PAgP LACP	On	Enables EtherChannel only.
LACP	Active	Enables LACP unconditionally.
LACP	Passive	Enables LACP only if an LACP device is detected.

For example, to configure switch port FastEthernet 0/1 as a member of EtherChannel port-group 1 and to configure it to actively negotiate an EtherChannel link, you should issue the following command from interface configuration mode:

```
SwitchA(config-if)#channel-group 1 mode active
```

When a channel group is configured on a switch, a corresponding Layer 2 EtherChannel virtual interface is automatically created. The virtual interface number corresponds to the number assigned to the channel group. For example, the **channel-group 1 mode active** command shown above would cause SwitchA to create a Layer 2 EtherChannel virtual interface named Port-Channel 1, if it did not already exist. You can configure a Layer 2 EtherChannel interface in the same way that you would configure a physical switch port. Any configuration changes made to the virtual interface are applied to the physical interfaces in the channel group as well. For example, if you were to change the encapsulation method used by the virtual interface, the encapsulation method of every physical interface would also be changed. However, changes made to individual members of a channel group are not replicated to the corresponding virtual interface or to other members of the channel group.

A trunk port allows traffic from multiple VLANs to travel across a single Ethernet link. A trunk port uses an encapsulation method to include VLAN information in each frame that is transmitted out the interface. Cisco supports two types of trunk encapsulation: the Cisco-proprietary Inter-Switch Link (ISL) protocol and the IEEE 802.1Q standard protocol. In this scenario, you are required to configure the EtherChannel virtual interface to use a standards-based encapsulation protocol and to always function as a trunk. You should issue the **switchport trunk encapsulation dot1q** command in interface configuration mode to configure a standards-based encapsulation protocol. Then you should issue the **switchport mode trunk** command to place the EtherChannel virtual interface into trunk mode. For example, to configure the Port-Channel 1 interface to operate in trunk mode and to use 802.1Q encapsulation, you should issue the following commands:

```
SwitchA(config)#interface port-channel 1
SwitchA(config-if)#switchport trunk encapsulation dot1q
SwitchA(config-if)#switchport mode trunk
```

You can verify the configuration of SwitchA by testing end-to-end connectivity with the **ping** command from any of the PCs on the network. For example, to test the connectivity between PC1 and PC3, you should issue the **ping 192.168.100.2** command from PC1. PC1 and PC3 should be able to communicate because they are in the same VLAN; however, PC1 should not be able to communicate with PC2 or PC4, because they are in a different VLAN than PC1.

Although you are not required to do so in this simulation, you should always save the configuration by issuing the copy running-config startup-config command. If you make changes to the running configuration and do not save it to the startup configuration by issuing the copy running-config startup-config command, you will lose those changes if the router is restarted.

2.5 (STP) Interpret basic operations of Rapid PVST+ Spanning Tree Protocol

Spanning Tree Protocol (STP) is standardized in IEEE 802.1D. STP prevents layer 2 loops by placing redundant ports in a blocking state.

Interfaces in the blocking state only send or receive Bridge Protocol Data Units (BPDUs). No frame forwarding.

STP-enabled switches send/receive Hello BPDUs out of all interfaces every 2 seconds. Switches receiving BPDUs will know about the other switch's existence.

Comparison of STP

a. **STP (802.1D)** is the legacy standard that provides a loop free topology in a network with redundant links.
b. **PVST+ (Cisco Standard)** is an enhancement on STP that provides a separate 802.1D spanning tree instance for each VLAN configured in the network (*enabled by default*).
c. **MSTP (802.1s)** is an IEEE standard that is inspired by the earlier Cisco Proprietary MISTP implementation. MSTP maps multiple VLANS into the same spanning tree instance.
d. **RSTP (802.1w)** is an evolution of STP that provides faster convergence of STP.
e. **Rapid PVST+** is an enhancement of RSTP that uses PVST+. Rapid PVST+ provides a separate instance of 802.1w per vlan.

Protocol	Standard	Resources Needed	Convergence	Number of Trees
STP	802.1D	Low	Slow	One
PVST+	Cisco	High	Slow	One for every VLAN
RSTP	802.1w	Medium	Fast	One
Rapid PVST+	Cisco	Very high	Fast	One for every VLAN
MSTP	802.1s	Medium or high	Fast	One for multiple VLANs

Table 10-2 STP Standards and Configuration Options

Name	Based on STP or RSTP?	# Trees	Original IEEE Standard	Config Parameter
STP	STP	1 (CST)	802.1D	N/A
PVST+	STP	1/VLAN	802.1D	**pvst**
RSTP	RSTP	1 (CST)	802.1w	N/A
Rapid PVST+	RSTP	1/VLAN	802.1w	**rapid-pvst**
MSTP	RSTP	1 or more*	802.1s	**mst**

* MSTP allows the definition of as many instances (multiple spanning tree instances, or MSTIs) as chosen by the network designer but does not require one per VLAN.

We select the protocol in use by: **(config)#spanning-tree mode [mst/rapid-pvst/pvst]**

In classic STP, a switch waits for 10 hello interval (20 seconds) to consider a neighbor is lost, but in RSTP, that time is reduced to 3 hello intervals (6 seconds). RSTP distinguishes between 3 link types:

- Edge: A port connected to an end host, moves directly to forwarding without negotiation (similar to Portfast enabled).
- Point-to-point: Link between two switches.
- Shared: A connection to a hub, operate in half-duplex mode.

2.5.a Root port, root bridge (primary/secondary), and other port names

Switches use the Bridge ID field in the BPDU to elect a root bridge for the network.
Bridge ID consists of: Bridge Priority (16 bits) + MAC Address (48 bits).

In Per VLAN Spanning Tree (PVST), the Bridge Priority is divided into:
Bridge Priority (4 bits) + Extended System ID (VLAN ID, 12 bits).
The default bridge priority is 32769 ($2^{15} + 1$).

Only root switches send BPDUs, however every devices will assume it is the root bridge when powering up and only give up its position when receive a BPDU with lower bridge ID.

Root Switch selection:

1. Switch with **lowest bridge ID** (priority + MAC address) becomes primary root bridge.
2. Switch with second lowest bridge ID becomes secondary root bridge.

The default priority is 32,768. The root bridge is the bridge with the lowest BID. Therefore, if the default priority value is not changed, the switch with the lowest MAC address becomes the root.

The root priority could be manually configured by:

(config)#spanning-tree vlan [vlan-id] priority [priority-number] Note that priority number must be a multiple of 4096.

or **(config)#spanning-tree vlan [vlan-id] root primary**

or **(config)#spanning-tree vlan [vlan-id] root secondary**

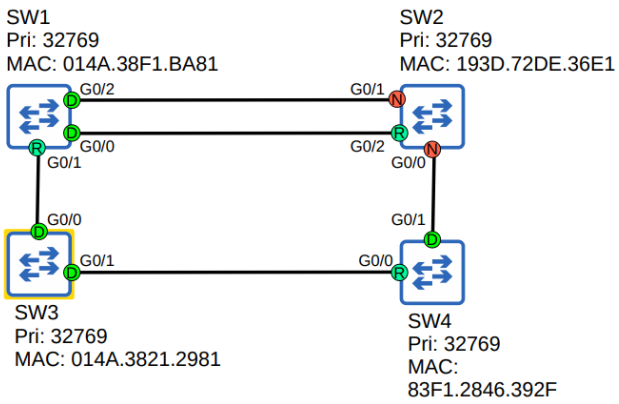
Each non-root switch will select one of its interfaces to be the root port, other ports are designated ports:

Root port selection:

1. Lowest root cost
2. Lowest neighbor bridge ID
3. Lowest neighbor port ID

Root cost is the sum of all port costs on the way from the root bridge to the port in question.
Port costs are determined by the actual link speed by default:

Speed	STP cost	RSTP cost
10 Mbps	100	2,000,000
100 Mbps	19	200,000
1 Gbps	4	20,000
10 Gbps	2	2000
100 Gbps	n/a	200
1 Tbps	n/a	20



Port cost could be manually configured by:

(config-if)# spanning-tree (vlan [vlan-id]) cost [cost]

If the vlan id is not given, the config will be applied to all vlan on that port.

Each remaining collision domain will select one interface to be the designated port, the other port will be non-designated (blocking in STP, discarding in RSTP) ports.

Designated port selection:

1. Interface on switch with lowest root cost.
2. Interface with lowest bridge ID.
3. Lowest neighbor port ID

In RSTP+, the non-designated ports role is split into two:

Alternate port: the backup to root port, if the root port fails, the switch immediately move its best Alternate port for forwarding.

Backup port: the backup for a designated port, not found in modern network since hubs are not used anymore.

2.5.b Port states (forwarding/blocking)

In **classic STP**, there are 4 port states:

- Forwarding: Send and receives all traffic, learn MAC address from frames.
- Blocking: Do not send any traffic, only receive BPDUs, do not learn MAC address.
- Listening: Send and receive only BPDUs, do not learn MAC address.
- Listening: Send and receive only BPDUs, learn MAC address.

In **RSTP**, the states are simplified:

- Discarding: The same as blocking in classic STP. Only receive BPDUs.
- Learning: The same as in classic STP.
- Forwarding: The same as in classic STP.

2.5.c PortFast and STP Toolkit (features)

Portfast allows a port to move immediately to the Forwarding state, bypassing Listening and Learning (eliminates a *DHCP timeout*).

If used, it must be enabled only on ports connected to end hosts (*Edge Ports*). If enabled on a port connected to another switch it could cause a Layer 2 loop.

Configure portfast on one port:
(config-if)#spanning-tree portfast

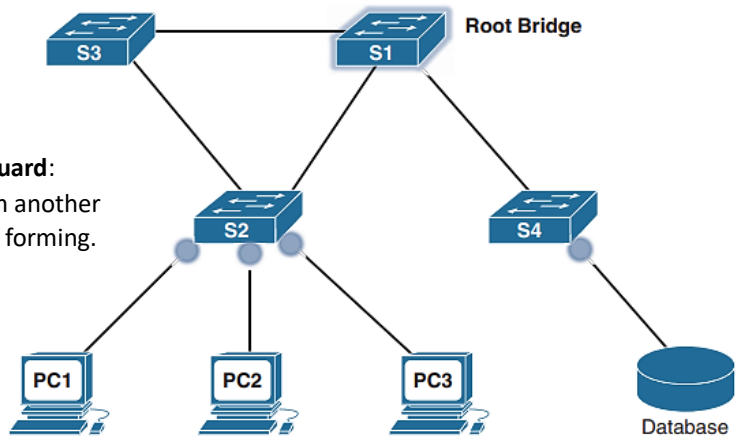
Configure portfast on all access ports:
(config)#spanning-tree portfast default

Interfaces w/ portfast enabled should be configed w/ **BDPU Guard**:
If an interface with BPDU Guard enabled receives a BPDU from another switch, the interface will be shut down to prevent a loop from forming.
It is enabled on all portfast-enabled interfaces by:
(config)#spanning-tree portfast bpduguard default

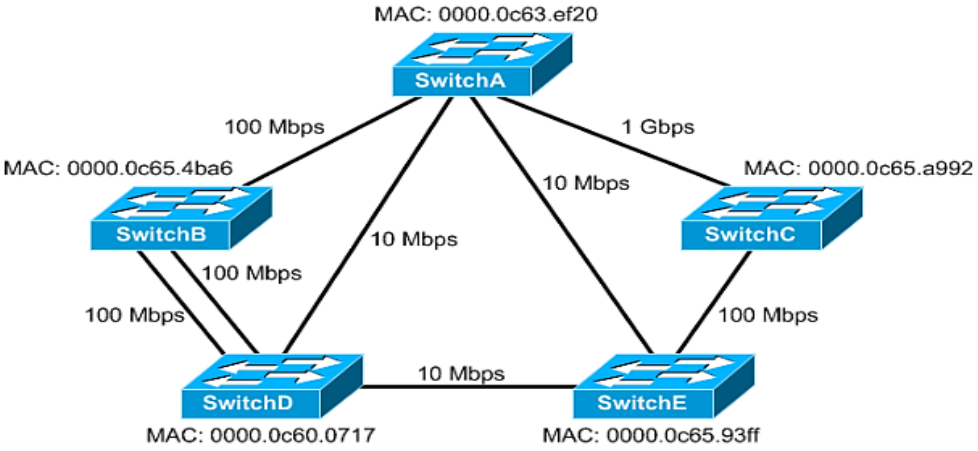
PortFast eliminates a DHCP timeout.

Root guard is used to prevent newly introduced switches from being elected as the new root switch.
This allows administrators to maintain control over which switch is the root.

Figure 25-7 Edge Ports in RSTP



Issue the **spanning-tree guard root** command on the switch's switch port that you are trying to block from becoming the root switch.



You recently became the administrator of the network shown above. You are attempting to determine the roles of each of the switches in the STP topology. You issue the **show spanning-tree** command on SwitchA and receive the following truncated output:

```
SwitchA#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    8192
             Address     0000.0c65.93ff
             Cost        23
             Port        1 (GigabitEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
             Address     0000.0c63.ef20
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300
<output omitted>
```

Which of the following switches is the root bridge? *(Select the best answer.)*

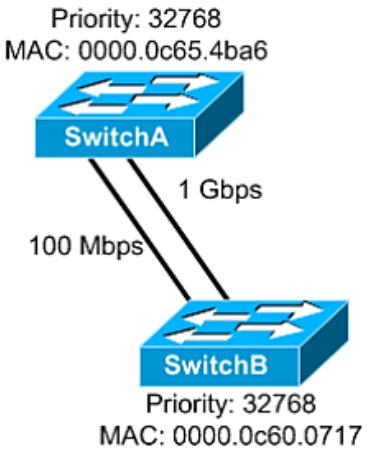
Explanation:
SwitchE is the root bridge. The **show spanning-tree** command displays the root ID of the root bridge and the bridge ID (BID) for the current switch. The Media Access Control (MAC) address listed under root ID is 0000.0c65.93ff, the MAC address for SwitchE; therefore, SwitchE must be the root bridge. Issuing the **show spanning-tree** command on SwitchE would show that the root ID and BID are the same. In addition, issuing the **show spanning-tree detail** command on SwitchE would return output containing the line `We are the root of the spanning tree`.

Electing a root bridge is one of the first steps in the Spanning Tree Protocol (STP) loop-prevention process. STP also optimizes the network by finding the lowest cost paths, determines which port becomes the root port of each switch, and determines which ports become designated ports. STP determines port states through the use of the Spanning Tree Algorithm (STA). The STA uses the following procedure to determine port state:

- A root switch is elected, and each active port on the root switch is placed into the forwarding state.
- Each nonroot switch places the port with the lowest-cost path to the root switch, which is known as the root port, into the forwarding state.
- Ports with the lowest cost to nonroot switches, which are known as designated ports, are also placed into the forwarding state.
- All other ports are placed into the blocking state.

SwitchA is not the root bridge. If the MAC address listed under `Root ID` were the same as SwitchA's MAC address listed under `Bridge ID`, SwitchA would have been the root bridge. In order for SwitchA to become the root bridge, the priority of SwitchA, which is 32769, would have to be lower than the priority of SwitchE, which is 8192. The bridge priority can be set by issuing the **spanning-tree priority** *value* command, where *value* is a multiple of 4096 in the range from 0 through 61440.

SwitchB, SwitchC, and SwitchD are not the root bridge, because their MAC addresses are not listed as the address for the root ID. In order for any of these switches to become the root bridge, the priority would have to be set to a value lower than 8192.



You need to determine how traffic will flow between the two switches in the topology shown above. No VLANs are configured on the switches.

Which port will be blocked? *(Select the best answer.)*

- A. The GigabitEthernet port on SwitchA will be blocked.
- **B. The FastEthernet port on SwitchA will be blocked.**
- C. The FastEthernet port on SwitchB will be blocked.
- D. The GigabitEthernet port on SwitchB will be blocked.
- E. No ports will be blocked.

1. **ROOT BRIDGE ELECTION (RB):**
    a. Root bridge selection criteria:
        i. Lowest bridge ID = aka Lowest Priority (default 32768)
        ii. Lowest MAC Address

    If default priority not changed, switch with the lowest MAC address becomes the root.

    Port State = Forwarding

    Rule #1 - *All ports on root bridge are ALWAYS designated ports (DP) (forwarding state)*

2. **ROOT PORT ELECTION (RP):** Each remaining switch must select ONE of its interfaces to be its root port.
    a. Root port selection criteria:
        i. Lowest root path cost to the root bridge
        ii. Lowest neighbor bridge ID
        iii. Lowest neighbor port ID (uses STP Port ID of Neighbor Port not it's local port)

    Port State = Forwarding

    Rule #2 - Ports across from the root port are ALWAYS designated ports (DP).

    Note: STP Port ID = port priority (default 128) + port number

3. **DESIGNATED PORT (DP) and NON-DESIGNATED (ND) PORT DETERMINATION:**
    Each remaining collision domain will select ONE interface to be a designated port (forwarding).
    The other port in the collision domain will be labeled as non-designated (blocking)
    a. Designated port selection criteria:
        i. Interface on switch with the lowest root path cost to the root bridge
        ii. Interface on switch with lowest bridge ID
    b. Non-Designated port selection:
        i. There isn't an election process like the previous three, it's simply the other/last remaining port in the collision domain.

    (DP) Port State = Forwarding
    (ND) Port State = Blocking
    Rule #3 - All connections with PCs are DPs (forwarding) because PCs don't participate in spanning tree.

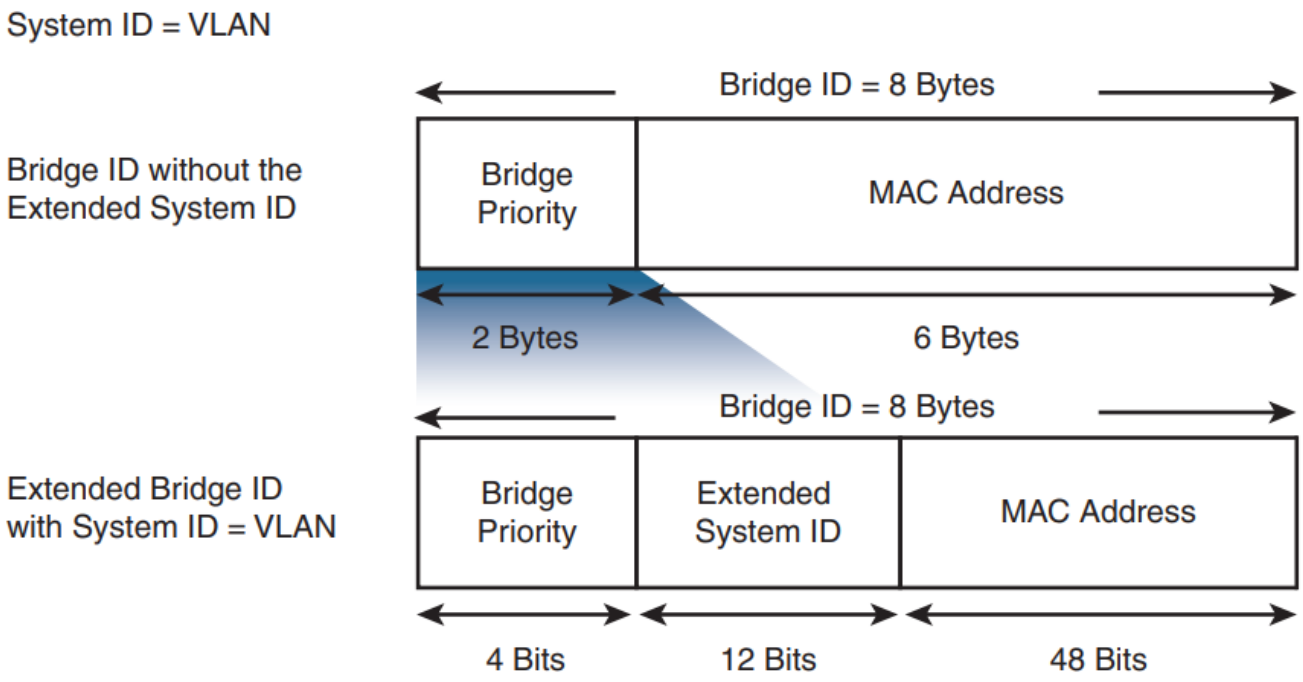
4. **NON-DESIGNATED PORT (ND):**
    a. The other port in the collision domain will be labeled as non-designated (blocking)
    b. There isn't an election process like the previous three, it's simply the last remaining port.

    Port State = Blocking
    Note*:* Every collision domain has a single spanning tree designated port. When we use switches, each link isa separate collision domain.

Extended System ID

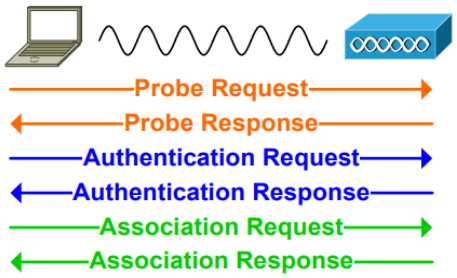
PVST+ requires a separate instance of spanning tree for each VLAN. The BID field in the BPDU must carry VLAN ID (VID) information, as Figure 25-5 shows.

Figure 25-5 Bridge ID for PVST+ with Extended System ID



2.6 Wireless Architectures and AP modes

Client Association



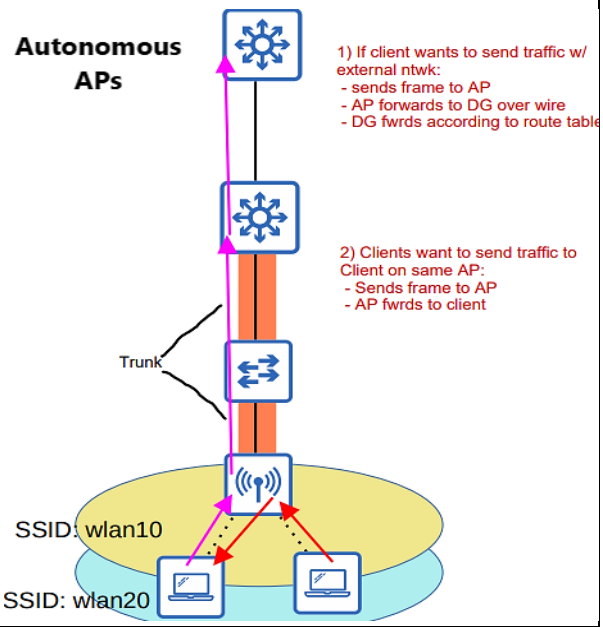
Three 802.11 message/frame types:

1. **Management**: used to manage the BSS.
2. **Control**: Used to control access to the medium (radio frequency). Assists with delivery of management and data frames.
3. **Data**: Used to send actual data packets.

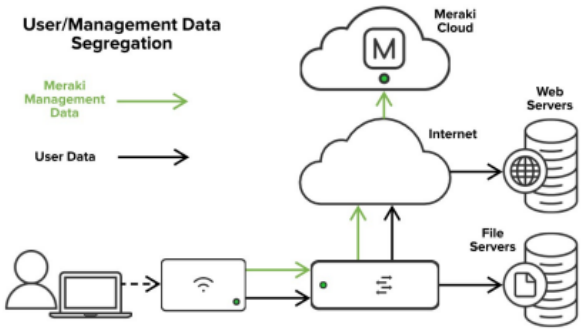
Type	Class
Association	Management
Authentication	Management
Probe	Management
Beacon	Management
Request to Send (RTS)	Control
Clear to Send (CTS)	Control
Acknowledgment (ACK)	Control
Data	Data

*Three main wireless AP deployment methods: **1)** Autonomous **2)** Lightweight **3)** Cloud-based*

1) **Autonomous**: Self-contained systems that do not rely on a WLC. Autonomous APs are configured individually. This type of AP management can also work as Repeater, Outdoor Bridge, and Workgroup Bridge.



3) **Cloud-based**: in-between autonomous AP and split-MAC architecture. Here, APs are autonomous but centrally managed in the cloud, such as Cisco Meraki.



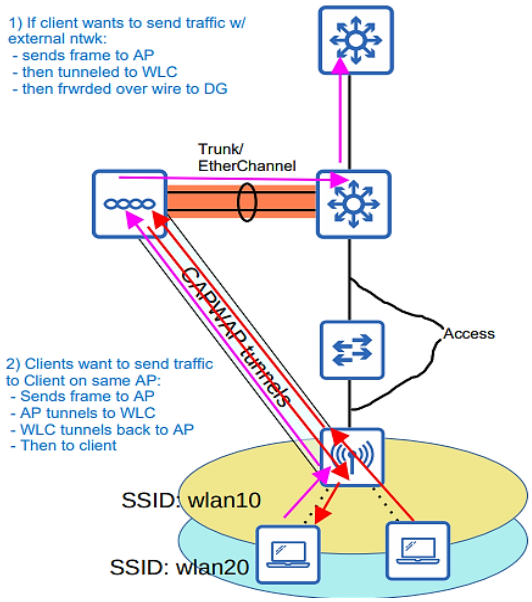
2) **Lightweight**: In Lightweight APs (LAP), the function of AP is split between the AP and the WLC. This is also known as **split-MAC architecture**.

AP MAC Functions (real-time)	WLC MAC Functions (mgmt)
▪ Beacons and probe responses ▪ Transmitting/receiving frames (packets) over the RF ▪ Encryption of wireless frames	▪ Authenticating clients ▪ Managing client associations ▪ Processing clients that are roaming ▪ Assigning nonoverlapping channels ▪ Termination of 802.11 traffic on a wired interface

Traffic from clients flows via the AP through a **Control and Provisioning of Wireless Access Points (CAPWAP)** layer 3 tunnel to the WLC, and then enters the rest of the network via the WLC. The WLC do not need to be in the same VLAN as the LAPs.

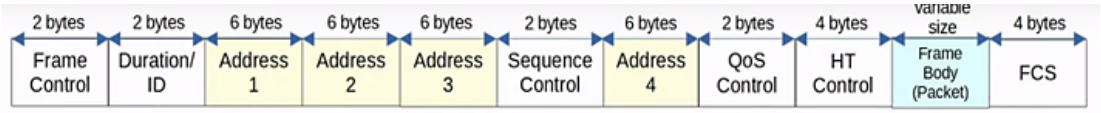
The CAPWAP tunnel consists of two tunnels:
- *Control tunnel*: UDP port 5246, always encrypted.
- *Data tunnel*: UDP port 5247, can be encrypted or not.

The split-MAC architecture has several benefits, such as: Scalability, Dynamic channel assignment, transmit pore optimization, self-healing wireless coverage, seamless roaming, client load balancing, security/QoS management, etc.



LAPs CAN OPERATE IN SEVERAL MODES:
- ***Local***: default, offers BSSs to wireless clients.
- ***FlexConnect***: allows LAP to transform between split-MAC to autonomous modes depends on the status of the connection to WLC (if WLC goes down, it switches to autonomous mode). Also offers BSSs to wireless clients.
- ***Monitor***: does not transmit, act as a sensor.
- ***Sniffer***: sniff the traffic and send to an analyzer.
- ***Rogue Detector***: detect rogue devices.
- ***Bridge/Mesh***: form a dedicated bridge/mesh between sites even at long distance.
- ***Flex plus Bridge***: adds FlexConnect to Bridge/Mesh.

802.11 Frame Format



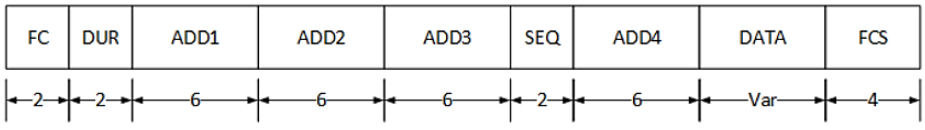
- **Frame Control**: Provides information such as the message type and subtype.
- **Duration/ID**: Depending on the message type, this field can indicate:
  → the time (in microseconds) the channel will be dedicated for transmission of the frame.
  → and identifier for the association (connection).
- **Addresses**: Up to four addresses can be present in an 802.11 frame. Which addresses are present, and their order, depends on the message type.
  → Destination Address (DA): Final recipient of the frame
  → Source Address (SA): Original sender of the frame
  → Receiver Address (RA): Immediate recipient of the frame
  → Transmitter Address (TA): Immediate sender of the frame
- **Sequence Control**: Used to reassemble fragments and eliminate duplicate frames.
- **QoS Control**: Used in QoS to prioritize certain traffic.
- **HT (High Throughput) Control**: Added in 802.11n to enable High Throughput operations.
  → 802.11n is also known as 'High Throughput' (HT) Wi-Fi
  → 802.11ac is also known as 'Very High Throughput' (VHT) Wi-Fi
- **FCS (Frame Check Sequence)**: Same as in an Ethernet frame, used to check for errors.

- 802.11 frames have a different format than 802.3 Ethernet frames.
- For the CCNA, you don't have to learn it in as much detail as the Ethernet and IP headers.
- Depending on the 802.11 version and the message type, some of the fields might not be present in the frame.
  → For example, not all messages use all 4 address fields.

How many address fields can be expected in an 802.11 data frame that is sent from a wireless station and destined to a host on the wired network? *(Select the best answer.)*

- ○ **A. three**
- ◉ B. four
- ○ C. one
- ○ D. two

Explanation:
An Institute of Electrical and Electronics Engineers (IEEE) 802.11 Media Access Control (MAC) frame is generally comprised of nine fields, as shown in the following diagram:



The address fields, Address 1 (ADD1), Address 2 (ADD2), Address 3 (ADD3), and Address 4 (ADD4), are 6-byte fields used to convey MAC address and Basic Service Set Identifier (BSSID) information. What information resides in which address field is entirely dependent on the type of frame. However, ADD1, ADD2, and ADD3 typically contain a source MAC address, destination MAC address, and BSSID with the order being dependent on whether the frame is entering the distribution system (DS), leaving the DS, or passing directly between ad-hoc wireless devices. The ADD4 field is only present for frames passing between devices in the DS, such as from one access point (AP) to another AP.

IEEE standard	Description
802.11k	provides assisted roaming in a wireless network
802.11r	provides support for fast transition (F T) roaming
802.11w	provides management frame protection (MFP)
802.11v	provides network-assisted power savings

Cisco AP Modes

Many Cisco APs can operate in either autonomous or lightweight mode, depending on which code image is loaded and run. From the WLC, you can also configure a lightweight AP to operate in one of the following special-purpose modes:

- **Local:** The default lightweight mode that offers one or more functioning BSSs on a specific channel. During times that it is not transmitting, the AP will scan the other channels to measure the level of noise, measure interference, discover rogue devices, and match against intrusion detection system (IDS) events.
- **Monitor:** The AP does not transmit at all, but its receiver is enabled to act as a dedicated sensor. The AP checks for IDS events, detects rogue access points, and determines the position of stations through location-based services.
- **FlexConnect:** An AP at a remote site can locally switch traffic between an SSID and a VLAN if its CAPWAP tunnel to the WLC is down and if it is configured to do so.
- **Sniffer:** An AP dedicates its radios to receiving 802.11 traffic from other sources, much like a sniffer or packet capture device. The captured traffic is then forwarded to a PC running network analyzer software such as Wildpackets OmniPeek or WireShark, where it can be analyzed further.
- **Rogue detector:** An AP dedicates itself to detecting rogue devices by correlating MAC addresses heard on the wired network with those heard over the air. Rogue devices are those that appear on both networks.
- **Bridge:** An AP becomes a dedicated bridge (point-to-point or point-to-multipoint) between two networks. Two APs in bridge mode can be used to link two locations separated by a distance. Multiple APs in bridge mode can form an indoor or outdoor mesh network.
- **Flex+Bridge:** FlexConnect operation is enabled on a mesh AP.
- **SE-Connect:** The AP dedicates its radios to spectrum analysis on all wireless channels. You can remotely connect a PC running software such as MetaGeek Chanalyzer or Cisco Spectrum Expert to the AP to collect and analyze the spectrum analysis data to discover sources of interference.

NOTE Remember that a lightweight AP is normally in local mode when it is providing BSSs and allowing client devices to associate to wireless LANs. When an AP is configured to operate in one of the other modes, local mode (and the BSSs) is disabled.

WLC Deployments

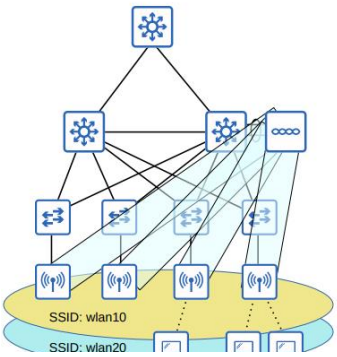
Table 27-2 Summary of WLC Deployment Models

Deployment Model	WLC Location (DC, Access, Central, AP)	APs Supported	Clients Supported	Typical Use
Unified	Central	6000	64,000	Large enterprise
Cloud	DC	3000	32,000	Private cloud
Embedded	Access	200	4000	Small campus
Mobility Express	Other	100	2000	Branch location
Autonomous	N/A	N/A	N/A	N/A

In a split-MAC architecture, there are four main WLC deployment models:

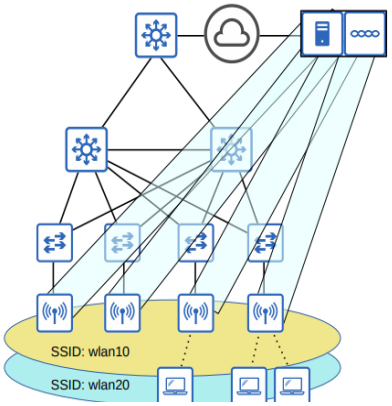
1. **UNIFIED**: The WLC is a hardware appliance in a central location of the network (*supports about **6000** Aps)*
2. **CLOUD-BASED**: The WLC is a VM running on a server, usually in a private cloud in a data center *(supports about **3000** Aps)*
   a. This is not the same as the cloud-based AP architecture discussed previously.
3. **EMBEDDED**: The WLC is integrated within a switch *(supports about **200** Aps)*
4. **MOBILITY EXPRESS**: The WLC is integrated within an AP *(supports about **100** Aps)*

1. Unified WLC



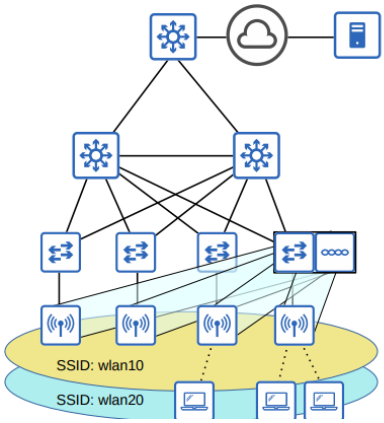
- The WLC is a hardware appliance deployed in a central location of the network.
- A unified WLC can support up to about 6000 APs.
- If more than 6000 APs are needed, additional WLCs can be added to the network.

2. Cloud-based WLC



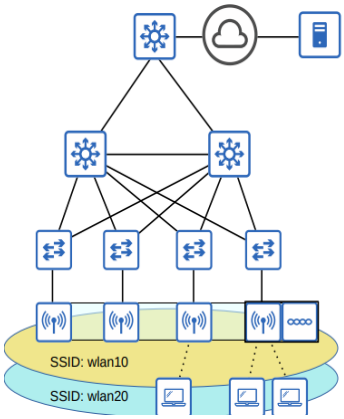
- The WLC is a VM running on a server, typically in a private cloud in a data center.
- Cloud-based WLCs can typically support up to about 3000 APs.
- If more than 3000 APs are needed, more WLC VMs can be deployed.

3. Embedded WLC

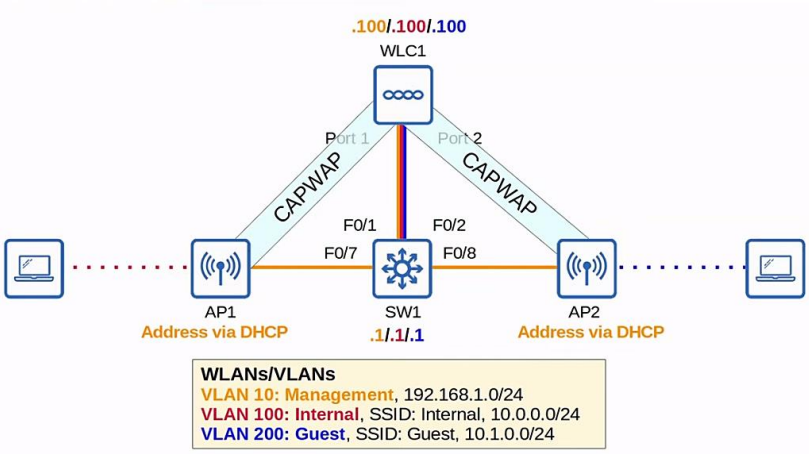


- The WLC is embedded within a switch.
- An embedded WLC can support up to about 200 APs.
- If more than 200 APs are needed, more switches with embedded WLCs can be added.

4. Cisco Mobility Express WLC



- The WLC is embedded within an AP.
- A Mobility Express WLC can support up to about 100 APs.
- If more than 100 APs are needed, more APs with embedded Mobility Express WLCs can be added.

In this Split-MAC deployment, WLC connects to SW1 via LAG (Link Aggregation Group) aka EtherChannel mode on and is Trunk (WLCs do not support PAgP/LACP. If PC on left (VLAN100) wants to msg PC2 (VLAN200): 1. PC1 sends traffic to DGtwy (SW1). 2. SW1 routes msg to VLAN200 and sends to WLC. 3. WLC sends msg to PC2 via CAPWAP tunnel and AP.	

WLC PORTS: PHYSICAL PORTS THAT CABLES CONNECT TO	**WLC INTERFACES: LOGICAL INTERFACES W/ IN WLC (SVIS)**
– **Service ports**: a dedicated management port used for out-of-band management. Must connect to an access port. – **Distribution system port**: Standard network ports that connect to the "distribution system" (the wired network) and used for data traffic. They usually connect to trunk ports and can form Link Aggregation Group (LAG), which operates as an EtherChannel without autogegotiation. – **Console port**: Standard console port. – **Redundancy port**: Used to connect to another WLC to form a High Availability (HA) pair	– **Management interface**: Used for management traffic such as Telnet, SSH, HTTP, HTTPS, RADIUS authentication, NTP, Syslog, etc. CAPWAP tunnels are also formed to/from the management interface. – **Redundancy management interface:** When two WLCs are connected by their redundancy ports, one will be active and the other will be 'stand by'. This interface is used to access the 'stand by' WLC. – **Virtual interface**: Used when communicating with wireless clients to relay DHCP requests, perform client web authentication, etc. – **Service port interface**: Used to connect to the service port. – **Dynamic interface**: Interfaces used to map WLAN to VLAN.

WLC Config

TASK	WLC STEPS
Create New Dynamic Interface: (a new *Dynamic Interface* should be created every VLAN connected to your WLC Split-Mac deployment)	CONTROLLER tab > New > setup Port num, Name of int, VLAN ID, IP Address, Mask, Gateway, DHCP Server > Apply (screen won't change) > Back Repeat if more interfaces required
Create new WLAN: (a new *WLAN* should be created for every VLAN on the network)	WLANs tab > Create New (dropdown) > Go (button) > Type: WLAN, Profile Name: [*just a name used to id WLAN within WLC*] > SSID: [*same as profile name*] > ID: [*needs to be unique to id WLAN within WLC*] > Apply (screen won't change) > Back move to Security tab > Layer 2 > [WPA+WPA2] > WPA2 Policy [enable via check box] > WPA2 Encryption [AES] > PSK [enable via check box] > Config password in blank text field [Cisco123} > Apply (screen won't change) > Back Repeat if more WLANs required
Associate Wireless Clients	Config > Wireless > SSID [Internal] > Authentication > WPA2-PSK [Cisco123]

LAYER 3 SECURITY (ADDITIONAL OPTIONS):

Web Authentication: After the wireless clients gets an IP address and tries to access a web page, they will have to enter a username and password to authenticate.

Web Passthrough: Similar to the above, but no username or password are required. A warning or statement is displayed, and the client simply has to agree to gain access to the Internet.

The **Conditional** and **Splash Page** web redirect options are similar, but additionally require 802.1X layer 2 authentication

WLC Miscellaneous

FlexConnect

Which of the following statements about FlexConnect ACLs is true? *(Select the best answer.)*

- A. They can be configured with a per-rule direction.
- B. They are not supported on the native VLAN.
- **C. They are applied per AP and per VLAN.**
- D. They do not support an implicit deny rule.

Explanation:
FlexConnect access control lists (ACLs) are applied per access point (AP) and per virtual local area network (VLAN). One possible application of FlexConnect ACLs is to prevent administration of the wireless local area network (WLAN) from a particular VLAN. FlexConnect ACLs are similar to traditional Cisco IOS ACLs in that they are rules that permit or deny traffic from a given source to a given destination. However, FlexConnect ACLs are configured on Cisco wireless lightweight AP VLAN interfaces if the lightweight AP is operating in FlexConnect mode. Even though FlexConnect ACLs are applied differently than traditional ACLs, it is important to name FlexConnect ACLs differently from any traditional ACLs that might be configured on the WLAN.

FlexConnect ACLs are supported on the native VLAN. Although it is possible to configure FlexConnect ACLs for the native VLAN, it is not possible to configure FlexConnect ACLs for the native VLAN if the VLAN configuration is inherited from a FlexConnect group.

FlexConnect ACLs cannot be configured with a per-rule direction. This is in contrast to a traditional ACL, which can be configured with inbound rules or outbound rules. A FlexConnect ACL is applied in the ingress direction or the egress direction as an entire set of rules, not on a per-rule basis.

FlexConnect ACLs support the implicit deny rule. In this way, FlexConnect ACLs work similarly to traditional ACLs. The implicit deny rule is an invisible rule that is applied to the end of an ACL. It ensures that traffic that is not explicitly matched by a previous rule in the ACL is denied by the ACL.

FlexConnect access control lists (ACLs) are supported on the native virtual local area network (VLAN).

You have enabled LAG on a WLC that contains eight distribution system ports.

How many ports will be included in the LAG bundle by default? *(Select the best answer.)*

- A. none
- B. four
- C. one
- **D. eight**

3. IP Connectivity

- A *connected route* is a route to the network an interface is connected to. Connected routes are indicated by code C in the routing table. If a router receives a packet destined for a host in a directly-connected network, it will forward the packet directly to the destination host (in a frame addressed to the host's MAC address).
- A *local route* is a route to the exact IP address configured on the interface. Local routes use a /32 prefix length to specify a single IP address. If a router receives a packet destined for the IP address of a local route, it means the packet is destined for the router itself; the router will receive the packet for itself, it will not forward it.
- A route to more than one destination IP address (any route with a prefix length shorter than /32) is called a *network route*. A connected route is an example of a network route.
- A route to a single destination IP address (a route with a /32 prefix length) is called a *host route*. A local route is an example of a host route.

3.1 Interpret the components of routing table

3.1.a Routing protocol code

Routing Protocol Legend in show ip route

To save space in the book, the various examples of the **show ip route** command omit the legend that begins the command's output. To bring attention to those codes, refer to the legend from the **show ip route** command in Example B-1. Make an effort to commit the highlighted codes to memory.

Example B-1 show ip route—*Most Common Legend Codes*

```
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
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, m - OMP
       n - NAT, Ni - NAT inside, No - NAT outside, Nd - NAT DIA
       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
       H - NHRP, G - NHRP registered, g - NHRP registration summary
       o - ODR, P - periodic downloaded static route, l - LISP
       a - application route
       + - replicated route, % - next hop override, p - overrides from PfR
       & - replicated local route overrides by connected
! Lines omitted for brevity
```

DYNAMIC ROUTING PROTOCOLS

CATEGORY	ALGORITHM	PROTOCOL	METRIC
IGP	DISTANCE VECTOR	RIP	hop Count
		EIRGP	bandwidth and delay
	LINK STATE	OSPF	cost
		IS-IS	cost
EGP	PATH VECTOR	BGP	

Table 19-3 Interior IP Routing Protocols Compared

Feature	RIPv2	EIGRP	OSPF
Classless/sends mask in updates/supports VLSM	Yes	Yes	Yes
Algorithm (DV, advanced DV, LS)	DV	Advanced DV	LS
Supports manual summarization	Yes	Yes	Yes
Cisco-proprietary	No	Yes[1]	No
Routing updates are sent to a multicast IP address	Yes	Yes	Yes
Convergence	Slow	Fast	Fast

Dynamic routing protocols can be divided into two main categories:

- **IGP**: Interior Gateway Protocol, to share routes within a single autonomous system (AS).

  There are two types of algorithm:

  - Distance Vector: including Routing Information Protocol (RIP), and Enhanced Interior Gateway Routing Protocol (EIGRP).
  - Link State: including Open Shortest Path First (OSPF), and Intermediate System to Intermediate System (IS-IS).

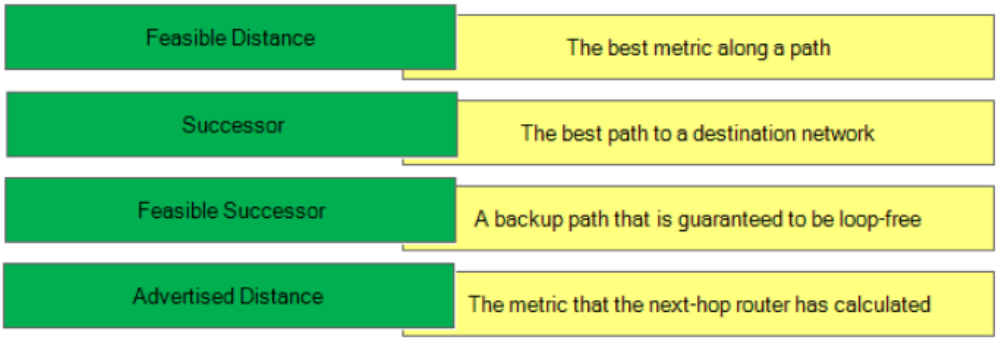
- **EGP**: Exterior Gateway Protocol, to share routes between autonomous systems.

  - EGP uses Path Vector algorithm, including Border Gateway Protocol (BGP).

Distance Vector Protocols operate by sending their known destination networks with metrics to reach these networks. This method of sharing route is often called "routing by rumor".

In a Link State Routing protocol, every router create a "connectivity map" of the network. Routers do that by advertising information of their connected networks to their neighbors and these advertisements are passes along to other routers, until all routers in the network develop the same map of the network. Each router independently uses this map to calculate the best routes to each destination.

EIGRP:



Explanation:

Feasible distance (FD) is the Enhanced Interior Gateway Routing Protocol (EIGRP) term for the best metric along a path to a destination. The FD includes the metric to a neighbor router plus that neighbor router's advertised distance (AD) to the destination network. The AD, which is also called the reported distance (RD), is the metric that has been calculated by the next-hop router.

The successor is the best path to a destination network. The route with the lowest FD is chosen as the successor. The successor route is placed in the routing table and is used to route packets to the destination network.

A feasible successor is a backup path that is guaranteed to be loop-free and can be used if the successor route goes down. If the AD for a nonsuccessor route is less than the FD of the successor, the route is a feasible successor. If the AD of a route is greater than the FD of the successor, the route cannot be guaranteed to be free of loops and cannot be chosen as a feasible successor.

EIGRP Convergence Terms

This final part of the EIGRP discussion defines a few EIGRP terms. EIGRP calls the best route for a subnet on a router the *successor* route. For example, in Figure B-2, Router A finds three routes to reach subnet 1. It calculates a metric for the routes through neighboring Routers B, C, and D. Router A chooses the least metric route through Router D as the successor route. EIGRP refers to the metric for that best route as the *feasible distance*, or FD. Router A places the successor route, and only that route, in its routing table.

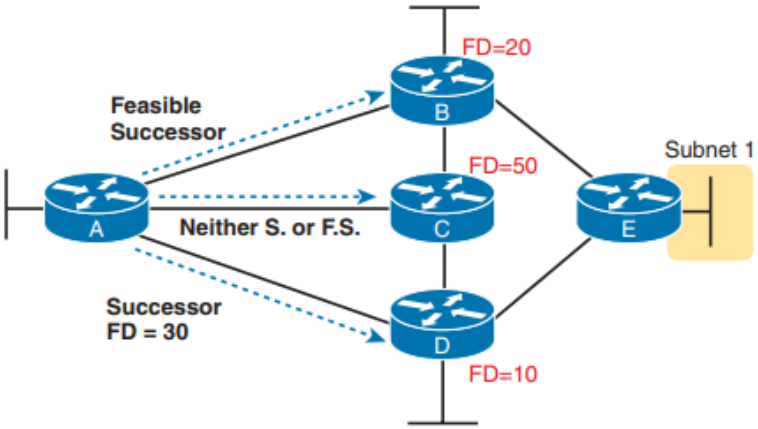


Figure B-2 *Successor and Feasible Successor Routes*

EIGRP does not place alternate routes in the routing table—but it will use those if the successor route fails. In anticipation of that, EIGRP categorizes alternate routes as either feasible successor routes or others (with no special name). EIGRP can replace a failed successor route with a feasible successor route immediately, without risking creating a routing loop—but not so with an alternate route that is not a feasible successor.

Now, knowing the categories of alternate routes, the question becomes: How does EIGRP determine whether an alternate route is a feasible successor route? Generally:

> On one router, with multiple known routes for the same subnet, knowing the successor route, if any alternate route's next-hop router has a lower FD than the local router's FD, the route is a feasible successor route.

The preceding statement begs for an example. Using Figure B-2 again, consider whether Router A's alternate routes with next-hop Routers B and C are feasible successor routes. First, Router A's successor route has an FD of 30. The figure shows the FD of next-hop routers B and C. Applying the logic:

- Router B's FD of 20 is less that Router A's FD of 30, so Router A's alternate route that uses Router B as the next hop router is a feasible successor.
- Router C's FD of 50 is more than Router A's FD of 30, so Router A's alternate route that uses Router C as the next hop router is NOT a feasible successor.

Dynamic Routing Protocol Metrics

- A router's route table contains the best route to each destination network it knows about.
- If a router using a dynamic routing protocol learns two different routes to the same destination, how does it determine which is 'best'?
- It uses the metric value of the routes to determine which is best. A lower metric = better
- Each routing protocol uses a different metric to determine which route is the best.

ECMP (Equal Cost Multi-Path):
If a router learns two (or more) routes via the same routing protocol to the same destination (same network address, same subnet mask) with the same metric, both will be added to the routing table. Traffic will be load balanced over both routes:

O 192.168.4.0/24 [110/3] via 10.0.13.2, GigabitEthernet1/0
192.168.4.0/24 [110/3] via 10.0.12.2, GigabitEthernet0/0

[AD/Metric]

ABOVE, BOTH ROUTES ARE:

1. OSPF (AD = 110)
2. To same Dest
3. Same Metric (3)

*This would result in the two routes Load Balancing aka **ECMP (Equal Cost Multi-Path)***

3.1.e Administrative distance

ADMINISTRATIVE DISTANCE

- In most cases a company will only use a single IGP — usually OSPF or EIGRP.
- However, in some rare cases they might use two. For example, if two companies connect their networks to share information, two different routing protocols might be in use.
- Metric is used to compare routes learned via the same routing protocol.
- Different routing protocols use totally different metrics, so they cannot be compared.
- For example, an OSPF route to 192.168.4.0/24 might have a metric of 30, while an EIGRP route to the same destination might have a metric Of 33280. Which route is better? Which route should the router put in the route table?
- The **administrative distance (AD)** is used to determine which routing protocol is preferred.
- A lower AD is preferred and indicates that the routing protocol is considered more 'trustworthy' (more likely to select good routes).

ROUTE SOURCE	ADMINISTRATIVE DISTANCE
Directly Connected	0
Static route	1
eBGP	20
EIGRP (internal)	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
EIGRP (external)	170
Internal BGP (iBGP)	200
Unknown*	255

Memory Table

3.2 Determine how a router makes a forwarding decision by default

3.2.a Longest prefix match

You issue the **show ip route** command on RouterA and receive the following partial output:

```
S 10.20.0.0/22 [1/0] via 192.168.10.2
R 10.20.0.0/24 [120/3] via 192.168.10.3, 00:33:38, Serial0/3
D 10.20.0.0/26 [90/2809856] via 192.168.10.4, 00:02:14, Serial0/4
O 10.20.0.0/28 [110/64] via 192.168.10.1, 00:02:38, Serial0/1
```

RouterA receives a packet that is destined for 10.20.0.14.

Which of the following routes will RouterA use to send the packet? (Select the best answer.)

A. ○ the RIP route, because it has the highest administrative distance
B. ◉ the OSPF route, because it is the route with the longest prefix match
C. ○ the static route, because static routes are preferred over dynamic routes
D. ○ the EIGRP route, because it has the lowest administrative distance

Prefix Lengths

Look at another scenario to see how the router handles another common situation: varying prefix lengths. Assume, again, that a router runs has four routing processes, and each process has received these routes:

- EIGRP (internal): 192.168.32.0/26
- RIP: 192.168.32.0/24
- OSPF: 192.168.32.0/19

Which of these routes can be installed in the routing table? Since EIGRP internal routes have the best administrative distance, you can assume the first one can be installed. However, since each of these routes has a different prefix length (subnet mask), they are considered different destinations, and they can all be installed in the routing table.

The next section provides the information from the routing table to make forwarding decisions.

Make Forwarding Decisions

Look at the three routes that were installed in the routing table and see how they look on the router.

```
router# show ip route
      ....
      D   192.168.32.0/26 [90/25789217] via 10.1.1.1
      R   192.168.32.0/24 [120/4] via 10.1.1.2
      O   192.168.32.0/19 [110/229840] via 10.1.1.3
      ....
```

If a packet arrives on a router interface destined for 192.168.32.1, which route would the router choose? It depends on the prefix length, or the number of bits set in the subnet mask. Longer prefixes are always preferred over shorter ones when forwarding a packet.

In this case, a packet destined to 192.168.32.1 is directed toward 10.1.1.1, because 192.168.32.1 falls within the 192.168.32.0/26 network (192.168.32.0 to 192.168.32.63). It also falls within the other two routes available, but the 192.168.32.0/26 has the longest prefix within the routing table (26 bits verses 24 or 19 bits).

Likewise, if a packet destined for 192.168.32.100 arrives on one of the router interfaces, it is forwarded to 10.1.1.2, because 192.168.32.100 does not fall within 192.168.32.0/26 (192.168.32.0 through 192.168.32.63), but it does fall within the 192.168.32.0/24 destination (192.168.32.0 through 192.168.32.255). Again, it also falls into the range covered by 192.168.32.0/19, but 192.168.32.0/24 has a longer prefix length.

Router Forwarding Decision Logic (Summary):

- **Lowest AD** (*different routing protocol*): when multiple routes to the EXACT same network exist (ie. all entries say 192.168.1.0/26) and each uses a *different* routing protocol, a router prefers the routing protocol with the lowest AD.
- **Lowest metric** (*same routing protocol*): when multiple routes to the EXACT same network exist (ie. all entries say 192.168.1.0/26) and each uses the *same* routing protocol, a router prefers the route with the lowest metric.
- **Longest Prefix Match**: when multiple routes to overlapping networks exist (ie. 192.168.1.0/26, 192.168.1.0/24, 192.168.1.0/19), a router will prefer the most specific route, which is the route with the longest prefix match.

3.3 Config/Verify IPv4/IPv6 static routing

Recursive Static Routes

In a recursive static route, only the next hop is specified. The output interface is derived from the next hop. This example shows such a definition:

```
ipv6 route 2001:DB8::/32 2001:DB8:3000:1
```

This example specifies that all destinations with address prefix 2001:DB8::/32 are reachable via the host with address 2001:DB8:3000:1.

Fully Specified Static Routes

In a fully specified static route, both the output interface and the next hop are specified. This form of static route is used when the output interface is a multi-access one and it is necessary to explicitly identify the next hop. The next hop must be directly attached to the specified output interface. The following example shows a definition of a fully specified static route:

```
ipv6 route 2001:DB8:/32 gigabitethernet1/0/0 2001:DB8:3000:1
```

3.3.d Floating static

A floating static route is used to provide link redundancy.

Floating Static Route

Floating static routes are typically configured with an AD value that is numerically higher than a dynamic routing protocol, which ensures that the routing protocol path is always selected by the router unless the path becomes unavailable.

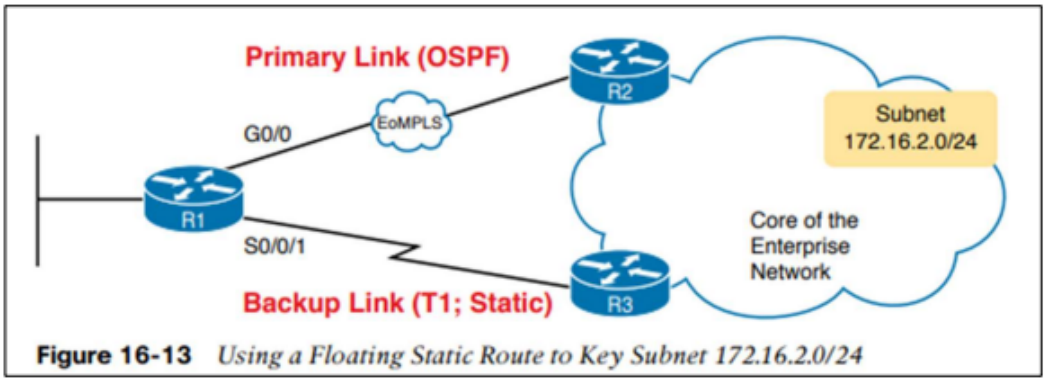


Figure 16-13 *Using a Floating Static Route to Key Subnet 172.16.2.0/24*

For static routes, to change the distance of each route enter a distance after the ip route command:

```
ip route network subnet mask next hop distance
```

The AD metric `distance` needs to be higher than the Dynamic Routing Protocol AD. So, if OSPF is used with an AD metric of 110, you would add a higher value in the `distance` value because we want the Floating Static route to be a backup while keeping the OSPF route as the default. So we used 111 for AD metric.

```
R1(config)#ip route 172.16.2.0 255.255.255.0 172.16.5.3 130
```

The Floating Static route we just configured does NOT show up in R1s route table. It's not here, because we set the AD higher than the OSPF route's 110, the OSPF route. So the OSPF route is selected instead of the Static one. The Static route will only appear in the route table if the dynamic routing protocol route is removed or goes down.

3.4 Configure and verify single area OSPFv2

3.4.a Neighbor adjacencies

Requirement for two connected routers to be neighbors:

- Area numbers must match.
- There are interfaces in the same subnet.
- OSPF process must not be shutdown.
- Router IDs must be unique.
- Hello and Dead timers must match.
- Authentication settings must match.
- IP MTU settings must match (only needed for proper function of OSPF).
- OSPF Network Type must match (only needed for proper function of OSPF).

Once routers become neighbors, they automatically do the work of sharing network information, calculating routes, etc.

When OSPF is activated on an interface, the router starts sending OSPF hello messages out of the interface at regular intervals (determined by the hello timer, which is 10 seconds by default). These are used to introduce the router to potential OSPF neighbors.

All OSPF messages are encapsulated in an IP header with the value of **89** in the Protocol Field.

7-Part Process of forming a neighbor:

Down: Initial state.

Init: Hello packet received, but own router ID is not in the hello packet.

2-way: 2 hello packets with both routers ID are transferred between 2 routers.

Now, the routers are OSPF neighbors, ready to share LSAs. However, for the Broadcast network type, more steps are needed:

Exstart: The router with higher Router ID become the master and start the exchange by sending DBD (Database Description) packets.

Exchange: The basic information using DBDs happen here.

Loading: Routers send Link State Request (LSR) messages to request their neighbors any LSAs they do not have. Other routers will send LSU (Link State Uupdate) and then LSAck is sent to acknowledge the LSU.

Full: The routers have full OSPF adjacency and identical LSDBs. They continue to send and listen for Hello packets (every **10 seconds**) by default) to maintain the adjacency. If no Hello packet is received after a Dead timer (**40 seconds** by default), the neighbor is removed.

MNUEMONIC

Davis **I**sland **2** **E**xtra **E**asy **L**azy **F**ugs

Configuring OSPF

`Router(config)#router ospf 123`	Starts OSPF process 123. The process ID is any positive integer value between 1 and 65,535. The process ID *is not related to* the OSPF area. The process ID merely distinguishes one process from another within the device.
`Router(config-router)#network 172.16.10.0 0.0.0.255 area 0`	OSPF advertises interfaces, not networks. Uses the wildcard mask to determine which interfaces to advertise. Read this line to say "Any interface with an address of 172.16.10.x is to be put into area 0."
	NOTE The process ID number of one router does not have to match the process ID of any other router. Unlike Enhanced Interior Gateway Routing Protocol (EIGRP), matching this number across all routers does *not* ensure that network adjacencies will form.

Loopback Interfaces

`Router(config)#interface loopback 0`	Creates a virtual interface named loopback 0, and then moves the router to interface configuration mode. The loopback interface number can be any number between 0 and 2147483647.
`Router(config-if)#ip address 192.168.100.1 255.255.255.255`	Assigns the IP address to the interface.
	NOTE Loopback interfaces are always "up and up" and do not go down unless manually shut down. This makes loopback interfaces great for use as OSPF router IDs.

Router ID

`Router(config)#router ospf 1`	Starts OSPF process 1.
`Router(config-router)#router-id 10.1.1.1`	Sets the router ID to 10.1.1.1. If this command is used on an OSPF router process that is already active (has neighbors), the new router ID is used at the next reload or at a manual OSPF process restart.
`Router(config-router)#no router-id 10.1.1.1`	Removes the static router ID from the configuration. If this command is used on an OSPF router process that is already active (has neighbors), the old router ID behavior is used until the next reload or at a manual OSPF process restart.

Note

The OSPF router ID is used to identify each router in the OSPF routing domain. It is a label and is expressed as an IPv4 address. The precedence used to determine the OSPF router ID is as follows:

1. The IP address set using the **router-id** command
2. The highest IP address of its loopback interfaces
3. The highest IP address of its physical interfaces
   a. This address does not have to be included in an OSPF **network** command, but it does have to be in an up/up state.
   b. If the interface used as the router ID goes down, this will trigger a recalculation of the OSPF routing table, starting with the reforming of neighbor adjacencies using the new router ID.

Note

Even though the router ID looks like an IPv4 address, it is not routable and therefore not included in the routing table unless the OSPF process chooses an interface that has been appropriately defined by a **network** command.

DR/BDR Elections

`Router(config)#interface fastethernet 0/0`	Changes the router to interface configuration mode.
`Router(config-if)#ip ospf priority 50`	Changes the OSPF interface priority to 50.
	NOTE The assigned priority can be between 0 and 255. A priority of 0 makes the router ineligible to become a designated router (DR) or backup designated router BDR). The highest priority wins the election. A priority of 255 guarantees a tie in the election, assuming other routers are also set to 255. If all routers have the same priority, regardless of the priority number, they tie. Ties are broken by the highest router ID.

Modifying Cost Metrics

`Router(config)#interface serial 0/0/0`	Changes the router to interface configuration mode.
`Router(config-if)#bandwidth 128`	If you change the bandwidth, OSPF recalculates the cost of the link.
Or	
`Router(config-if)#ip ospf cost 1564`	Changes the cost to a value of 1564.
	NOTE The cost of a link is determined by dividing the reference bandwidth by the interface bandwidth. The bandwidth of the interface is a number between 1 and 10,000,000. The unit of measurement is kilobits.The cost is a number between 1 and 65,535. The cost has no unit of measurement—it is just a number.

OSPF auto-cost reference-bandwidth

`Router(config)#router ospf 1`	Starts OSPF process 1.
`Router(config-router)#auto-cost reference-bandwidth 1000`	Changes the reference bandwidth that OSPF uses to calculate the cost of an interface.
	NOTE The range of the reference bandwidth is 1 to 4,294,967. The default is 100. The unit of measurement is megabits per second (Mbps).
	NOTE The value set by the **ip ospf cost** command overrides the cost resulting from the **auto-cost** command.
	TIP If you use the command **auto-cost reference-bandwidth** *bandwidth*, configure all the routers to use the same value. Failure to do so will result in routers using a different reference cost to calculate the shortest path, resulting in potential suboptimum routing paths.

Timers

`Router(config-if)#ip ospf hello-interval timer 20`	Changes the Hello Interval timer to 20 seconds.
`Router(config-if)#ip ospf dead-interval 80`	Changes the Dead Interval timer to 80 seconds.
	NOTE Hello and Dead Interval timers must match for routers to become neighbors.

Propagating a Default Route

`Router(config)#ip route 0.0.0.0 0.0.0.0 s0/0/0`	Creates a default route.
`Router(config)#router ospf 1`	Starts OSPF process 1.
`Router(config-router)#default-information originate`	Sets the default route to be propagated to all OSPF routers.
`Router(config-router)#default-information originate always`	The **always** option propagates a default "quad-zero" route even if one is not configured on this router.
	NOTE The **default-information originate** command or the **default-information originate always** command is usually only to be configured on your "entrance" or "gateway" router, the router that connects your network to the outside world—the Autonomous System Boundary Router (ASBR).

Verifying OSPF Configuration

`Router#show ip protocol`	Displays parameters for all protocols running on the router
`Router#show ip route`	Displays a complete IP routing table
`Router#show ip ospf`	Displays basic information about OSPF routing processes
`Router#show ip ospf interface`	Displays OSPF info as it relates to all interfaces
`Router#show ip ospf interface fastethernet 0/0`	Displays OSPF information for interface fastethernet 0/0
`Router#show ip ospf border-routers`	Displays border and boundary router information
`Router#show ip ospf neighbor`	Lists all OSPF neighbors and their states
`Router#show ip ospf neighbor detail`	Displays a detailed list of neighbors
`Router#show ip ospf database`	Displays contents of the OSPF database
`Router#show ip ospf database nssa-external`	Displays NSSA external link states

Table 19-7 OSPF Design Terminology

Term	Description
Area Border Router (ABR)	An OSPF router with interfaces connected to the backbone area and to at least one other area
Backbone router	A router connected to the backbone area (includes ABRs)
Internal router	A router in one area (not the backbone area)
Area	A set of routers and links that shares the same detailed LSDB information, but not with routers in other areas, for better efficiency
Backbone area	A special OSPF area to which all other areas must connect—area 0
Intra-area route	A route to a subnet inside the same area as the router
Interarea route	A route to a subnet in an area of which the router is not a part

Table 20-3 Faster Interfaces with Equal OSPF Costs

Interface	Interface Default Bandwidth (Kbps)	Formula (Kbps)	OSPF Cost
Serial	1544 Kbps	100,000 / 1544	64
Ethernet	10,000 Kbps	100,000 / 10,000	10
Fast Ethernet	100,000 Kbps	100,000/100,000	1
Gigabit Ethernet	1,000,000 Kbps	100,000/1,000,000	1
10 Gigabit Ethernet	10,000,000 Kbps	100,000/10,000,000	1
100 Gigabit Ethernet	100,000,000 Kbps	100,000/100,000,000	1

BROADCAST	POINT-TO-POINT
Default on **Ethernet, FDDI** interfaces	Default on **HDLC, PPP** (serial) interfaces
DR/DBR elected	No DR/BDR
Neighbors dynamically discovered	Neighbors dynamically discovered
Default timers: Hello 10. Dead 40	Default timers: Hello 10. Dead 40
Multicast address 224.0.0.5	Multicast address 224.0.0.5

(**Non-broadcast** network type default timers = Hello 30, Dead 120)

3.4c Broadcast (DR/BDR selection)

The priority of selecting DR/DBR, where the highest will become the DR, the second highest will become the BDR:

1. OSPF interface priority (default is 1 on all interfaces).
2. Highest OSPF Router ID.

3.4.d Router ID

Each router is assigned a Router ID, in order of:

1. Manual configuration by **(config-router)#router-id [router-id]**
2. Highest IP address on a loopback interface.
3. Highest IP address on a physical interface.

- OSPF's metric is called **cost**
- It is automatically calculated based on the bandwidth (speed) of the interface.
- It is calculated by dividing a **reference bandwidth** value by the interface's bandwidth.
- The default reference bandwidth is 100 mbps.
  Reference: 100 mbps / **Interface**: 10 mbps = cost of 10
  Reference: 100 mbps / **Interface**: 100 mbps = cost of 1
  Reference: 100 mbps / **Interface**: 1000 mbps = cost of 1??
  Reference: 100 mbps / **Interface**: 10000 mbps = cost of 1??
- All values less than 1 will be converted to 1.
- Therefore FastEthernet, Gigabit Ethernet, 10Gig Ethernet, etc. are equal and all have a cost of 1 by default.
- You can (and should!) change the reference bandwidth with this command:
  `R1(config-router)# auto-cost reference-bandwidth megabits-per-second`

Three ways to modify the OSPF cost:

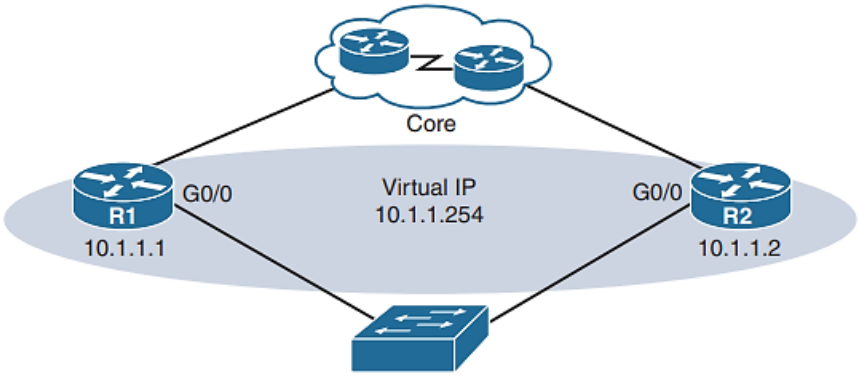
1) Change the **reference bandwidth**:
`R1(config-router)# auto-cost reference-bandwidth megabits-per-second`

2) Manual configuration
`R1(config-if)# ip ospf cost cost`

3) Change the **interface bandwidth**
`R1(config-if)# bandwidth kilobits-per-second`

3.5 FHRP: Describe the purpose, function, and concepts of first hop redundancy protocols

Figure 24-3 Redundant Default Gateway Example



A first hop redundancy protocol (FHRP) is a networking protocol designed to protect the default gateway used on a network by allowing two or more routers to provide backup for that address. If an active router fails, the backup router will quickly take over the address.

FHRP is 'non-preemptive'. The current active router will not automatically give up its role, even if the former active router returns.

To set up FHRP, a virtual IP is configured on the two routers, and a virtual MAC address is generated on the virtual IP. Then, and active and a standby routers are elected.

The active routers reply to ARP request using virtual MAC address. When the standby router takes over the active one, a **gratuitous ARP** (ARP that does not need reply) is broadcasted to inform the new router location in the network.

There are 3 main FHRPs:

- Hot Standby Router Protocol (HSRP).
- Virtual Router Redundancy Protocol (VRRP).
- Gateway Load Balancing Protocol (GLBP), using Active Virtual Gateway/Forwarders instead of active/standby routers.

Those 3 FHRPs are as follow:

FHRP	TERMINOLOGY	MULTICAST IP	VIRTUAL MAC	CISCO PROPRIETARY	ACTIVE/ACTIVE LOAD BALANCING
HSRPv1	Active/Standby	224.0.0.2	0000.0c07.acXX	Yes	No

HSRPv2	Active/Standby	224.0.0.102	0000.0c9f.fXXX	Yes	No
VRRP	Master/Backup	224.0.0.18	0000.5e00.01XX	No	No
GLBP	AVG/AVF	224.0.0.102	0007.b400.XXYY	Yes	Yes

The following statements are true regarding Hot Standby Router Protocol (HSRP):

- One router is elected as the active router, and another router is elected as the standby router.
- In an HSRP group, only one virtual IP address and one virtual MAC address is used.

Only the active router will use the Hot Standby Router Protocol (HSRP) virtual Internet Protocol (IP) address and will respond to Address Resolution Protocol (ARP) requests with the HSRP virtual Media Access Control (MAC) address. HSRP is a Cisco-proprietary protocol that enables two or more routers to act as a single virtual router. Multiple routers are assigned to an HSRP group, and the routers function as a single gateway. The HSRP virtual IP address can then be configured as the default gateway address for client devices.

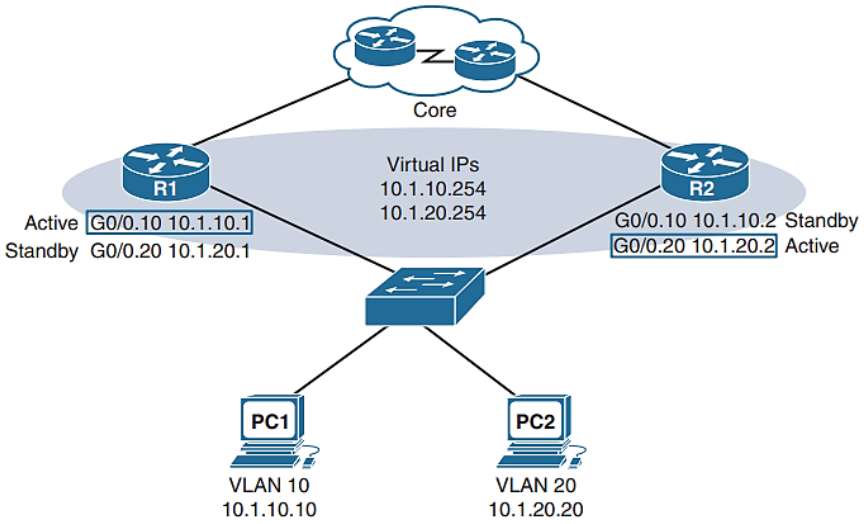
HSRP is a Cisco-proprietary First-Hop Redundancy Protocol (FHRP). Based on priority value, HSRP elects a single active router and a standby router. The active router is the router with the highest priority; it forwards packets, responds to ARP requests with a virtual MAC address, and can be the only router that is explicitly configured with the virtual IP address.

The standby router is the router with the second-highest priority. If multiple HSRP routers have the same priority, the router with the highest IP address will be elected as the active router. The router with the second-highest IP address will be elected as the standby router, which will assume the role of the active router if the active router fails. To participate in the active and standby router election process, each HSRP router must be a member of the same group. An HSRP group is identified by a group number from 0 through 255. The default HSRP group value is 0.

The election of a master router and the placement of all other routers in the group into the backup state are characteristics of Virtual Router Redundancy Protocol (VRRP). Like HSRP, VRRP provides router redundancy and only one router is active at any time. If the master router becomes unavailable, one of the backup routers will become the master router.

Routers in a single HSRP group cannot forward a portion of the traffic in a load-balancing fashion. You can provide load balancing by configuring multiple HSRP groups or by configuring Gateway Load Balancing Protocol (GLBP). GLBP elects an active virtual gateway (AVG) and up to four primary active virtual forwarders (AVFs). The AVG in a GLBP group assigns a virtual MAC address to the AVFs. When the AVG receives Address Resolution Protocol (ARP) requests that are sent to the virtual IP address for the GLBP group, the AVG responds with different virtual MAC addresses. This provides load balancing, because each of the primary AVFs will participate by forwarding a portion of the traffic sent to the virtual IP address.

Figure 24-4 HSRP Load Balancing Example



To implement HSRP load balancing for different VLANs, configure R1 as the active router for half the VLANs and R2 as the active router for the other half of the VLANs (see Example 24–7).

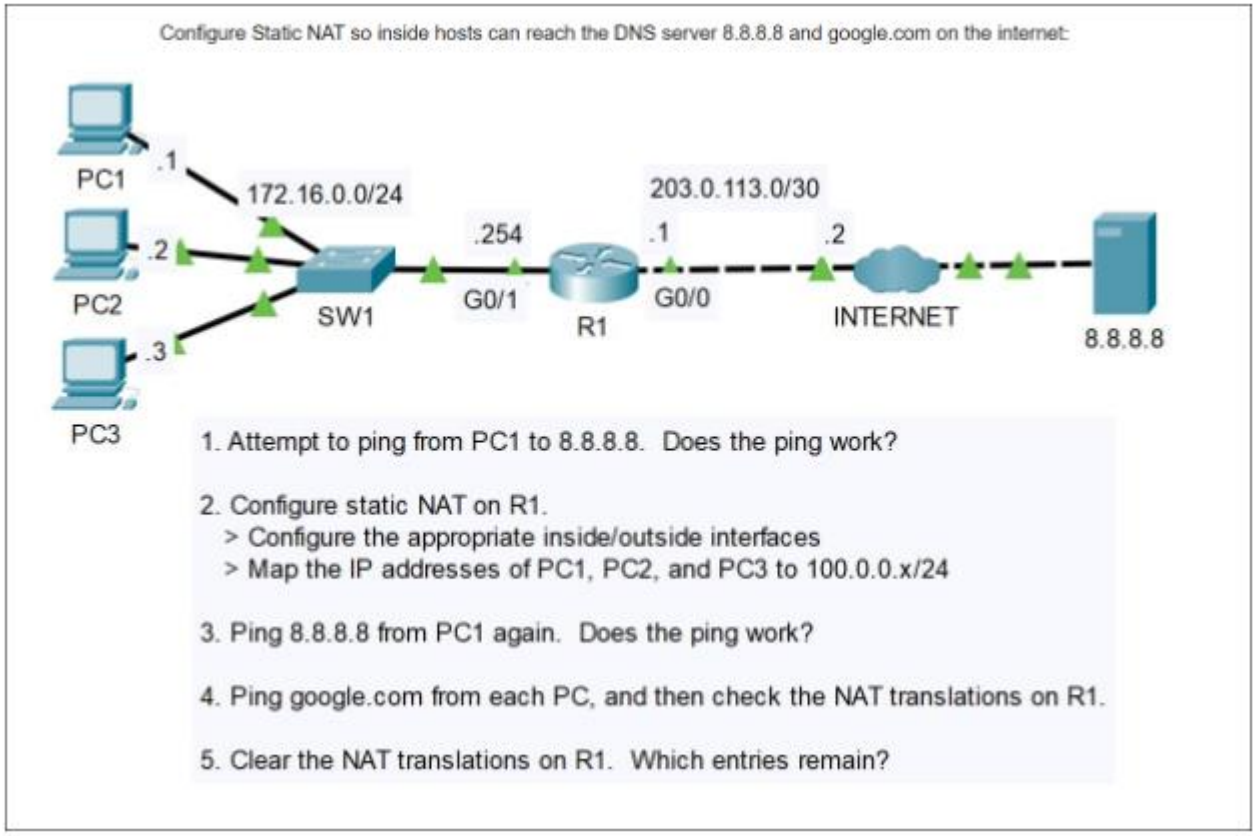
Hot Standby Routing Protocol (HSRP) does not do load balancing.
Therefore, only one router is forwarding per VLAN per group in HSRP. You must implement the above.

To allow the hosts to remain unchanged, the routers have to do some more work, as defined by one of the FHRP protocols. Generically, each FHRP makes the following happen:

1. All hosts act like they always have, with one default router setting that never has to change.
2. The default routers share a virtual IP address in the subnet, defined by the FHRP.
3. Hosts use the FHRP virtual IP address as their default router address.
4. The routers exchange FHRP protocol messages so that both agree as to which router does what work at any point in time.
5. When a router fails or has some other problem, the routers use the FHRP to choose which router takes over responsibilities from the failed router.

FHRP Option	HSRPv2	VRRPv3	GLBP
Cisco Proprietary	Yes	No	Yes
VIP must differ from the routers' interface IP addresses	Yes	No	Yes
Preemption off by default	Yes	No	Yes
Allows preemption (or not)	Yes	Yes	Yes
Active/active load balancing with multiple active routers in one group	No	No	Yes
IPv4 multicast address used	224.0.0.102	224.0.0.18	224.0.0.102
Group numbers supported in IOS	0–4095	1–255	0–1023
Virtual MAC address pattern	0000.0c9f.fzzz	0000.5e00.01xx	0007.b40x.xxrr

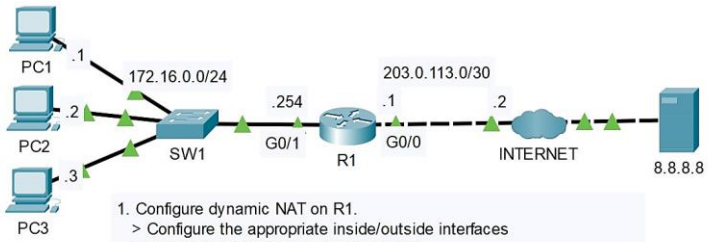
Static NAT




```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int g0/1
R1(config-if)#ip nat inside
R1(config-if)#int g0/0
R1(config-if)#ip nat outside
R1(config-if)#ex
R1(config)#ip nat inside source static 172.16.0.1 100.0.0.1
R1(config)#ip nat inside source static 172.16.0.2 100.0.0.2
R1(config)#ip nat inside source static 172.16.0.3 100.0.0.3
R1(config)#
```

Verify the static NATmappings in the NAT table:

```
R1#show ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
---  100.0.0.1         172.16.0.1         ---                ---
---  100.0.0.2         172.16.0.2         ---                ---
---  100.0.0.3         172.16.0.3         ---                ---
```



1. Configure dynamic NAT on R1.
   > Configure the appropriate inside/outside interfaces
   > Translate all traffic from 172.16.0.0/24
   > Create a pool of 100.0.0.1 to 100.0.0.2 from the 100.0.0.0/24 subnet

2. Ping google.com from PC1 and PC2. Then, ping it from PC3.
   What happens to PC3's ping?

3. Clear the NAT translations and remove the current NAT configuration.
   Switch the configuration to PAT using R1's public IP address.

4. Ping google.com from each PC. Do the pings work?
   Examine the NAT translations on R1.

Configuring Dynamic NAT

Dynamic NAT maps private IPv4 addresses to public addresses drawn from a NAT pool. The steps and syntax to configure dynamic NAT are as follows:

1) Specify the inside interface:

```
Router(config)# interface type number
Router(config-if)# ip nat inside
```

2) Specify the outside interface:

```
Router(config)# interface type number
Router(config-if)# ip nat outside
```

3) Define a standard access list permitting addresses that are to be translated:

```
Router(config)# access-list access-list-number source source-wildcard
```

4) Define a pool of global addresses to be allocated:

```
Router(config)# ip nat pool name start-ip end-ip {netmask | prefix-length
prefix-length}
```

5) Bind the pool of addresses to the access list:

```
Router(config)# ip nat inside source list access-list-number pool name
```

```
R1(config)#int g0/1
R1(config-if)#ip nat inside
R1(config-if)#int g0/0
R1(config-if)#ip nat outside
R1(config-if)#
R1(config-if)#access-list 1 permit 172.16.0.0 0.0.0.255
R1(config)#ip nat pool POOL1 100.0.0.1 100.0.0.2 netmask 255.255.255.0
R1(config)#
R1(config)#ip nat inside source list 1 pool POOL1
R1(config)#
```

NAT Overload (PAT) configuration:

Same as dynamic NAT, but adding overload to the mapping command:

```
(config)#ip nat inside source list [ACL-id] pool [pool-name] overload
```

```
R1(config)#ip nat inside source list 1 pool POOL1 overload
```

1) Specify the inside interface:

```
Router(config)# interface type number
Router(config-if)# ip nat inside
```

2) Specify the outside interface:

```
Router(config)# interface type number
Router(config-if)# ip nat outside
```

3) Define a standard access list permitting addresses that are to be translated:

```
Router(config)# access-list access-list-number source source-wildcard
```

4) Define a pool of global addresses to be allocated:

```
Router(config)# ip nat pool name start-ip end-ip {netmask | prefix-length
prefix-length}
```

5) Bind the pool of addresses to the access list:

```
Router(config)# ip nat inside source list access-list-number pool name
```

4.2 NTP

NTP static client mode is enabled on a Cisco router when you issue the ntp server command from global configuration mode.

4.4 Explain the function of SNMP in network operations

4.4 Explain the function of SNMP in network operations

SNMP (Simple Network Management Protocol) is used to monitor the status of devices, make configuration changes, etc. In a network, the SNMP server (that manages the clients) is called the NMS (Network Management Station).

Information in devices are gathered as variables in a **MIB** (Management Information Base), which uses a tree-like structure and its variables are identified by Object ID (OID) that could be either numbers or names.

There are 3 main operations used in SNMP:

- Managed devices notify the NMS of events.
- The NMS ask the managed devices for status update.
- The NMS tell the managed devices to change their configurations.

There are 3 main types of SNMP messages:

Message class	Description	Messages
Read	Sent by NMS to read info of managed devices.	Get, GetNext, GetBulk
Write	Sent by NMS to change info on manges devices.	Set
Notification	Sent by managed devices to alert NMS of events.	Trap, Inform
Response	Sent in response of a previous message.	Response

SNMP Agent uses UDP port 161, SNMP Manager uses UDP port 162.
There are 3 versions of SNMP, only version 3 uses authentication.

Which three SNMP messages are sent from an SNMP agent to an SNMP manager? (Choose three.)

- [ ] GetRequest
- [ ] GetNextRequest
- [x] InformRequest
- [x] Response
- [ ] SetRequest
- [x] Trap

4.5 Syslog

There are **8 syslog levels from 0 to 7**, with corresponding names:
0.**E**mergency; 1.**A**lert; 2.**C**ritical; 3.**E**rror; 4.**W**arning; 5.**N**otice; 6.**I**nformational; 7.**D**ebugging.

Syslog could be saved/shown in 4 places, with [min-severity] is the minimum of the severities of events that will be logged, [min-severity] could be in number (0 to 7) or in names (Emergency–Debugging):

- Console line: Display Syslog in the CLI when connected to the console line. By default, all messages (level 0 to 7) are displayed. Configured by:

  (config)#logging console [min-severity]

- VTY lines: Display Syslog in the CLI when connected by Telnet/SSH. Disabled by default. Configured by:

  (config)#logging monitor [min-severity]

- Buffer: Syslog saved to RAM. By default, all messages (level 0 to 7) are saved.

  View by **#show logging**.

  Configured by:

  (config)#logging buffered [buffer-size] [min-severity]

- External server: Send syslog to external servers. Syslog servers listen for messages on **UDP port 514**. Configured by:

  (config)#logging (host) [server-ip]

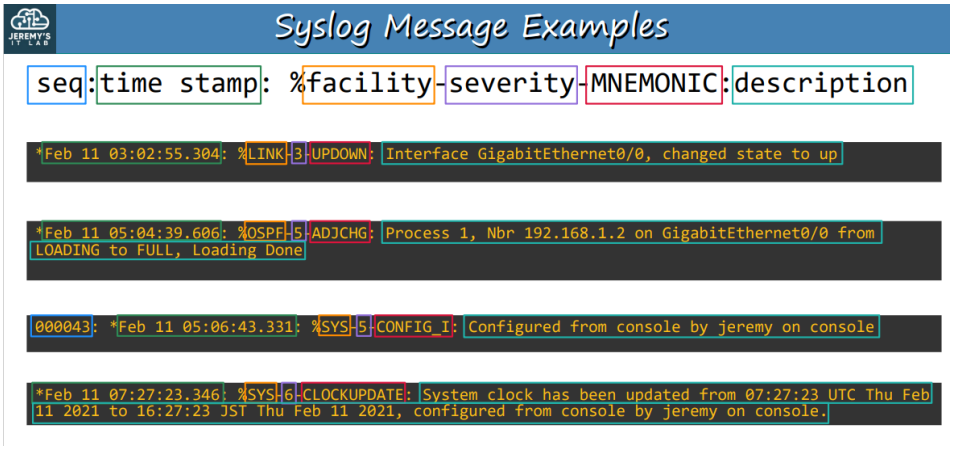
  Set min severity by **(config)#logging trap [min-severity]**

We can modify log message format by:

- Enable timestamp: **(config)#service timestamp**
- Enable sequence number: **(config)#service sequence-numbers**

SYSLOG SEVERITY LEVELS

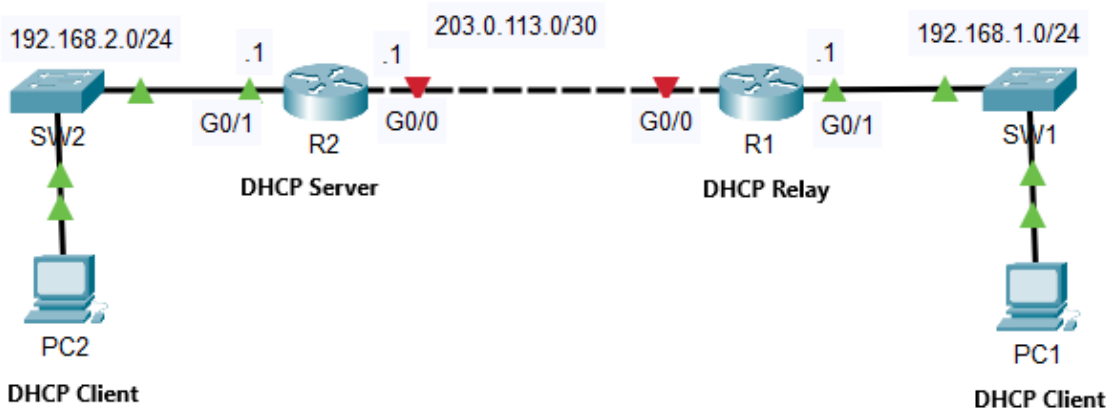
NUMERAL	KEYWORD	MNEMONIC	DESCRIPTION	CATEGORY
0	Emergency	**E**very	System unusable	Severe
1	Alert	**A**wesome	Immediate action required	
2	Critical	**C**isco	Critical Event (Highest of 3)	Impactful
3	Error	**E**ngineer	Error Event (Middle of 3)	
4	Warning	**W**ill	Warning Event (Lowest of 3)	
5	Notification	**N**eed	Normal, More Important	Normal
6	Informational	**I**ce Cream	Normal, Less Important	
7	Debug	**D**aily	Requested by User Debug	



4.6 Configure and verify DHCP client and relay

- DHCP Server Configuration:
  - Range of addresses that will not be given to clients:
    (config)#ip dhcp excluded-address [start-ip] [end-ip]
  - Create a DHCP pool and enter DHCP config mode: **(config)#ip dhcp pool [pool-name]**
  - Specify the subnet of addresses to be assigned to clients:
    (dhcp-config)#network [network-ip] [/prefix, or network-mask]
  - Set up DNS server: **(dhcp-config)#dns-server [ip-address]**
  - Set up the domain name of the network: **(dhcp-config)#domain-name [domain-name]**
  - Specify the default gateway: **(dhcp-config)#default-router [gateway-ip]**
  - Specify the lease time:
    (dhcp-config)#lease [days] [hours] [minutes] or **(dhcp-config)#lease infinite**
  - Show all DHCP binding: **#show ip dhcp binding**
- DHCP Relay Agent Configuration:
  - Enter the configuration mode of the interface connected to the subnet of the client devies:
    (config)#interface [interface-id]
  - Configure the IP address of the DHCP server:
    (config-if)#ip helper-address [DHCP-server-ip]
- DHCP client configuration:
  - On Cisco devices: **(config-if)#ip address dhcp**
  - On Windows: **>ipconfig /release** and **>ipconfig /renew**.

Config Router2 to be a DHCP Server and Router1 to act as a DHCP Relay Agent



1. Configure the following DHCP pools on R2:
POOL1: 192.168.1.0/24 (reserve .1 to .10)
DNS 8.8.8.8
Domain: jeremysitlab.com
Default Gateway: R1
POOL2: 192.168.2.0/24 (reserve .1 to .10)
DNS 8.8.8.8
Domain: jeremysitlab.com
Default Gateway: R2
POOL3: 203.0.113.0/30 (reserve .1)

2. Configure R1's G0/0 interface as a DHCP client.
What IP address did it configure?

3. Configure R1 as a DHCP relay agent for the 192.168.1.0/24 subnet.

4. Use the CLI of PC1 and PC2 to make them request an IP address from their DHCP server.

`R2(config)#ip dhcp excluded-address 192.168.1.1 192.168.1.10`
`R2(config)#ip dhcp excluded-address 192.168.2.1 192.168.2.10`
`R2(config)#ip dhcp excluded-address 203.0.113.1`
	The IP range that a DHCP Server should not assign to DHCP Clients. Notice this command is configured under global configuration mode

`R2(config)#ip dhcp pool POOL1`
`R2(dhcp-config)#network 192.168.1.0 255.255.255.0`
`R2(dhcp-config)#dns-server 8.8.8.8`
`R2(dhcp-config)#domain-name jeremysitlab.com`
`R2(dhcp-config)#default-router 192.168.1.1`
	- Create a DHCP Pool named POOL1
	- Specifies the subnet and mask of the DHCP address pool
	- Configure a Domain Name Server (DNS)
	- Configure a domain-name
	- Set the default gateway of the DHCP Clients

`R2(config)#ip dhcp pool POOL2`
`R2(dhcp-config)#network 192.168.2.0 255.255.255.0`
`R2(dhcp-config)#dns-server 8.8.8.8`
`R2(dhcp-config)#domain-name jeremysitlab.com`
`R2(dhcp-config)#default-router 192.168.2.1`
	- Create a DHCP Pool named POOL2
	- Specifies the subnet and mask of the DHCP address pool
	- Configure a Domain Name Server (DNS)
	- Configure a domain-name
	- Set the default gateway of the DHCP Clients

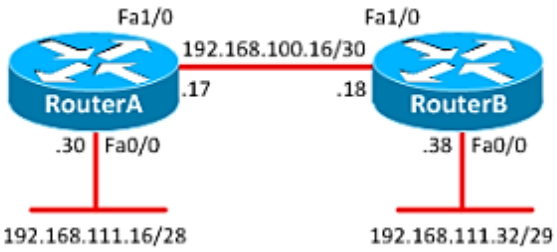
`R2(config)#ip dhcp pool POOL3` `R2(dhcp-config)#network 203.0.113.0 255.255.255.252`	- Create a DHCP Pool named POOL3 - Specifies the subnet and mask of the DHCP address pool

`R2#show run | include dhcp`
`ip dhcp excluded-address 192.168.1.1 192.168.1.10`
`ip dhcp excluded-address 192.168.2.1 192.168.2.10`
`ip dhcp excluded-address 203.0.113.1`
`ip dhcp pool POOL1`
`ip dhcp pool POOL2`
`ip dhcp pool POOL3`
	Verify

`R1(config)#int g0/0`
`R1(config-if)#ip address dhcp`
`R1(config-if)#no shut`
 	
`R1(config-if)#`
`%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up`
 	
`%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up`
 	
`%DHCP-6-ADDRESS_ASSIGN: Interface GigabitEthernet0/0 assigned DHCP address 203.0.113.2, mask 255.255.255.252, hostname R1`
	Configure R1's G0/0 interface as a DHCP client.
	
	What IP address did it configure?
	
	203.0.113.2

`R1(config)#int g0/1` `R1(config-if)#ip helper-address 203.0.113.1`	Configure R1 as a DHCP relay agent for the 192.168.1.0/24 subnet.

`C:\>ipconfig /release`
 	
`IP Address.....: 0.0.0.0`
`Subnet Mask.....: 0.0.0.0`
`Default Gateway.....: 0.0.0.0`
`DNS Server.....: 0.0.0.0`
 	
`C:\>ipconfig /renew`
 	
`IP Address.....: 192.168.1.11`
`Subnet Mask.....: 255.255.255.0`
`Default Gateway.....: 192.168.1.1`
`DNS Server.....: 8.8.8.8`
	Use the CLI of PC1 and PC2 to make them request an IP address from their DHCP server.



You administer the network in the topology diagram.

RouterA is configured as a DHCP server. In an effort to centralize the administration of DHCP services, you have decided to configure RouterB as a DHCP relay agent rather than configure it as a DHCP server. You issue the **show running-config** command on RouterA and receive the following partial output:

```
RouterA#show running-config
<output omitted>
ip dhcp pool DHCP
   network 192.168.111.16 255.255.255.240
   default-router 192.168.111.30
   dns-server 192.168.111.67
<output omitted>
```

Which of the following commands should you issue on RouterB? *(Select the best answer.)*

- ○ A. **ip default-network 192.168.111.30**
- ○ B. **network 192.168.111.32 255.255.255.248**
- ○ C. **service dhcp**
- ○ D. **ip helper-address 192.168.100.18**
- ◉ **E. ip helper-address 192.168.111.30**

Configure a DHCP Server on a Cisco router:

Configuration	Description
Router(config)#ip dhcp pool CLIENTS	Create a DHCP Pool named CLIENTS
Router(dhcp-config)#network 10.1.1.0 /24	Specifies the subnet and mask of the DHCP address pool
Router(dhcp-config)#default-router 10.1.1.1	Set the default gateway of the DHCP Clients
Router(dhcp-config)#dns-server 10.1.1.1	Configure a Domain Name Server (DNS)
Router(dhcp-config)#domain-name.com	Configure a domain-name
Router(dhcp-config)#lease 0 12	Duration of the lease (the time during which a client computer can use an assigned IP address). The syntax is "**lease** {days[hours] [minutes] | infinite}". In this case the lease is 12 hours. The default is a one-day lease. Before the lease expires, the client typically needs to renew its address lease assignment with the server
Router(dhcp-config)#exit	
Router(config)# ip dhcp excluded-address 10.1.1.1 10.1.1.10	The IP range that a DHCP Server should not assign to DHCP Clients. Notice this command is configured under global configuration mode

Discover	Client → Server	Broadcast
Offer	Server → Client	Broadcast or Unicast
Request	Client → Server	Broadcast
Ack	Server → Client	Broadcast or Unicast

Day 58 Lab - Wireless LANs.pkt

4.7 QoS- Explain the forwarding per-hop behavior (PHB) for QoS, such as classification, marking, queuing, congestion, policing, and shaping

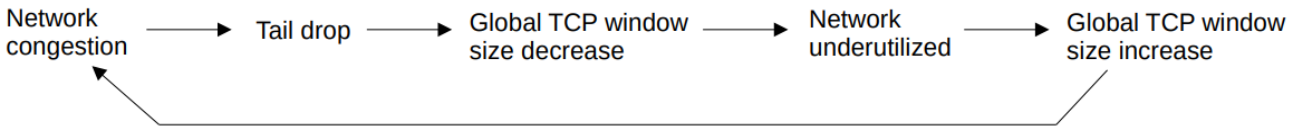
- QoS (Quality of Service) is a set of tools and protocols used to manage:
  - Bandwidth: QoS tools allow reservation of a certain amount of bandwidth for a certain type of traffic.
  - Delay: The amount of time it takes to go from source to destination (one-way) and also to go back (round-trip).
  - Jitter: The variation in one-way delay between packets sent by the same application.
  - Loss: the percent of packets sent that do not reach the destination.
- The following standard is recommended for interactive audio, such as phone call:
  - One-way delay $\leq$ 150 ms
  - Jitter $\leq$ 30 ms
  - Loss $\leq$ 1%

- Traffic msgs are placed in a Que if receiving device is busy processing/forwarding other msgs.
- By default, msgs are processed in a **FIFO (First In First Out)** manner.
  - So msgs will be sent in the order they're received.
- If the Que is full, new packets will be dropped (this is called **Tail Drop**)

- **Tail Drop** is harmful because can lead to **TCP global synchronization**.

- Review of the **TCP sliding window**:
  - Hosts using TCP use the 'sliding window' increase/decrease the rate at which they send traffic as needed.
  - When a packet is dropped it will be re-transmitted.
  - When a drop occurs, the sender will reduce the rate it sends traffic.
  - It will then gradually increase the rate again.

- When the queue fills up and **tail drop** occurs, all TCP hosts sending traffic will slow down the rate at which they send traffic.
- They will all then increase the rate at which they send traffic, which rapidly leads to more congestion, dropped packets, and the process repeats again.



Weighted Random Early Detection (WRED)

- A solution to prevent tail drop and TCP global synchronization is **Random Early Detection** (RED).
- When the amount of traffic in the queue reaches a certain threshold, the device will start randomly dropping packets from select TCP flows.
- Those TCP flows that dropped packets will reduce the rate at which traffic is sent, but you will avoid global TCP synchronization, in which ALL TCP flows reduce and then increase the rate of transmission at the same time in waves.
- In standard RED, all kinds of traffic are treated the same.
- An improved version, **Weighted Random Early Detection** (WRED), allows you to control which packets are dropped depending on the traffic class.

WRED is a congestion avoidance that drops lower-priority packets if network congestion is detected.

Which of the following statements best describe why WRED is useful for networks where the majority of traffic uses TCP? *(Select 2 choices.)*

- ☑ **A. TCP packets that are dropped must be retransmitted.**
- ☑ B. TCP packets have large header sizes.
- ☐ C. TCP packets must have priority over UDP packets.
- ☐ D. TCP packets cannot arrive out of sequence.
- ☐ **E. TCP sources reduce traffic flow when congestion occurs.**

Explanation:

Weighted random early detection (WRED) is useful for networks where the majority of traffic uses Transmission Control Protocol (TCP) because TCP packets that are dropped must be retransmitted. Additionally, TCP sources reduce traffic flow when congestion occurs, thereby further slowing down the network.

WRED is a congestion avoidance mechanism that addresses packet loss caused by tail drop, which occurs when new incoming packets are dropped because a router's queues are too full to accept them. Tail drop causes a problem called global TCP synchronization, whereby all of the TCP sources on a network reduce traffic flow during periods of congestion and then the TCP sources increase traffic flow when the congestion is reduced, which again causes congestion and dropped packets. When WRED is implemented, you can configure different tail drop thresholds for each IP precedence or Differentiated Services Code Point (DSCP) value so that lower-priority traffic is more likely to be dropped than higher-priority traffic, thereby avoiding global TCP synchronization.

Per-hop behaviors:

- Classification and Marking: organizes network traffic (packets) into traffic classes (categories). There are many methods of classifying traffic:
  - ACL: Traffic pass through the ACL obviously has higher priority than the denied traffic.
  - NBAR (Network Based Application Recognition): Look deep into Layer 7 to identify the kind of traffic.
  - PCP (Priority Code Point): Also known as CoS (class of service). This field is found in the 802.1q tag, can be used to identify traffic priority, but only applicable when there is a 802.1q tag (on trunk links). Since PCP uses 3 bits, there are 8 possible values, including:
    - 0. Best effort.
    - 3. Critical application (signal traffic of IP phones).
    - 5. Voice (voice traffic of IP phones).
  - DSCP (Differentiated Services Code Point): This field in the IP header could be used to identify traffic priority. There are some standard markings for this field:
    - DF (Default Forwarding): Best effort traffic. The **DSCP marking is 0**.
    - EF (Expedited Forwarding): low loss/latency/jitter traffic (usually voice). The **DSCP marking is 46**.
    - AF (Assured Forwarding): A set of 12 sub-values. The **DSCP marking is AFxy** where $x \in 1, 2, 3, 4$ with ascending order of priority, and $y \in 1, 2, 3$ with ascending order of drop precedence.
    - CS (Class selector): A set of 8 sub-values.
    - Recommended classes:
      - Voice traffic: **EF**
      - Interactive video: **AF4x**
      - Streaming video: **AF3x**
      - High priority data: **AF2x**
      - Best effort: **DF**
- Queuing and Congestion management: QoS allows multiple queues bases on traffic classification.

  A standard queue uses a FIFO scheduler – the packet that arrives first in a given queue is sent first. Weighted round-robin scheduler – e.g. Class-Based Weighted Fair Queuing (**CBWFQ**) is commonly used to prioritise particular queues – in a given time period, each queue is guaranteed a minimum share of bandwidth.

  LLQ (Low Latency Queuing) designates one queue as strict priority queue, which always have priority over other queues. However, it might not leave room for any other traffic.
- Shaping and Policing: **Shaping buffers traffic** in a queue if the traffic rate goes over a configured rate, while **Policing drops traffic** is the traffic rate goes over a configured rate.

4.8/5.3 SSH- Config SSH / Device Hardening

There are four main steps required to enable SSH support on a Cisco IOS router/switch:

When configuring SSH you have to set hostname, username/pw, ip domain-name, crypto key and transport input ssh.

There are four steps required to enable SSH support on a Cisco IOS router:

1. Configure the **hostname** command.
2. Configure the DNS domain.
3. Generate the SSH key.
4. Enable SSH transport support for the vty.

SSH also requires a Username/PW on the VTY lines.

1. Configure the router with a host name other than Router by issuing the **hostname** command.
2. Configure the router with a domain name by issuing the **ip domain-name** command.
3. Generate an RSA key pair for the router by issuing the **crypto key generate rsa** command.
4. Configure the VTY lines to use SSH by issuing the **transport input ssh** command from line configuration mode.

If you config SSH out of sequence of these 4 steps, you'll get an error message stating "you need to do previous step".

SSH Configuration

Secure Shell (SSH) is considered a security best practice because Telnet (port 23) uses insecure plaintext transmission of both the login and the data across the connection. SSH (port 22) is a more secure form of remote access:

- It requires a username and a password, both of which are encrypted during transmissions.
- The username and password can be authenticated using the local database method.
- It provides more accountability because the username is recorded when a user logs in.

Example 20-3 illustrates SSH and local database methods of remote access.

Connect

```
telnet ip-address
ssh -l username ip-address
ssh username@ip-address
```

Example 20-3 Configuring SSH Remote Access on a Switch

```
S1# show ip ssh
SSH Disabled-version 1.99
%Please create RSA keys to enable SSH (of at least 768 bits size) to enable SSH v2.
Authentication timeout: 120 secs; Authentication retries:3
S1# conf t
S1(config)# ip domain-name cisco.com
S1(config)# crypto key generate rsa
The name for the keys will be: S1.cisco.com
Choose the size of the key modulus in the range of 360 to 4096 for your
   General Purpose Keys. Choosing a key modulus greater than 512 may take
   a few minutes.

How many bits in the modulus [512]:1024
% Generating 1024 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 4 seconds)

*Mar 1 02:20:18.529: %SSH-5-ENABLED: SSH 1.99 has been enabled
S1(config)# line vty 0 15
S1(config-line)# login local
S1(config-line)# transport input ssh
S1(config-line)# username allanj secret 31daysCCNA
!The following commands are optional SSH configurations.
S1(config)# ip ssh version2
S1(config)# ip ssh authentication-retries 5
S1(config)# ip ssh time-out 60
S1(config)# end
S1# show ip ssh
SSH Enabled - version 2.0
Authentication timeout: 60 secs; Authentication retries: 5
S1#
```

The following steps occur in Example 20-3:

Step 1. Verify that the switch supports SSH using the **show ip ssh** command. If the command is not recognized, you know that SSH is not supported.

Step 2. Configure a DNS domain name with the **ip domain-name** global configuration command.

Step 3. Configure the switch using the **crypto key generate rsa** command to generate an RSA key pair and automatically enable SSH. When generating RSA keys, you are prompted to enter a modulus length. Cisco recommends a minimum modulus size of 1024 bits, as in Example 20-3.

NOTE: To remove the RSA key pair, use the **crypto key zeroize rsa** command. This disables the SSH service.

Step 4. Change the vty lines to use usernames, with either locally configured usernames or an authentication, authorization, and accounting (AAA) server. In Example 20-3, the **login local** vty subcommand defines the use of local usernames, replacing the **login** vty subcommand.

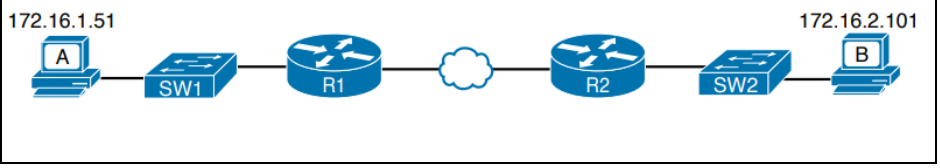
Step 5. Configure the switch to accept only SSH connections with the **transport input** ssh vty subcommand. (The default is **transport input telnet**.)

Step 6. Add one or more **username password** global configuration commands to configure username/password pairs.

Step 7. If desired, modify the default SSH configuration to change the SSH version to 2.0, the number of authentication tries, and the timeout, as in Example 20-3.

Step 8. Verify your SSH parameters by using the **show ip ssh** command.

Device Hardening Config 1

In this lab, you'll learn how to harden the routers on your network using a number of different features including proper secured passwords, limited SSH access (Management server 192.168.1.100), banners, and disabling CDP. This begins with all devices already configured with IP addresses and routing.	

`R2#conf t`	Move into global configuration mode on the router you want to harden (R2 in this example, but you should do all routers)
`R2(config)#`**`enable secret cs`**	Configure an enable password in the most secure method supported with a password of **cs**
`R2(config)#`**`username cisco secret cs`**	Configure a user named **cisco** with a secret password of **cs**
`R2(config)#`**`ip domain-name testing.com`**	Configure a domain-name of **testing.com**
`R2(config)#`**`crypto key generate rsa`**	Generate an RSA key pair using a 2048 bit key.
`R2(config)#`**`ip ssh version 2`**	Enable SSH version 2
`R2(config)#`**`access-list 1 permit 192.168.1.100`**	Configure a standard ACL 1 permitting the management server IP address 192.168.1.100
`R2(config)#`**`line console 0`** `R2(config-line)#`**`login local`**	Move into console line configuration mode and configure it to require login using the local user database.
`R2(config-line)#`**`line vty 0 4`** `R2(config-line)#`**`login local`**	Move into terminal line configuration mode and also configure it to require a login using the local user database
`R2(config-line)#`**`transport input ssh`**	Configure the lines to only support SSH logins
`R2(config-line)#`**`access-class 1 in`**	Configure the lines to only allow access from the address specified in ACL 1.

`R2(config)#`**`banner motd #`**
`Enter TEXT message. End with the character '#'.`
 	
`UNAUTHORIZED ACCESS TO THIS DEVICE IS PROHIBITED`
 	
`All activities performed on this device are logged and monitored`**`#`**
	Exit into global configuration mode and configure the following Message of the Day (MOTD) banner:
	
	UNAUTHORIZED ACCESS TO THIS DEVICE IS PROHIBITED
	
	All activities performed on this device are logged and monitored.

`R2(config)#`**`banner login #`**	Configure the following login banner

Enter TEXT message. End with the character '#'. LOGIN IS REQUIRED - UNAUTHORIZED ATTEMPTS TO ACCESS THIS DEVICE WILL BE LOGGED AND REPORTED**#**	LOGIN IS REQUIRED - UNAUTHORIZED ATTEMPTS TO ACCESS THIS DEVICE WILL BE LOGGED AND REPORTED
R2(config)#**banner exec #** Enter TEXT message. End with the character '#'. BY LOGGING INTO THIS SYSTEM, YOU ACKNOWLEDGE THAT YOU ARE AUTHORIZED If any changes are made to the configuration, ensure documentation has been provided on appropriate systems**#**	Configure the following EXEC banner BY LOGGING INTO THIS SYSTEM, YOU ACKNOWLEDGE THAT YOU ARE AUTHORIZED If any changes are made to the configuration, ensure documentation has been provided on appropriate systems
R2(config)#**no cdp run**	Disable CDP globally.

4.9 Compare TFTP/FTP

FTP	TFTP
TCP port 20 for Data, TCP port 21 for Control	UDP port 69
Connection-oriented	Connectionless but is embedded w/ a basic form of 'connection'
Clients can perform various actions (like a file directory)	Can only copy files to or from the server
Username / Password authentication	No authentication
More complex	Simpler
The FTP Data connection (port 20) has two modes: **Active mode**, the server initiates the data connection. **Passive mode**, the client initiates the data connection. *This is often necessary when the client is behind a firewall.*	TFTP uses 'lock-step' communication. The client and server alternately send a message and then wait for a reply. (+retransmissions are sent as needed)

Where 'connection' means that both party's have initiated some kind of TCP handshake before file transfer begins.

Quiz

- Use the **copy tftp: flash:** cmd to transfer a file from an external TFTP server to the local device's flash storage.
- If a device is behind a firewall and wants to connect to an external FTP server, the device should use **FTP passive mode**.
- **NVRAM** store the startup-config of a device running Cisco IOS.

5. Security Fundamentals

Common Attacks

- DoS (denial-of-service) attacks
  - → target the availability of a system so users can't access it
- Spoofing attacks
  - → involve using fraudulent source IP/MAC addresses
- Reflection/amplification attacks
  - → involve spoofing a source IP address to cause a reflector to send lots of traffic to the target
- Man-in-the-middle attacks
  - → an attacker intercepts traffic between the source and destination to eavesdrop and/or modify the traffic
- Reconnaissance attacks
  - → used to gather information on the target to perform future attacks
- Malware
  - → malicious software such as viruses, worms, and trojan horses that infect a system
- Social engineering attacks
  - → attacks that use psychological manipulation to target people and make them reveal info or perform an action
- Password-related attacks
  - → attacks such as dictionary attacks and brute force attacks, used to guess the target's password

Spoofing attacks

- To **spoof** an address is to use a fake source address (IP or MAC address).
- Numerous attacks involve spoofing, it's not a single kind of attack.
- An example is a **DHCP exhaustion** attack.
- An attacker uses spoofed MAC addresses to flood DHCP Discover messages.
- The target server's DHCP pool becomes full, resulting in a denial-of-service to other devices.

ARP Poisoning/Spoofing

In an Address Resolution Protocol (ARP) poisoning attack, which is also known as an ARP spoofing attack, the attacker sends a gratuitous ARP (GARP) message to a host. The GARP message associates the attacker's MAC address with the Internet Protocol (IP) address of a valid host on the network. Subsequently, traffic sent to the valid host address will go through the attacker's computer rather than directly to the intended recipient. Implementing Dynamic ARP Inspection (DAI) can help mitigate ARP poisoning attacks.

Reflection/Amplification attacks

- In a **reflection** attack, the attacker sends traffic to a reflector, and spoofs the source address of its packets using the target's IP address.
- The reflector (ie. a DNS server) sends the reply to the target's IP address.
- If the amount of traffic sent to the target is large enough, this can result in a denial-of-service.

- A reflection attack becomes an **amplification** attack when the amount of traffic sent by the attacker is small, but it triggers a large amount of traffic to be sent from the reflector to the target. 5.6.7.8

Which two attack categories does the smurf attack belong to? (Choose two.)

- [x] amplification
- [ ] brute-force
- [ ] man-in-the-middle
- [ ] exfiltration
- [x] reflection

Address Spoofing Attack Summary

As you work through the various types of address spoofing attacks, remember that the attacker's goal is to disguise his identity and fool other systems in a malicious way. Use Table 4-2 to review the concepts and characteristics of each attack type.

Table 4-2 Summary of Address Spoofing Attacks

Goal	DoS/DDoS	Reflection	Amplification	Man-in-the-Middle
Exhaust a system service or resource; crash the target system	Yes	No	No	No
Trick an unwitting accomplice host to send traffic to target	No	Yes	Yes	No
Eavesdrop on traffic	No	No	No	Yes
Modify traffic passing through	No	No	No	Yes

Malware

- Malware (malicious software) refers to a variety of harmful programs that can infect a computer.
- **Viruses** infect other software (a 'host program'). The virus spreads as the software is shared by users. Typically they corrupt or modify files on the target computer.
- **Worms** do not require a host program. They are standalone malware and they are able to spread on their own, without user interaction. The spread of worms can congest the network, but the 'payload' of a worm can cause additional harm to target devices.
- **Trojan Horses** are harmful software that is disguised as legitimate software. They are spread through user interaction such as opening email attachments, or downloading a file from the Internet.

The above malware types can exploit various vulnerabilities to threaten any of the CIA of the target device.
*there are many types of malware!

VLAN Hopping

VLAN Hopping: By altering the VLAN ID on packets encapsulated for trunking, an attacking device can send or receive packets on various VLANs, bypassing Layer 3 security measures. VLAN hopping can be accomplished by switch spoofing or double tagging.

In this attack, the attacking computer generates frames with two 802.1Q tags. The first tag matches the native VLAN of the trunk port (VLAN 10 in this case), and the second matches the VLAN of a host it wants to attack (VLAN 20).

When the packet from the attacker reaches Switch A, Switch A only sees the first VLAN 10 and it matches with its native VLAN 10 so this VLAN tag is removed. Switch A forwards the frame out all links with the same native VLAN 10. Switch B receives the frame with an tag of VLAN 20 so it removes this tag and forwards out to the Victim computer.

Note: This attack only works if the trunk (between two switches) has the same native VLAN as the attacker. In other words, this attack is only successful if the attacker belongs to the native VLAN of the trunk link. Another important point is, this attack is strictly one way as it is impossible to encapsulate the return packet.

To mitigate this type of attack, we can use VLAN access control lists (VACLs, which applies to all traffic within a VLAN. We can use VACL to drop attacker traffic to specific victims/servers); or implement Private VLANs; or keep the native VLAN of all trunk ports different from user VLANs.

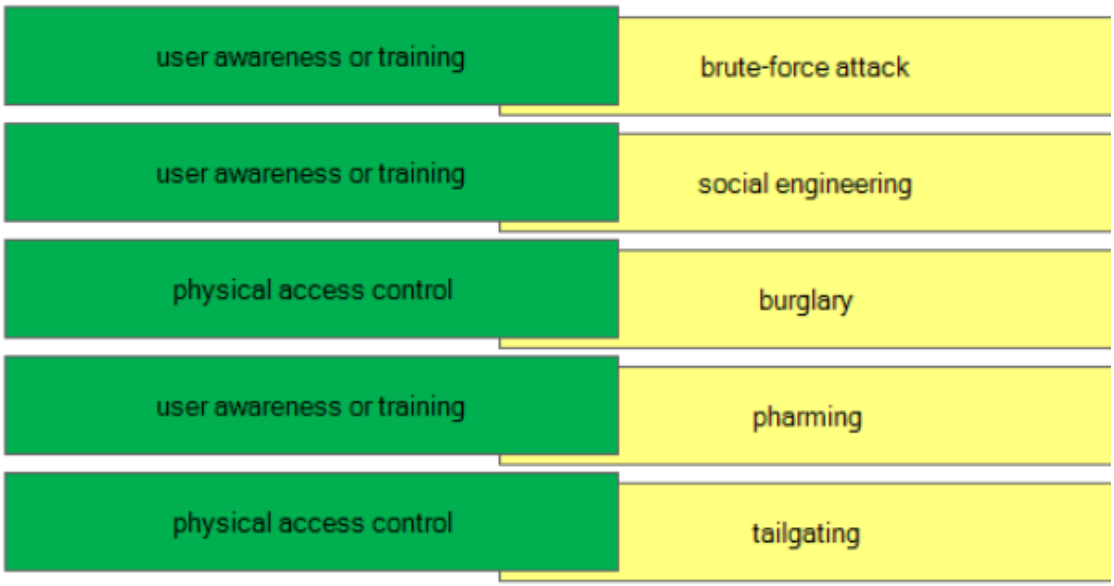
In a virtual local area network (VLAN) hopping attack, an attacker attempts to inject packets into other VLANs by accessing the VLAN trunk and double-tagging 802.1Q frames. A successful VLAN hopping attack enables an attacker to send traffic to other VLANs without the use of a router. You can prevent VLAN hopping by disabling Dynamic Trunking Protocol (DTP) on trunk ports, by changing the native VLAN, and by configuring user-facing ports as access ports.

Summary of Human Security Vulnerabilities:

Social engineering	Exploits human trust and social behavior
Phishing	Disguises a malicious invitation as something legitimate
Spear phishing	Targets group of similar users
Whaling	Targets high-profile individuals
Vishing	Uses voice calls
Smishing	Uses SMS text messages
Pharming	Uses legitimate services to send users to a compromised site
Watering hole	Targets specific victims who visit a compromised site

Reference: CCNA 200-301 Official Cert Guide Volume 2

Tailgating attacks involve entering restricted, secured areas by simply walking in behind an authorized person as they enter.
Often, the target will hold the door open for the attacker to be polite, assuming the attacker is also authorized to enter.



5.1 Define key security concepts (threats, vulnerabilities, exploits, and mitigation techniques)

Threat Landscape

- **Threat:** Any circumstance or event with the potential to cause harm to an asset in the form of destruction, disclosure, adverse modification of data, or denial of service (DoS). An example of a threat is malicious software that targets workstations.
- **Vulnerability:** A weakness that compromises either the security or the functionality of a system. Weak or easily guessed passwords are considered vulnerabilities.
- **Exploit:** A mechanism that uses a vulnerability to compromise the security or functionality of a system. An example of an exploit is malicious code that gains internal access. When a vulnerability is disclosed to the public, attackers often create a tool that implements an exploit for the vulnerability. If they release this tool or proof of concept code to the internet, other less-skilled attackers and hackers –the so called script kiddies– can then easily exploit the vulnerability.
- **Risk:** The likelihood that a particular threat using a specific attack will exploit a particular vulnerability of an asset that results in an undesirable consequence.
- **Mitigation techniques**: Methods and corrective actions to protect against threats and different exploits, such as implementing updates and patches, to reduce the possible impact and minimize risks.

5.2 Describe security program elements (user awareness, training, and physical access control)

- **User awareness** programs are designed to make employees aware of potential security threats and risks.
  → For example, a company might send out false phishing emails to make employees click a link and sign in with their login credentials.
  →Although the emails are harmless, employees who fall for the false emails will be informed that it is part of a user awareness program and they should be more careful about phishing emails.

- **User training** programs are more formal than user awareness programs.
  → For example, dedicated training sessions which educate users on the corporate security policies, how to create strong passwords, and how to avoid potential threats.

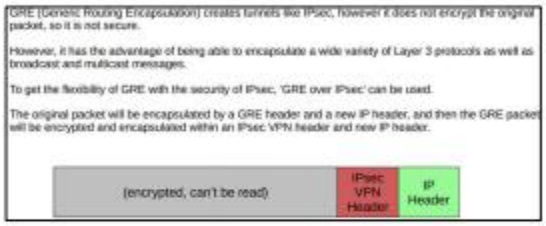
- **Physical access control** protects equipment and data from potential attackers by only allowing authorized users into protected areas such as network closets or data center floors.
  → Multifactor locks can protect access to restricted areas.
  → ie. a door that requires users to swipe a badge and scan their fingerprint to enter.
  → Permissions of the badge can easily be changed, for example permissions can be removed when an employee leaves the company.

5.5 Describe IPsec remote access and site-to-site VPNs

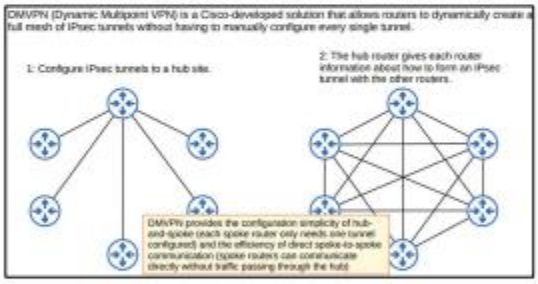
There are two main VPN technologies used:

1. **Site-to-site VPN using IP security (IPsec)**: a VPN tunnel is created between two devices by encapsulating the encrypted original IP packet with a new VPN header and new IP header (original packet is encrypted before being encapsulated). A tunnel is formed only between two tunnel endpoints (for example, the two routers connected to the Internet). All other devices in each site don't need to create a VPN for themselves. They can send unencrypted data to their site's router, which will encrypt it and forward it in the tunnel as described above.

    a. GRE (Generic Routing Encapsulation) is able to encapsulate a wide variety of L3 protocols, as well as broadcast and multicast messages, therefore GRE over IPsec can be used. The original packet is encapsulated with GRE header, IP header then the GRE packet is encapsulated with an IPsec VPN header and another IP heade



    b. DMVPN (Dynamic Multipoint VPN) is a Cisco-developed solution that allows routers to dynamically create a full mesh of IPsec tunnels without having to manually configure every single tunnel.



2. **Remote-access VPNs** are used to allow end devices (PCs, mobile phones) to access the company's internal resources securely over the Internet. Remote-access VPNs typically use TLS (Transport Layer Security) VPN client software is installed on end devices, these end devices can then form secure tunnels to the company's router/firewalls acting as a TLS server

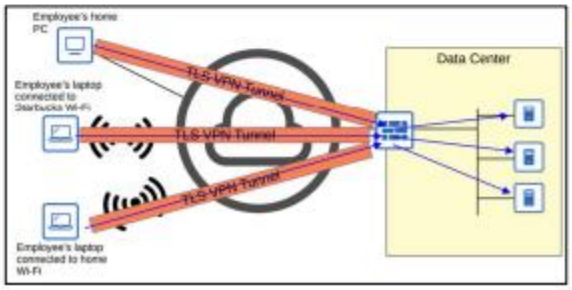


Table B-7 Comparisons of Site-to-Site and Remote Access IPsec VPNs

Attribute	Site-to-Site IPsec VPN	Remote Access IPsec VPN
Does the end-user device need VPN client software?	No	Yes
Devices supported by one VPN: one or many?	Many	One
Typical use: on-demand or permanent?	Permanent	On-demand
Does the VPN use IPsec tunnel mode?	Yes	No
Does the VPN use IPsec transport mode?	No	Yes

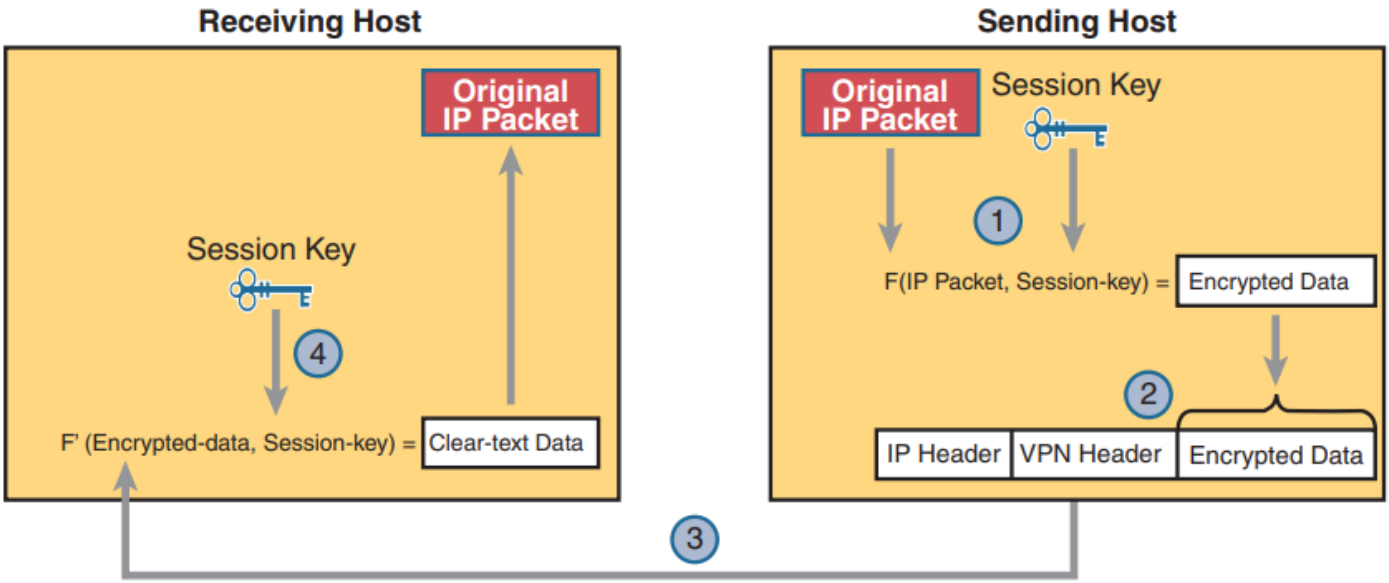


Figure 14-21 *Basic IPsec Encryption Process*

The four steps highlighted in the figure are as follows:

1. The sending VPN device (like the remote office router in Figure 14-21) feeds the original packet and the session key into the encryption formula, calculating the encrypted data.
2. The sending device encapsulates the encrypted data into a packet, which includes the new IP header and VPN header.
3. The sending device sends this new packet to the destination VPN device (FW1 back in Figure 14-21).
4. The receiving VPN device runs the corresponding decryption formula, using the encrypted data and session key—the same key value as was used on the sending VPN device—to decrypt the data.

Basic IPsec Encryption Process

1. The sending device encrypts the original packet and the session key.
2. The sending device encapsulates the encrypted data with new headers.
3. The sending device sends the encrypted packet to the destination device.
4. The destination device decrypts the data and the session key.

The sending device encrypts the original packet and the session key.

The sending device encapsulates the encrypted data with new headers.

The sending device sends the encrypted packet to the destination device.

The destination device decrypts the data and the session key.

IPsec Functionality Overview

IPsec provides the following network security services. (In general, the local security policy dictates the use of one or more of these services.)

- Data confidentiality–The IPsec sender can encrypt packets before transmitting them across a network.
- Data integrity–The IPsec receiver can authenticate packets sent by the IPsec sender to ensure that the data has not been altered during transmission.
- Data origin authentication–The IPsec receiver can authenticate the source of the sent IPsec packets. This service is dependent upon the data integrity service.
- Anti-replay–The IPsec receiver can detect and reject replayed packets.

Differences between Tunnel and Transport Mode

TUNNEL MODE	TRANSPORT MODE
Encrypts the entire packet, including the IP header. A new IP header is added to the packet after encryption.	Encrypts only the payload, while the original IP header is retained.
Tunnel monitoring uses the tunnel interface IP address.	Tunnel monitoring automatically uses the IP address of the physical interface (gateway interface IP address), and tunnel interface IP address is ignored.
Supports double encapsulation.	No support for double encapsulation.
This mode is commonly used for site-to-site communications.	This mode is commonly used for host-to-host communications.

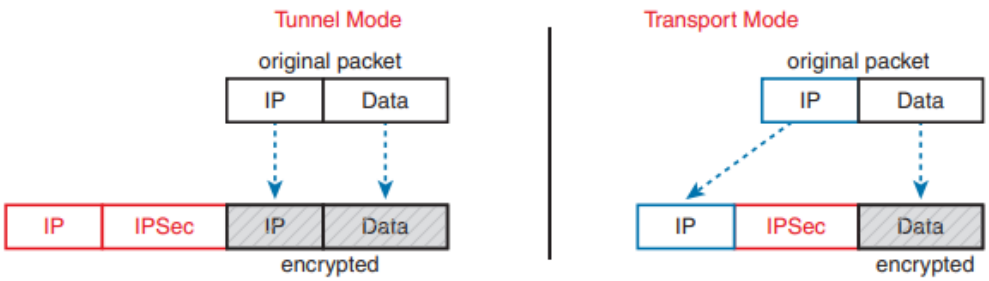
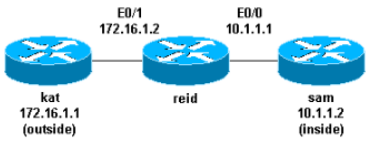


Figure B-6 *IPsec Tunnel and Transport Mode—What Is Encrypted*

5.6 Configure and verify access control lists



Standard ACLs

Standard ACLs are the oldest type of ACL. They date back to as early as Cisco IOS Software Release 8.3. Standard ACLs control traffic by the comparison of the source address of the IP packets to the addresses configured in the ACL.

This is the command syntax format of a standard ACL.

```
access-list <access-list-number> {permit|deny} {host|source source-wildcard|any}
```

In all software releases, the access-list-number can be anything from 1 to 99. In Cisco IOS Software Release 12.0.1, standard ACLs begin to use additional numbers (1300 to 1999). These additional numbers are referred to as expanded IP ACLs. Cisco IOS Software Release 11.2 added the ability to use list name in standard ACLs.

A source/source-wildcard setting of 0.0.0.0/255.255.255.255 can be specified as any . The wildcard can be omitted if it is all zeros. Therefore, host 10.1.1.2 0.0.0.0 is the same as host 10.1.1.2.

After the ACL is defined, it must be applied to the interface (inbound or outbound). In early software releases, out was the default when a keyword out or in was not specified. The direction must be specified in later software releases.

```
interface <interface-name>
 ip access-group number {in|out}
```

This is an example of the use of a standard ACL in order to block all traffic except that from source 10.1.1.x.

```
interface Ethernet0/0
 ip address 10.1.1.1 255.255.255.0
 ip access-group 1 in
!
access-list 1 permit 10.1.1.0 0.0.0.255
```

Extended ACLs

Extended ACLs were introduced in Cisco IOS Software Release 8.3. Extended ACLs control traffic by the comparison of the source and destination addresses of the IP packets to the addresses configured in the ACL.

This is the command syntax format of extended ACLs. Lines are wrapped here for space considerations.

IP

```
access-list access-list-number
     [dynamic dynamic-name [timeout minutes]]
     {deny|permit} protocol source source-wildcard destination destination-wildcard [precedence precedence]
     [tos tos] [log|log-input] [time-range time-range-name]
```

ICMP

```
access-list access-list-number
     [dynamic dynamic-name [timeout minutes]]
     {deny|permit} icmp source source-wildcard destination destination-wildcard
     [icmp-type [icmp-code] |icmp-message]
     [precedence precedence] [tos tos] [log|log-input]
     [time-range time-range-name]
```

TCP

```
access-list access-list-number
     [dynamic dynamic-name [timeout minutes]]
     {deny|permit} tcp source source-wildcard [operator [port]]
     destination destination-wildcard [operator [port]]
     [established] [precedence precedence] [tos tos]
     [log|log-input] [time-range time-range-name]
```

UDP

```
access-list access-list-number
     [dynamic dynamic-name [timeout minutes]]
     {deny|permit} udp source source-wildcard [operator [port]]
     destination destination-wildcard [operator [port]]
     [precedence precedence] [tos tos] [log|log-input]
     [time-range time-range-name]
```

In all software releases, the access-list-number can be 100 to 199. In Cisco IOS Software Release 12.0.1, extended ACLs begin to use additional numbers (2000 to 2699). These additional numbers are referred to as expanded IP ACLs. Cisco IOS Software Release 11.2 added the ability to use list *name* in extended ACLs.

The value of 0.0.0.0/255.255.255.255 can be specified as any. After the ACL is defined, it must be applied to the interface (inbound or outbound). In early software releases, out was the default when a keyword out or in was not specified. The direction must be specified in later software releases.

```
interface <interface-name>
 ip access-group {number|name} {in|out}
```

This extended ACL is used to permit traffic on the 10.1.1.x network (inside) and to receive ping responses from the outside while it prevents unsolicited pings from people outside, which permits all other traffic.

```
interface Ethernet0/1
 ip address 172.16.1.2 255.255.255.0
 ip access-group 101 in
!
access-list 101 deny icmp any 10.1.1.0 0.0.0.255 echo
access-list 101 permit ip any 10.1.1.0 0.0.0.255
```

5.8 Compare authentication, authorization, and accounting concepts

5.8 Compare authentication, authorization, and accounting concepts

AAA (triple-A) stands for Authentication, Authorization, and Accounting. It is a framework for controlling and monitor users of a computer system.

- **Authentication** is the process of verifying a user's identity. (i.e. Logging in).
- **Authorization** is the process of granting the user the appropriate access and permissions.
- **Accounting** is the process of recording the user's activities on the system.

Enterprises typically use AAA servers to provide AAA services. AAA servers usually support two AAA protocols:

- RADIUS: Open standard, uses UDP port 1812 and 1813.
- TACACS+: Cisco propriety, uses TCP port 49.

AAA

Configuring usernames and passwords on all your network devices is not very scalable. A better option is to use an external server to centralize and secure all username/password pairs. To address this issue, Cisco devices support the authentication, authorization, and accounting (AAA) framework to help secure device access.

Cisco devices support two AAA authentication protocols:

- Terminal Access Controller Access Control System Plus (TACACS+, pronounced as "tack-axe plus")
- Remote Authentication Dial-In User Service (RADIUS)

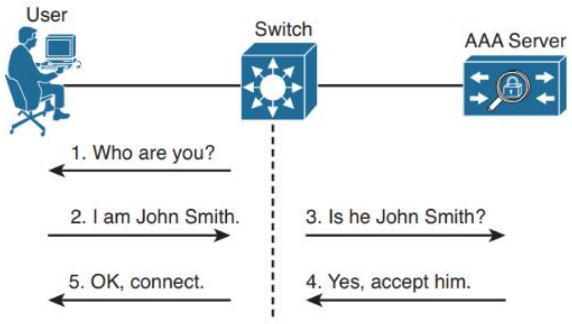
The choice of TACACS+ or RADIUS depends on the needs of the organization. For example, a large ISP might select RADIUS because it supports the detailed accounting required for billing users. An organization with various user groups might select TACACS+ because it requires authorization policies to be applied on a per-user or per-group basis. Table 20-1 compares TACACS+ and RADIUS.

Table 20-1 Comparison of TACACS+ and RADIUS

Feature	TACACS+	RADIUS
Most often used for	Network devices	Users
Transport protocol	TCP	UDP
Authentication port number(s)	49	1645, 1812
Protocol encrypts the password	Yes	Yes
Protocol encrypts entire packet	Yes	No
Supports function to authorize each user to a subset of CLI commands	Yes	No
Defined by	Cisco	RFC 2865

Both TACACS+ and RADIUS use a client/server model, where an authenticating device is the client talking to an AAA server. Figure 20-3 shows a simplified view of the process, where a user is attempting to connect to a switch for management purposes.

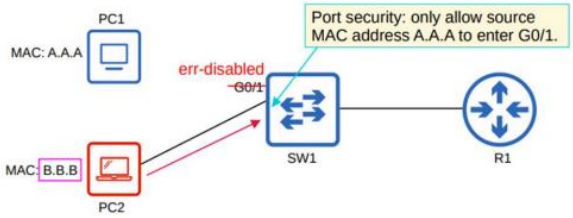
Figure 20-3 A Simplified View of AAA



5.7 L2 Security (Port Security, DHCP Snooping, DAI)

By Default, only ONE MAC addresses is allowed to connect to a port with Port Security enabled

- Port security allows network admins to control which devices are allowed to access the network.
- However, MAC address spoofing is a simple task.
  → It's easy to configure a device to send frames with a different source MAC address.
- Rather than manually specifying the MAC addresses allowed on each port, port security's ability to limit the number of MAC addresses allowed on an interface is more useful.
- Think of the DHCP starvation attack carried out in the Day 48 Lab video.
  → the attacker spoofed thousands of fake MAC addresses
  → the DHCP server assigned IP addresses to these fake MAC addresses, exhausting the DHCP pool
  → the switch's MAC address table can also become full due to such an attack
- Limiting the number of MAC addresses on an interface can protect against those attacks.
- Port security is a security feature of Cisco switches.
- It allows you to control which source MAC address(es) are allowed to enter the switchport.
- If an unauthorized source MAC address enters the port, an action will be taken.
  → The default action is to place the interface in an 'err-disabled' state.



Step 1. Configure the interface for static access mode by using the **switchport mode access** interface subcommand.

Step 2. Enable port security by using the **switchport port-security** interface subcommand.

Step 3. (Optional) Override the maximum number of allowed MAC addresses associated with the interface (1) by using the **switchport port-security maximum** *number* interface subcommand.

Step 4. (Optional) Override the default action when there is a security violation (shutdown) by using the **switchport port-security violation {protect | restrict | shutdown}** interface subcommand.

Step 5. (Optional) Predefine any allowed source MAC address(es) for this interface by using the **switchport port-security mac-address** *mac-address* command. Use the command multiple times to define more than one MAC address.

Step 6. (Optional) Instead of taking step 5, configure the interface to dynamically learn and configure the MAC addresses of currently connected hosts by configuring the **switchport port-security mac-address sticky** interface subcommand.

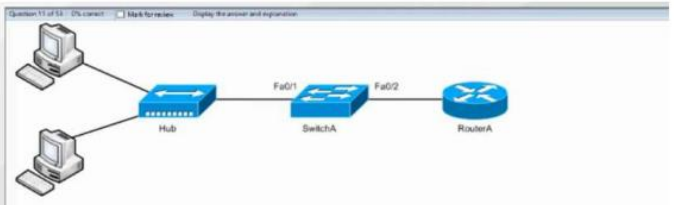
Table 20-2 Actions When Port Security Violation Occurs

Option on the switchport port-security violation Command	protect	restrict	shutdown
Discards offending traffic	Yes	Yes	Yes
Sends log and SNMP messages	No	Yes	Yes
Disables the interface, discarding all traffic	No	No	Yes

```
S1(config)# interface range fa 0/5 - fa 0/24
S1(config-if-range)# switchport mode access
S1(config-if-range)# switchport port-security
S1(config-if-range)# switchport port-security maximum 3
S1(config-if-range)# switchport port-security violation restrict
S1(config-if-range)# switchport port-security mac-address sticky
```

To verify port security configuration, use the more general **show port-security** command or the more specific **show port-security interface** *type number* command. Example 20-6 demonstrates the use of both commands. In the examples, notice that only one device is currently attached to an access port on S1.

You can use the **show interface** *type number* **status** or **show port-security interface** *type number* command to verify the current state of the port. To restore the port, you must first manually shut down the interface and then reactivate it, as in Example 20-8.



You administer the network shown in the exhibit. You issue the following commands on SwitchA:

```
SwitchA#configure terminal
SwitchA(config)#interface fastethernet 0/1
SwitchA(config-if)#switchport port-security
SwitchA(config-if)#switchport port-security violation protect
```

Which of the following statements are true regarding this configuration? (Select 2 choices.)

A. ☐ Static MAC addresses cannot be configured on the interface.
B. ☐ The interface will remain up but will drop all packets generated by the hosts connected to th
C. ☑ The interface will dynamically learn a MAC address.
D. ☐ Any previously learned MAC addresses will be removed.
E. ☑ One of the hosts will not be able to access the network.

Port security:

- Can only be enabled on static access or trunk ports by:
  (config-if)#switchport port-security
- Show port security status on an interface by:
  #show port-security interface [interface-name]
- When port security is violated, there are 3 violation mode, enable by **(config-if)# switchport port-security violation [shutdown/restrict/protect]**:
  - Shutdown: err-disabled the interface, generate a syslog and/or SNMP message, set the violation counter to 1.
  - Restrict: not disable the interface, but discard traffic from unauthorized MAC address. Generate syslog and/or SNMP message each time unauthorized MAC is detected. Violation counter is increased by 1 for each unauthorized frame.
  - Protect: the traffic from unauthorized MAC is discarded, without doing anything else.
- By default, secure MAC address will not age out (aging time: 0 mins), enable MAC aging by:
  (config-if)#switchport port-security aging time [time-in-minutes]
- Sticky MAC address could be enabled by:
  (config-if)#switchport port-security mac-address sticky
  Sticky MAC addresses will be added to the running config as:
  switchport port-security mac-address sticky [mac-address]

STICKY MAC ADDRESSES ARE BASICALLY A WAY TO CONFIG MAC ADDRESSES FOR PORT SECURITY, WITHOUT HAVING TO MANUALLY CONFIG EACH ONE.

RE-ENABLING AN INTERFACE (ERRDISABLE RECOVERY):

SW1(config)#**errdisable recovery cause psecure-violation**

SW1#show errdisable recovery

Every 5 mins(300s) by default, all err-disabled interfaces will be re-enabled if err-disable recovery has been enabled for the cause of the interface's disablement.

Port Security

Step 1. Use the **switchport mode access** or the **switchport mode trunk** interface subcommands, respectively, to make the switch interface either a static access or trunk interface.

Step 2. Use the **switchport port-security** interface subcommand to enable port security on the interface.

Step 3. (Optional) Use the **switchport port-security maximum** *number* interface subcommand to override the default maximum number of allowed MAC addresses associated with the interface (1).

Step 4. (Optional) Use the **switchport port-security violation {protect | restrict | shutdown}** interface subcommand to override the default action to take upon a security violation (shutdown).

Step 5. (Optional) Use the **switchport port-security mac-address** *mac-address* interface subcommand to predefine any allowed source MAC addresses for this interface. Use the command multiple times to define more than one MAC address.

Step 6. (Optional) Use the **switchport port-security mac-address sticky** interface subcommand to tell the switch to "sticky learn" dynamically learned MAC addresses.

To demonstrate how to configure this variety of the settings, Figure 6-2 and Example 6-1 show four examples of port security. Three ports operate as access ports, while port F0/4, connected to another switch, operates as a trunk.

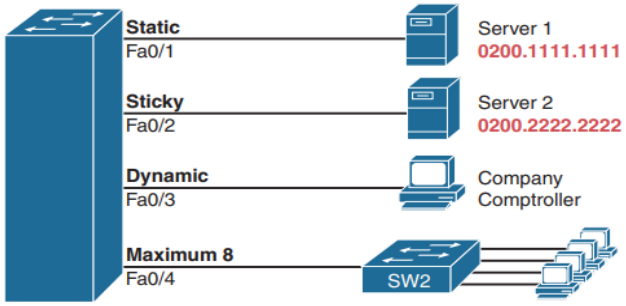


Figure 6-2 *Port Security Configuration Example*

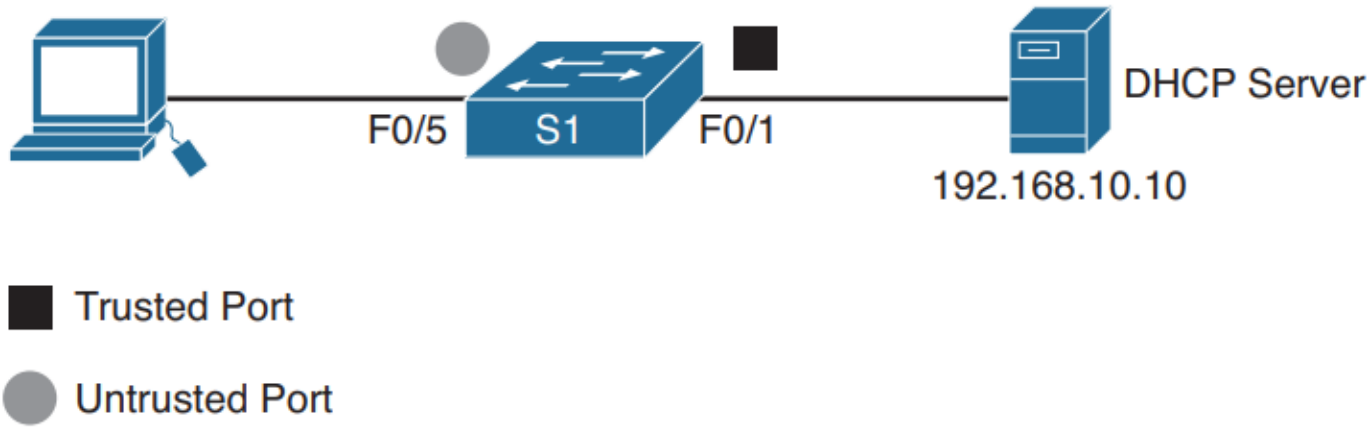
Table 20-2 Actions When Port Security Violation Occurs

Option on the switchport port-security violation Command	protect	restrict (+1 counter)	shutdown
Discards offending traffic	Yes	Yes	Yes
Sends log and SNMP messages	No	Yes	Yes
Disables the interface, discarding all traffic	No	No	Yes

```
Switch(config)#int f0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security mac-address 0200.1111.1111

Switch(config-if)#int f0/2
Switch(config-if)#switchport mode access
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security mac-address sticky

Switch(config-if)#int f0/3
Switch(config-if)#switchport mode access
Switch(config-if)#switchport port-security

Switch(config-if)#int f0/4
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security maximum 8
Switch(config-if)#switchport port-security violation restrict
```

DHCP Snooping / DAI Config

Sequence and Description	Command
Configure global DHCP snooping	`Switch(config)# ip dhcp snooping`
Enable DHCP snooping for the selected VLANs	`Switch(config)# ip dhcp snooping vlan {VLAN-ID | VLAN range}`
Enable Dynamic ARP Inspection (DAI) for the selected VLANs	`Switch(config)# ip arp inspection vlan {VLAN-ID | VLAN range}`
Configure trusted ports for DHCP Snooping and DAI. *(at least on 1 port). By default, all ports are untrusted.*	`Switch(config-if)# ip dhcp snooping trust` `Switch(config-if)# ip arp inspection trust`




```
SW1(config)#ip dhcp snooping
SW1(config)#ip dhcp snooping vlan 1
SW1(config)#ip arp inspection vlan 1
SW1(config)#int f0/1
SW1(config-if)#ip dhcp snooping trust
SW1(config-if)#ip arp inspection trust
```

```
SW1#show ip dhcp snooping

SW1#show ip dhcp snooping binding

SW1#show ip arp inspection interfaces

SW1#show ip arp inspection vlan 1
```

```
SW1#show ip dhcp snooping
Switch DHCP snooping is enabled
DHCP snooping is configured on following VLANs:
1
Insertion of option 82 is enabled
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Interface                    Trusted    Rate limit (pps)
-----------------------      -------    ----------------
FastEthernet0/1              yes        unlimited
```

```
SW1#show ip dhcp snooping binding
MacAddress          IpAddress        Lease(sec)  Type           VLAN  Interface
------------------  ---------------  ----------  -------------  ----  --------------------
00:30:F2:90:04:D1   192.168.10.1     86400       dhcp-snooping  1     FastEthernet0/5
Total number of bindings: 1
```

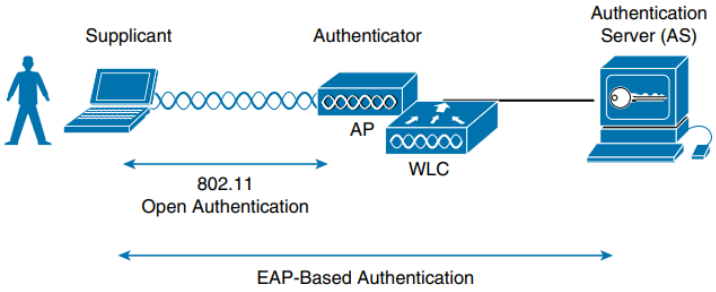
```
SW1#show ip arp inspection interfaces
Interface        Trust State     Rate(pps)    Burst Interval
---------------  -----------     ---------    --------------
Fa0/1            Trusted                15                 1
Fa0/2            Untrusted              15                 1
Fa0/3            Untrusted              15                 1
Fa0/4            Untrusted              15                 1
Fa0/5            Untrusted              15                 1
Fa0/6            Untrusted              15                 1
Fa0/7            Untrusted              15                 1
```

```
SW1#show ip arp inspection vlan 1

Source Mac Validation      : Disabled
Destination Mac Validation : Disabled
IP Address Validation      : Disabled

 Vlan     Configuration    Operation   ACL Match          Static ACL
 ----     -------------    ---------   ---------          ----------
    1     Enabled          Active

 Vlan     ACL Logging      DHCP Logging      Probe Logging
 ----     -----------      ------------      -------------
    1     Deny             Deny              Off
```

5.9 Wireless Security

The IEEE **802.1x** standard defines a client-server-based access control and authentication protocol that prevents unauthorized clients from connecting to a LAN through publicly accessible ports unless they are properly authenticated. The authentication server authenticates each client connected to a switch port before making available any services offered by the switch or the LAN. Until the client is authenticated, 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL), Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

- **Supplicant:** The client device that is requesting access
- **Authenticator:** The network device that provides access to the network (usually a wireless LAN controller [WLC])
- **Authentication server (AS):** The device that takes user or client credentials and permits or denies network access based on a user database and policies (usually a RADIUS server)



	WEP	***WPA-Personal***	***WPA-Enterprise***	***WPA2-Personal***	***WPA2-Enterprise***	***WPA3-Pers/Ent***
Encryption	uses RC4 w/ WEP 24-bit IV 40-bit key / 128-bit key w/ TKIP	uses RC4 w/ TKIP 48-bit IV 128-bit key	uses RC4 w/ TKIP 48-bit IV 128-bit key	*AES-CCMP* 48-bit IV 128-bit key	*AES-CCMP* 48-bit IV 128-bit key	*AES-GCMP*
Authentication	Optional Shared Key OAS, SKA	PSK (Pre-Shared Key) 802.1x, EAP (RADIUS)	802.1x Various EAP-Types	PSK (Pre-Shared Key) 802.1x, EAP, RSNA	802.1x Various EAP-Types	PSK (*Personal*) 802.1x, EAP (*Enterprise*)
Integrity	Checksum / CRC	MIC (64-bit)	MIC (64-bit)	CBC-MAC 128-bit key	CBC-MAC 128-bit key	GMAC (Galois Msg Auth Code)

Which of the following are used by WPA2 to provide MICs and encryption? *(Select 2 choices.)*

- ☐ A. GCMP
- ☑ **B. AES**
- ☐ C. RC4
- ☐ D. TKIP
- ☑ **E. CCMP**

L2 Security dropdown (WLC)

On the **Layer 2** tab of the **Security** tab, you can select one of the following Layer 2 wireless security features from the **Layer 2 Security** drop-down list box:

- **None**, which disables Layer 2 security and allows open authentication to the WLAN
- **WPA+WPA2**, which enables Layer 2 security by using Wi-Fi Protected Access (WPA) or the more secure WPA2
- **802.1X**, which enables Layer 2 security by using Extensible Authentication Protocol (EAP) authentication combined with a dynamic Wired Equivalent Privacy (WEP) key
- **Static WEP**, which enables Layer 2 security by using a static shared WEP key
- **Static WEP + 802.1X**, which enables Layer 2 security by using either a static shared WEP key or EAP authentication
- **CKIP**, which enables Layer 2 security by using the Cisco Key Integrity Protocol (CKIP)
- **None + EAP Passthrough**, which enables Layer 2 security by using open authentication combined with remote EAP authentication

L3 Security dropdown (WLC)

- **None**, which disables Layer 3 security no matter which Layer 2 security option is configured and regardless of whether you are configuring a WLAN or a Guest LAN
- **IPSec**, which enables Layer 3 security for WLANs by using Internet Protocol Security (IPSec)
- **VPN Pass-Through**, which enables Layer 3 security for WLANs by allowing a client to establish a connection with a specific virtual private network (VPN) server
- **Web Authentication**, which enables Layer 3 security for Guest LANs by prompting for a user name and password when a client connects to the network
- **Web Passthrough**, which enables direct access to the network for Guest LANs without prompting for a user name and password

There are two types of WLANs that you can configure by using the WLC GIJI: a **WLAN** and a **Guest LAN**. The **VPN Pass-Through** setting is only available when you are configuring a WLAN. So, when you are configuring a new wireless local area network (WLAN), you are most likely to configure the **VPN Pass-Through** setting by using the **Layer 3 Security drop-down** list box on the Layer 3 tab of the Cisco wireless LAN controller (WLC) graphical user interface (GIJI).

AAA Override feature on a Cisco wireless LAN controller (WLC)

The AAA Override feature on a Cisco wireless LAN controller (WLC) can be used to configure virtual local area network (VLAN) tagging, Quality of Service (QoS), and access control lists (ACLs) to individual clients based on Remote Authentication Dial-In User Service (RADIUS) attributes. When using the AAA Override feature, the access control server, such as Cisco Identity Services Engine (ISE), should be configured with the appropriate override properties. For example, you should configure the appropriate QoS-Level and VLAN-Tag attribute for each user.

5.10 Configure and verify WLAN within the GUI using WPA2 PSK

6. Automation and Programmability

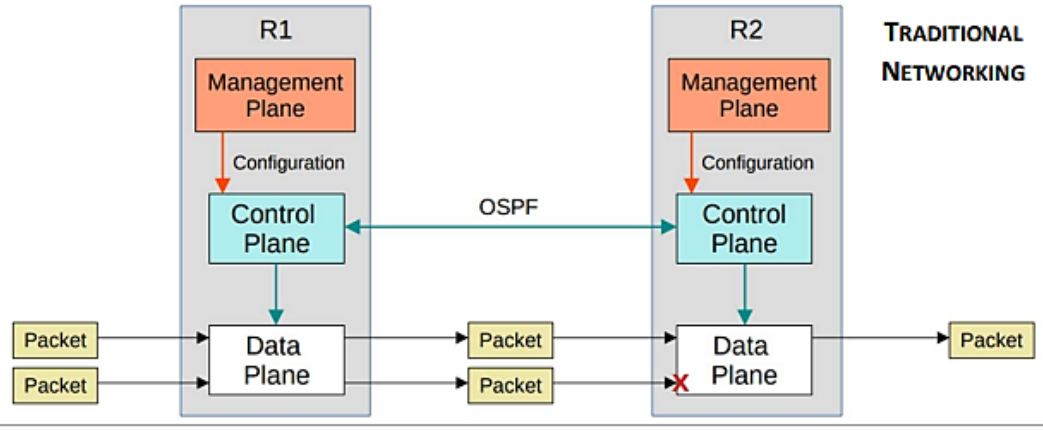
6.3 SDN- Describe controller-based, software defined architecture (overlay, underlay, and fabric)

SDN vs TRADITIONAL - DATA, CONTROL, AND MANAGEMENT PLANES

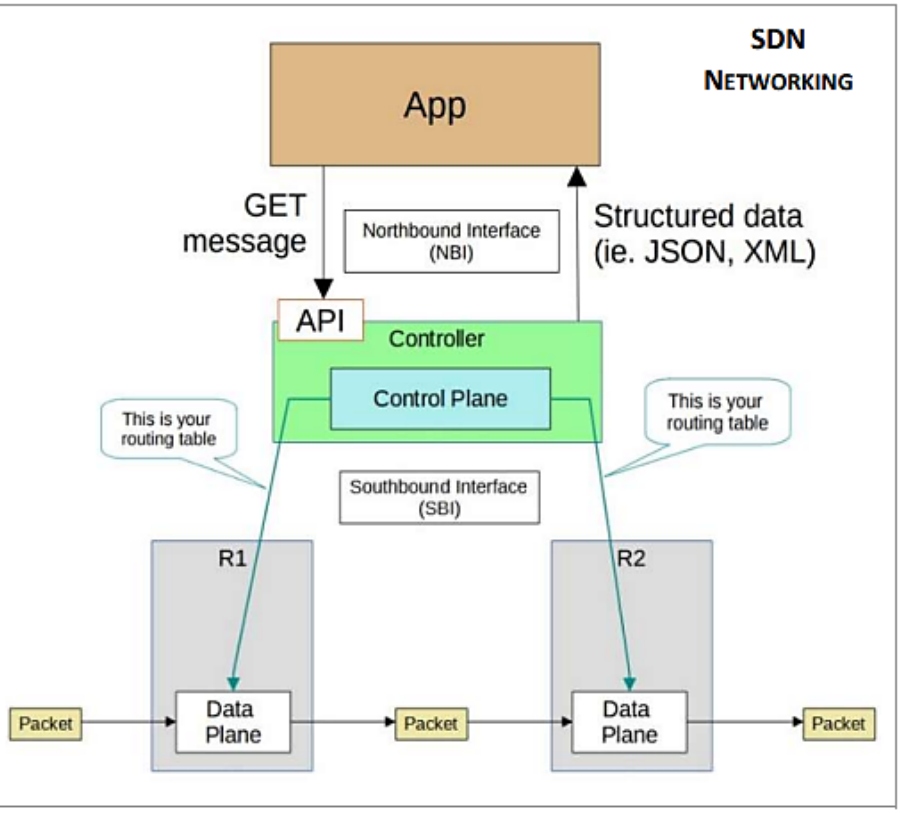
Everything that networking devices do can be categorized into three planes: Data Plane, Control Plane and Mgmt Plane.

- **Data Plane**: responsible for the switching of packets through the router. In short, it includes any action after receiving data (processing, encapsulating/decapsulating, matching destination MAC & IP addresses, forwarding, QoS, filtering with access-list)
- **Control Plane**: responsible for maintaining sessions and exchanging protocol information with other network devices. It consists of dynamic IP routing protocols (OSPF, EIGRP, BGP…), the RIB, routing updates, in addition to other protocols such as STP, ARP, ICMP, PIM, IGMP, LACP…
- **Management Plane**: is used to manage a device through its connection to the network. Examples of protocols processed in the management plane include Simple Network Management Protocol (SNMP), Telnet, File Transfer Protocol (FTP), Secure FTP, and Secure Shell (SSH). These management protocols are used for monitoring and for command-line interface (CLI) access.

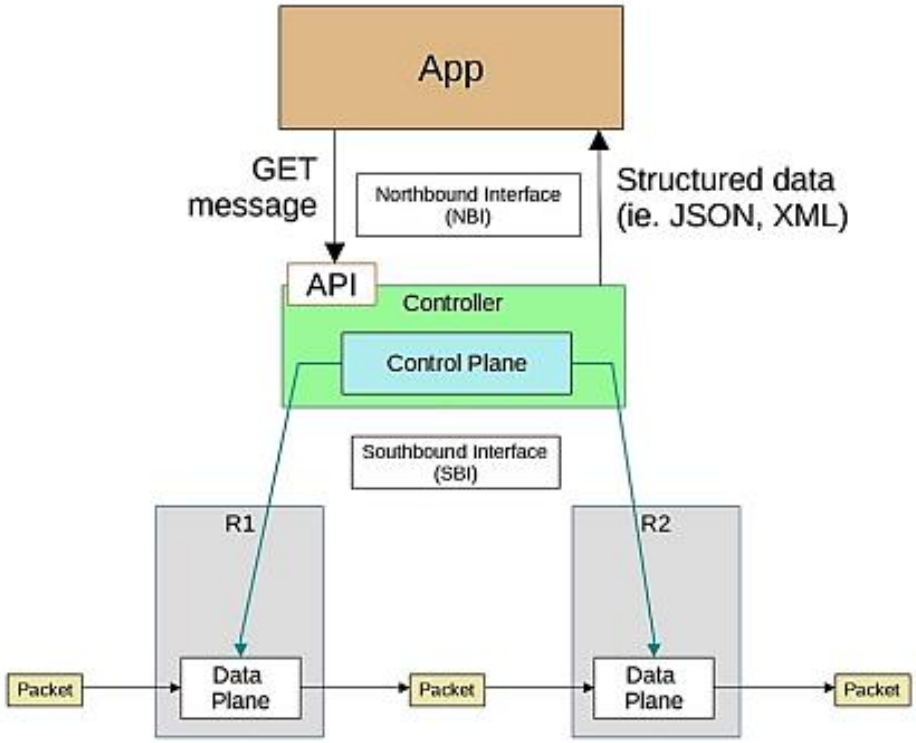
Most traditional devices use a distributed architecture, in which each control plane is resided in a networking device. Therefore, they need to communicate with each other via messages to work correctly.



- In contrast to distributed architecture, **centralized (or controller-based)** architectures centralizes the control of networking devices into one device, called SDN controller.
- As we took the **control planes** off networking devices but not **data planes** so we need a way to communicate with them. So we put a **southbound interface (SBI)** at the bottom of SDN controller for this task. *An **SBI** communicates with the devices via an application programming interface (API).*
- Now, in turn, the networking administrators and SDN applications want to control the controller! So the controller need a **northbound interface (NBI)** *to communicate with us*. The NBI applications included various network services, including network virtualization, dynamic virtual network provisioning, firewall monitoring, user identity management and access policy control.



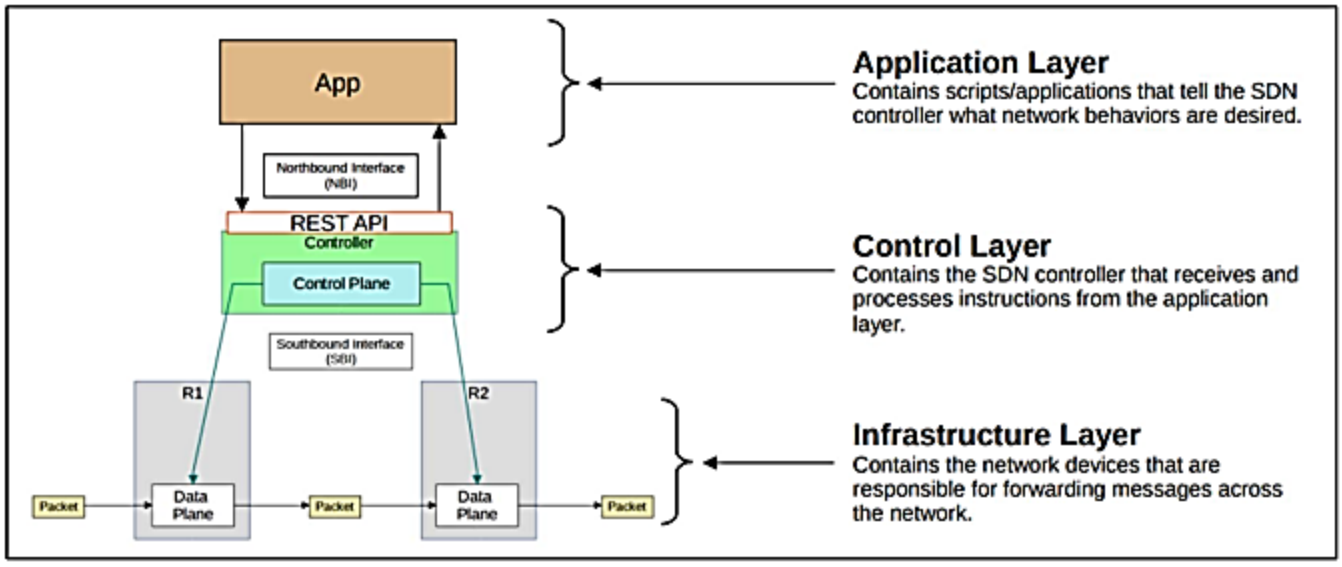
Main Components of SDN Architecture



- Using the **Southbound Interface (SBI)**, the controller communicates with the managed devices and gathers information about them:
  - The devices in the network
  - The topology (how the devices are connected together)
  - The available interfaces on each device
  - Their configurations
- It typically consists of a communication protocol and API (Application Programming Interface).
- APIs facilitate data exchanges between programs.
  - Data is exchanged between the controller and the network devices.
  - An API on the network devices allows the controller to access information on the devices, control their data plane tables, etc.
- Some examples of SBIs:
  - OpenFlow
  - Cisco OpFlex
  - Cisco onePK (Open Network Environment Platform Kit)
  - NETCONF

- The **Northbound Interface (NBI)** is what allows us to interact with the controller, access the data it gathers about the network, program it, and make changes in the network via the SBI.
- A REST API is used on the controller as an interface for apps to interact with it.
  - REST = Representational State Transfer
- Data is sent in a structured (serialized) format such as JSON or XML.
  - This makes it much easier for programs to use the data.

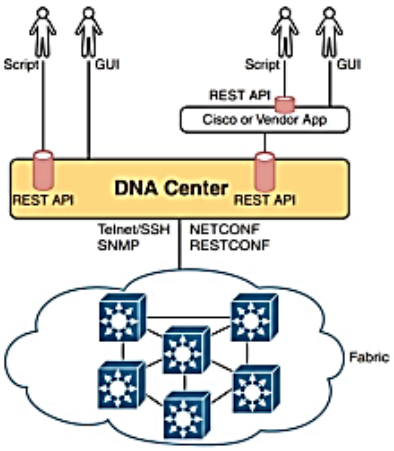
SDN Architecture - Layers



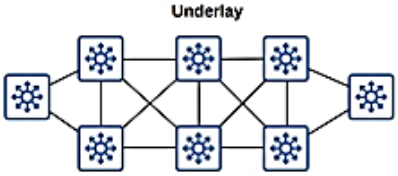
Cisco Software-Defined Access (SDA-Access)

- Cisco SD-Access is Cisco's SDN solution for automating campus LANs.
  - ACI (Application Centric Infrastructure) is their SDN solution for automating data center networks.
  - SD-WAN is their SDN solution for automating WANs.
- **Cisco DNA (Digital Network Architecture) Center** is the controller at the center of SD-Access.
- SDA uses a controller and application programming interfaces (APIs) to communicate via southbound interfaces (SBIs) with the network infrastructure, as shown in Figure 2-1. Cisco DNA Center is an example of a controller. SBIs include Telnet/SSH, SNMP, NETCONF, and RESTCONF.

Figure 2-1 SDA Architecture with DNA Center

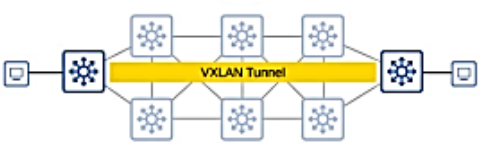


- The **underlay** is the underlying physical network of devices and connections (including wired and wireless) which provide IP connectivity (ie. using IS-IS).
  - Multilayer switches and their connections.
- The underlay's purpose is to support the VXLAN tunnels of the overlay.
- There are three different roles for switches in SD-Access:
  - *Edge nodes*: Connect to end hosts
  - *Border nodes*: Connect to devices outside of the SD-Access domain, ie. WAN routers.
  - *Control nodes*: Use LISP (Locator ID Separation Protocol) to perform various control plane functions.



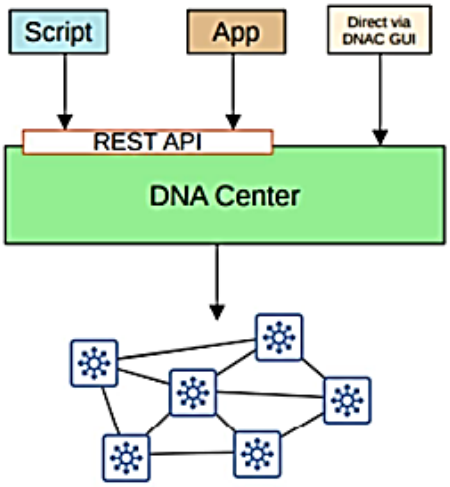
- The **overlay** is the virtual network built on top of the physical underlay network.
  - SD-Access uses VXLAN (Virtual Extensible LAN) to build tunnels

- The **fabric** is the combination of the *overlay* and *underlay*; the physical and virtual network as a whole.



SD-Access could be added on top of existing network (*brownfield deployment*) or a brand-new network (*greenfield deployment*).

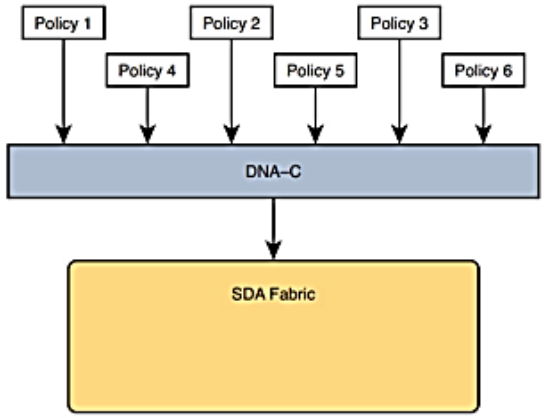
CISCO DNA CENTER



- Cisco DNA Center has two main roles:
  - → The SDN controller in SD-Access
  - → A network manager in a traditional network (non-SD-Access)
- DNA Center is an application installed on Cisco UCS server hardware.
- It has a REST API which can be used to interact with DNA center.
- The SBI supports protocols such as NETCONF and RESTCONF (as well as traditional protocols like Telnet, SSH, SNMP).
- DNA Center enables *Intent-Based Networking* (IBN).
  - → More buzzwords! Yay!
  - → The goal is to allow the engineer to communicate their intent for network behavior to DNA Center, and then DNA Center will take care of the details of the actual configurations and policies on devices.

- Traditional security policies using ACLs can become VERY cumbersome.
  - → ACLs can have **thousands** of entries.
  - → The intent of entries is forgotten with time and as engineers leave and new engineers take over.
  - → Configuring and applying the ACLs correctly across a network is cumbersome and leaves room for error.
- DNA Center allows the engineer to specify the intent of the policy (this group of users can't communicate with this group, this group can access this server but not that server, etc.), and DNA Center will take care of the exact details of implementing the policy.

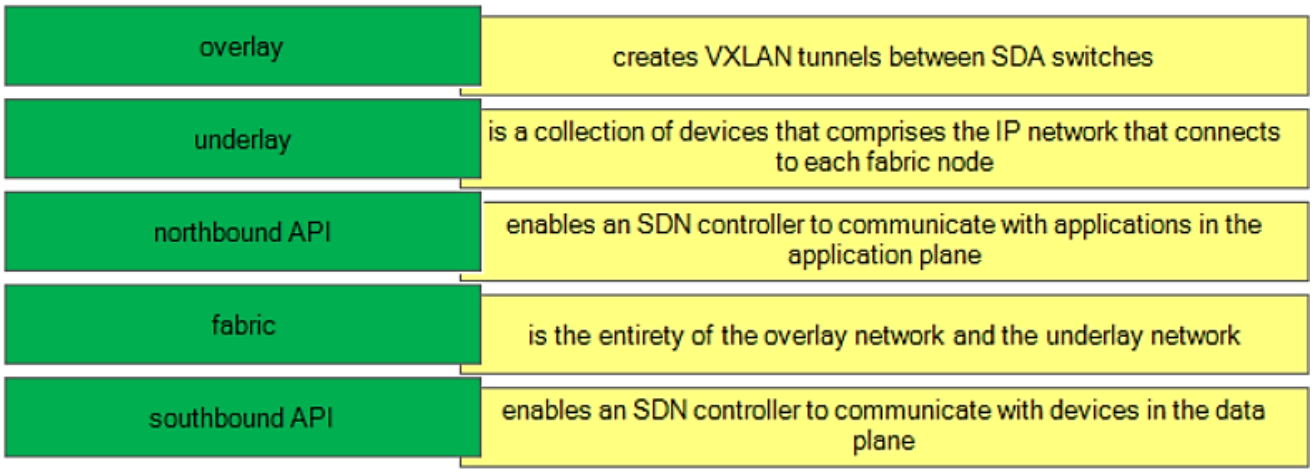
Figure 2-5 Six Policies Implemented by DNA Center



DNA CENTER VS TRADITIONAL NETWORK MANAGEMENT:

- Traditional network management:
  - → Devices are configured one-by-one via SSH or console connection.
  - → Devices are manually configured via console connection before being deployed.
  - → Configurations and policies are managed per-device. (distributed)
  - → New network deployments can take a long time due to the manual labor required.
  - → Errors and failures are more likely due to increased manual effort.

- DNA Center-based network management:
  - → Devices are centrally managed and monitored from the DNA Center GUI or other applications using its REST API.
  - → The administrator communicates their intended network behavior to DNA Center, which changes those intentions into configurations on the managed network devices.
  - → Configurations and policies are centrally managed.
  - → Software versions are also centrally managed. DNA Center can monitor cloud servers for new versions and then update the managed devices.
  - → New network deployments are much quicker. New devices can automatically receive their configurations from DNA Center without manual configuration.



REST is a northbound API architecture that uses Hypertext Transfer Protocol (HTTP) or HTTP Secure (HTTPS) to enable external resources to access and make use of programmatic methods that are exposed by the API. REST APIs typically return data in either Extensible Markup Language (XML) or JavaScript Object Notation (JSON) format.

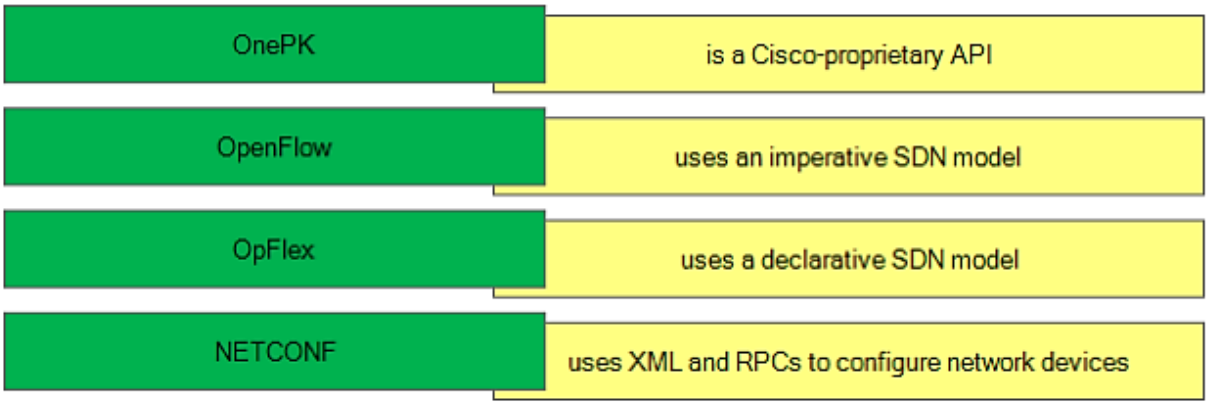
OSGi is a Java-based northbound API framework that is intended to enable the development of modular programs. OSGi also allows the use of the Python programming language as a means of extended controller functions. For transport, OSGi deployments often rely on HTTP.

SDA Miscellaneous

Cisco Network Assistant is a Cisco management solution that is typically installed on an administrator's desktop workstation. Cisco Network Assistant is a free Java-based desktop application that enables a local area network (LAN) administrator to perform network operations, diagnose problems, and interact with network devices by using a graphical user interface (GUI). A typical Cisco Network Assistant installation supports the management of up to 80 devices. Cisco Network Assistant predates Cisco Software-Defined Access (SDA) and does not support Cisco SDA.

SBIs

Answer:



Explanation:

Software-Defined Networking (SDN) is an intelligent network architecture in which a software controller assumes the control plane functionality for all network devices. Southbound Application Programming Interfaces (APIs) enable an SDN controller to communicate with devices on the network data plane. A southbound API is sometimes called a southbound interface (SBI). NETCONF, OnePK, OpenFlow, and OpFlex are all examples of southbound APIs. A northbound API is sometimes called a northbound interface (NBI). Northbound APIs enable an SDN controller to communicate with applications in the application plane. Examples of northbound APIs include Java Open Services Gateway initiative (OSGi) and Representational State Transfer (REST).

NETCONF uses Extensible Markup Language (XML) and Remote Procedure Calls (RPCs) to configure network devices. XML is used for both data encoding and protocol messages. NETCONF typically relies on Secure Shell (SSH) for transport.

OpFlex uses a declarative SDN model in which the instructions that are sent to the controller are not so detailed. The controller allows the devices in the data plane to make more network decisions about how to implement the policy.

OpenFlow uses an imperative SDN model in which detailed instructions are sent to the SDN controller when a new policy is to be configured. The SDN controller manages both the network and the policies applied to the devices.

The OnePK API is a Cisco-proprietary API. It uses Java, C, or Python to configure network devices. It can use either Secure Sockets Layer (SSL) or Transport Layer Security (TLS) to encrypt data in transit.

ACI Physical Design: Spine and Leaf

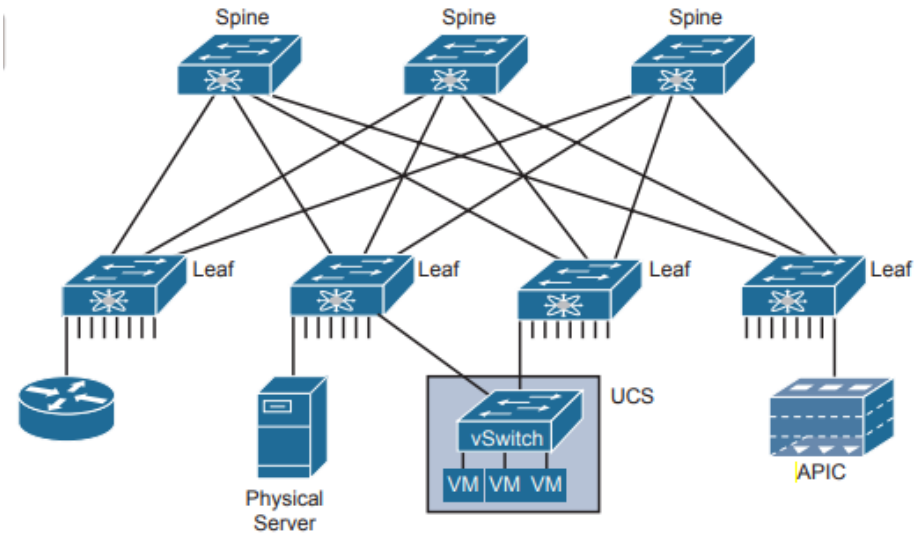


Figure 16-10 *Endpoints Found on the Leaf Switches Only*

- Each leaf switch must connect to every spine switch.
- Each spine switch must connect to every leaf switch.
- Leaf switches cannot connect to each other.
- Spine switches cannot connect to each other.
- Endpoints connect only to the leaf switches.

Leaf Node	Spine Node
must connect to every spine node	must connect to every leaf node
cannot connect to a leaf node	cannot connect to a spine node
can connect to an APIC	
can connect to an EPG	

Explanation:

Cisco Application Centric Infrastructure (ACI) is a data center technology that uses switches, categorized as spine and leaf nodes, to dynamically implement network application policies in response to application-level requirements. Network application policies are defined on a Cisco Application Policy Infrastructure Controller (APIC) and are implemented by the spine and leaf nodes.

Redundancy is also provided by the presence of multiple APICs, which are typically deployed as a cluster of three controllers. APICs are not directly involved in forwarding traffic and are therefore not required to connect to every spine or leaf node. Instead, the APIC cluster is connected to one or more leaf nodes in much the same manner that other endpoint groups (EPGs), such as application servers, are connected. Because APICs are not directly involved in forwarding traffic, the failure of an APIC does not affect the ability of the fabric to forward traffic.

6.7 JSON, XML, & YAML

XML and JSON enable easier integration and data sharing between web-based applications and lower bandwidth usage. JSON and XML are open standard file formats that are platform independent. JSON is easier to read format that is faster and requires less bandwidth than XML. JSON is less secure than XML and only supports UTF-8 character set.

Acronym	Name	Origin/Definition	Central Purpose	Common Use	Whitespace	Misc
JSON	JavaScript Object Notation	JavaScript (JS) language; RFC 8259	General data modeling and serialization	REST APIs	Insignificant	Uses human-readable text to store/transmit data objects, derived from JavaScript
XML	eXtensible Markup Language	World Wide Web Consortium (W3C.org)	Data-focused text markup that allows data modeling	REST APIs, Web pages	Insignificant	Less human-readable than JSON, similar to html
YAML	YAML Ain't Markup Language	YAML.org	General data modeling	Ansible	Significant Indention is important.	Very human-readable

6.7 Recognize components of JSON-encoded data

JSON (JavaScript Object Notation) is an open standard file format and data interchange format that uses human-readable text to store and transmit data objects.

JSON has:

- 4 'primitive' data types:
  - String: "Hello", "five", "4", "true", "null", etc.
  - Number: 1, 4, 1199, etc.
  - Boolean: true, false.
  - Null: null
- 2 'structured' data types:
  - Object: unordered list of key-value pairs, surrounded by curly brackets {}.
  - Array: is a series of values separated by commas, surrounded by square brackets [].

Example 18-6 *Simple JSON That Lists a Router's Interfaces*

```
{
    "R1": ["GigabitEthernet0/0", "GigabitEthernet0/1", "GigabitEthernet0/2/0"],
    "R2": ["GigabitEthernet1/0", "GigabitEthernet1/1", "GigabitEthernet0/3/0"]
}
```

Example 18-7 *One JSON Object (Dictionary) with Three Key:Value Pairs*

```
{
    "1stbest": "Messi",
    "2ndbest": "Ronaldo",
    "3rdbest": "Pele"
}
```

NOTE Python, the most common language to use for network automation, converts JSON objects to Python dictionaries, and JSON arrays to Python lists. For general conversation, many people refer to the JSON structures as dictionaries and lists rather than as objects and arrays

Example 18-8 *A JSON Snippet Showing a Single JSON Array (List)*

```
    [
        "Messi",
        "Ronaldo",
        "Dybala"
    ]
```

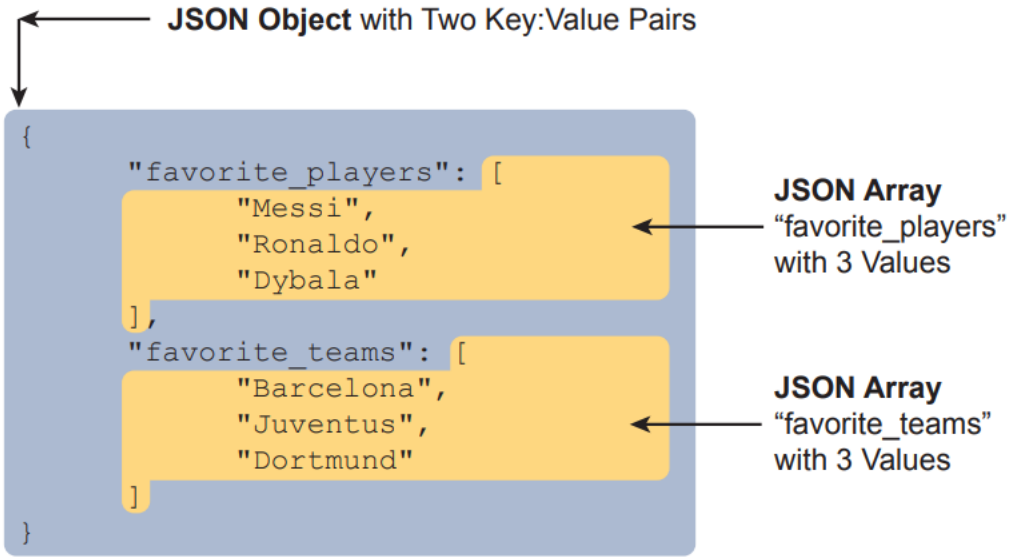


Figure 18-12 *Accurate/Complete JSON Data with One Object, Two Keys, Two JSON List Values*

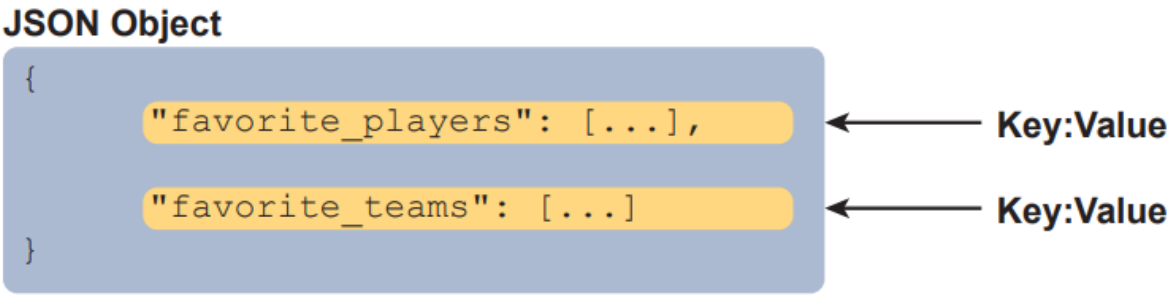


Figure 18-13 *Structural Representation of Figure 18-13's Primary Object and Two Key:Value Pairs*

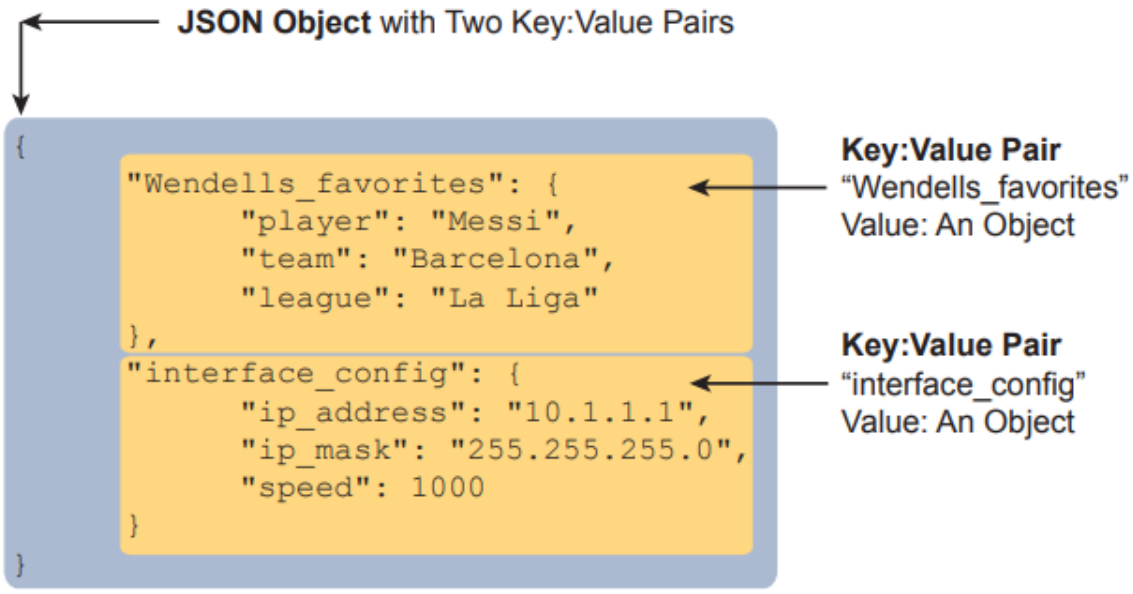


Figure 18-14 *A JSON Object, with Two Key:Value Pairs, Each Value Another Object*

JSON stored in a file or sent over the network looks like this:

```
{"1stbest": "Messi", "2ndbest": "Ronaldo", "3rdbest": "Pele"}
```

6.5 REST APIs (CRUD, HTTP verbs, and data encoding)

Representational State Transfer (**REST**) Application Programming Interfaces (APIs) **encode data in either Extensible markup Language (XML) format or in JavaScript Object Notation (JSON) format**. In addition, **REST APIs are typically used to communicate with** a Software-Defined Networking (SDN) **application plane**.

6.5 Describe characteristics of REST-based APIs (CRUD, HTTP verbs, and data encoding)

- CRUD (Create, Read, Update, Delete) refers to the operations we perform using REST API.
- HTTP uses **verbs** that map to these CRUD operations. REST APIs typically use HTTP.

Purpose	CRUD Operation	HTTP Verb
Creat new variables	Create	POST
Retrieve values of variables	Read	GET
Change values of variables	Update	PUT, PATCH
Delete variables	Delete	DELETE

- When sending a HTTP request to a HTTP server, the server will response will a status code:
  - 1xx: Informational. Such as:
    102 Processing: The server has received the request, currently processing, response is not yet available.
  - 2xx: Successful. Such as:
    200 OK: request succeeded.
    201 Created: request succeeded, new resource was created.
  - 3xx: Redirection. Such as:
    301 Moved Permanently: the server has moved to a new location.
  - 4xx Client Error:
    401 Unauthorized: Client needs authentication.
    404 Not Found: The requested resource was not found.
  - 5xx Server Error:
    500 Internal Server Error: server encountered something unexpected, does not know how to handle.
- REST is not an API, but a set of APIs, which bounded by 6 constrains:
  - Uniform Interface.
  - Client-Server.
  - Stateless (each API exchange is independent, not rely on past exchanges).
  - Cacheable or non-cacheable (cacheable resources must be cached).
  - Layered system:
  - Code-on-demand (optional):

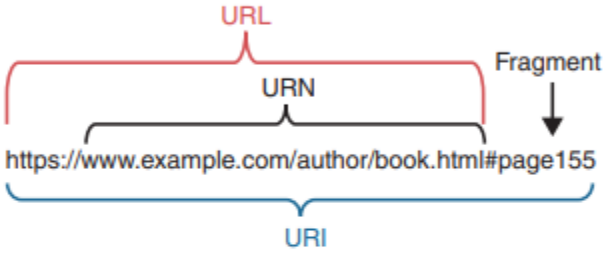
RESTful API Requests

A RESTful API is requested by using a URI, which is a string of characters that identifies a specific network resource. As shown in the Figure 1-1, a URI has two specializations:

- **Uniform resource name (URN):** Identifies only the namespace of the resource without reference to the protocol.

- **Uniform resource locator (URL):** Defines the network location of a specific resource on the network.

Figure 1-1 Structure of a URI

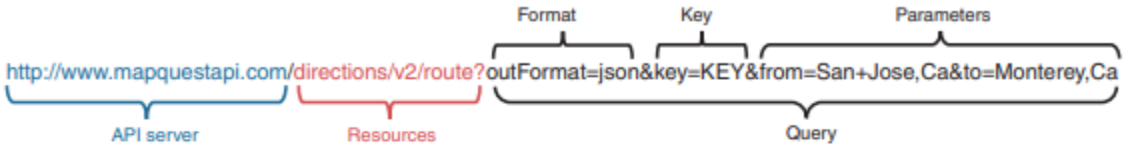


These are the parts of a URI, as shown in Figure 1-1:

- **Protocol/scheme:** HTTPS or another protocol, such as FTP, SFTP, mailto, or NNTP
- **Hostname:** In this case, www.example.com
- **Path and file name:** In this case, /author/book.html
- **Fragment:** In this case, #page155

A RESTful API request elicits a response from the API server. For example, the URI in Figure 1-2 is a properly formed GET request to the MapQuest API server for directions from San Jose to Monterey in JSON format.

Figure 1-2 RESTful API Request to the MapQuest API Server



These are the different parts of the API request:

- **API server:** The URL for the server that answers REST requests.
- **Resources:** Specifies the API that is being requested.
- **Query:** Specifies the data format and information the client is requesting from the API service. Queries can include
  - **Format:** This is usually JSON but can be YAML or XML.
  - **Key:** The key is for authorization, if required.
  - **Parameters:** Parameters are used to send information pertaining to the request.

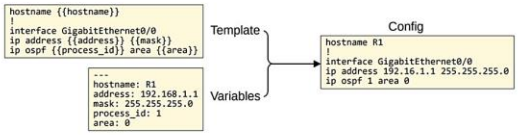
REST is a northbound API architecture that uses Hypertext Transfer Protocol (HTTP) or HTTP Secure (HTTPS) to enable external resources to access and make use of programmatic methods that are exposed by the API. REST APIs typically return data in either Extensible Markup Language (XML) or JavaScript Object Notation (JSON) format.

OSGi is a Java-based northbound API framework that is intended to enable the development of modular programs. OSGi also allows the use of the Python programming language as a means of extended controller functions. For transport, OSGi deployments often rely on HTTP.

6.6 Ansible, Puppet and Chef

CONFIGURATION MANAGEMENT TOOLS (ANSIBLE, PUPPET, CHEF)

- *Configuration drift* is when individual changes made over time cause a device's configuration to deviate from the standard/correct configurations as defined by the company.
- *Configuration provisioning* refers to how configuration changes are applied to devices.
  - This includes configuring new devices, too.
- Traditionally, configuration provisioning is done by connecting to devices one-by-one via SSH.
  - This is not practical in large networks.
- Config mgmt tools (Ansible, Puppet, Chef) allow us to make changes to devices at mass scale w/ a fraction of the time/effort.
- Two essential components: *templates* and *variables*



Configuration management tools are network automation tools that facilitate the centralized control of large numbers of network devices. They include: Ansible, Puppet, and Chef (which all use a **client-server model**). They can:

- Generate configurations for new devices in a large scale.
- Perform configuration changes on devices.
- Check device configurations for compliance with defined standards.
- Compare configurations between devices, between different versions of configurations on the same devices.

However, they have some differences:

	ANSIBLE	PUPPET	CHEF
KEY FILES DEFINING ACTIONS	Playbook	Manifest	Recipe, Run-list
COMMUNICATION PROTOCOL	SSH	HTTPS (via REST API)	HTTPS (via REST API)
KEY PORT	22 (SSH)	8140	10002
AGENT?	Agentless	Agent-based (or Agentless)	Agent-based
PUSH/PULL	Push	Pull	Pull
WRITTEN IN	Python	Ruby	Ruby
FILES	YAML	Proprietary	Ruby