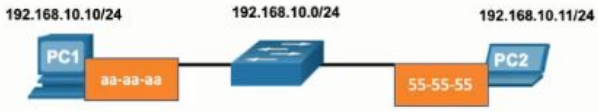
CHAPTER 9

Destination on Same Network

There are two primary addresses assigned to a device on an Ethernet LAN:

- **Layer 2 physical address (the MAC address)** – Used for NIC to NIC communications on the same Ethernet network.
- **Layer 3 logical address (the IP address)** – Used to send the packet from the source device to the destination device.

Layer 2 addresses are used to deliver frames from one NIC to another NIC on the same network. If a destination IP address is on the same network, the destination MAC address will be that of the destination device.



Destination MAC	Source MAC	Source IPv4	Destination IPv4
55-55-55	aa-aa-aa	192.168.10.10	192.168.10.11

Good job!

You have successfully identified the correct answers.

1. When sending a frame to another device on the same local network, the device sending the frame will use the MAC address of the destination device.
2. When sending a frame to another device on a remote network, the device sending the frame will use the MAC address of the local router interface, which is the default gateway.
3. Address Resolution Protocol (ARP) is used to determine the device MAC address of a known destination device IPv4 address. Neighbor Discovery (ND) is used to determine the MAC address of a known destination device IPv6 address.

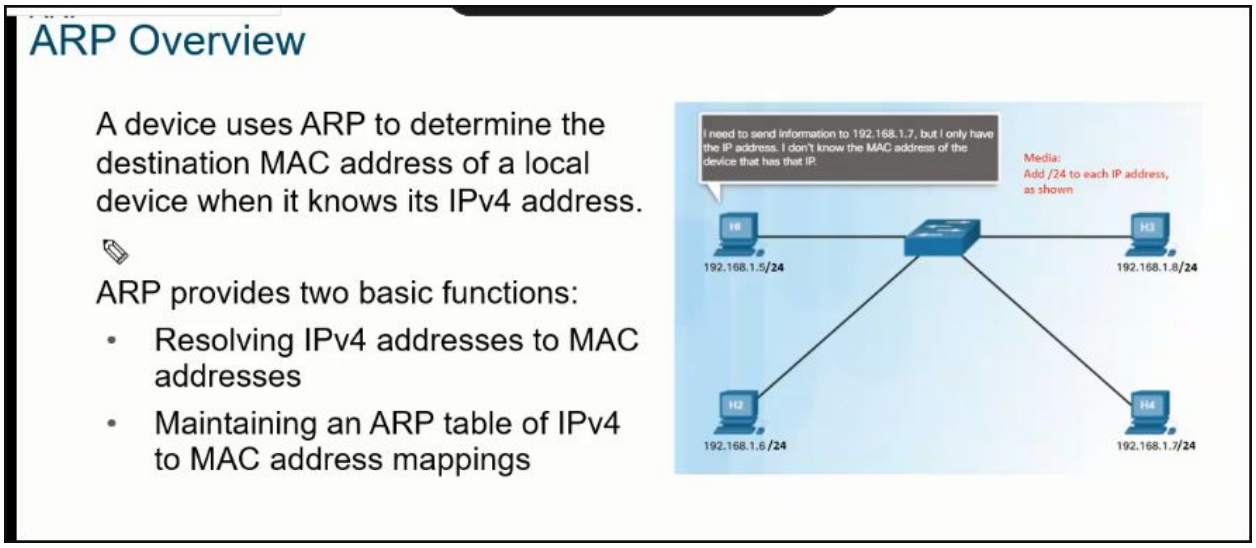
You answered 3 out of 3 questions correctly.

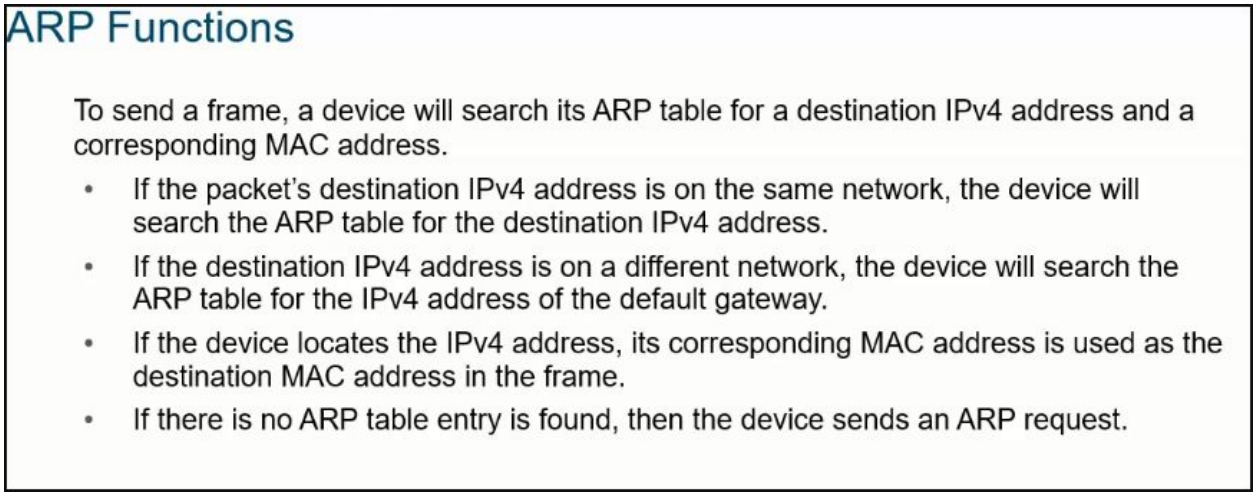
Every IP device on an Ethernet network has a unique Ethernet MAC address. When a device sends an Ethernet Layer 2 frame, it contains these two addresses:

- **Destination MAC address** - The Ethernet MAC address of the destination device on the same local network segment. If the destination host is on another network, then the destination address in the frame would be that of the default gateway (i.e., router).
- **Source MAC address** - The MAC address of the Ethernet NIC on the source host.

ARP provides two basic functions:

- Resolving IPv4 addresses to MAC addresses
- Maintaining a table of IPv4 to MAC address mappings





When a packet is sent to the data link layer to be encapsulated into an Ethernet frame, the device refers to a table in its memory to find the MAC address that is mapped to the IPv4 address. This table is stored temporarily in RAM memory and called the ARP table or the ARP cache.

The sending device will search its ARP table for a destination IPv4 address and a corresponding MAC address.

- If the packet's destination IPv4 address is on the same network as the source IPv4 address, the device will search the ARP table for the destination IPv4 address.
- If the destination IPv4 address is on a different network than the source IPv4 address, the device will search the ARP table for the IPv4 address of the default gateway.

In both cases, the search is for an IPv4 address and a corresponding MAC address for the device.

Each entry, or row, of the ARP table binds an IPv4 address with a MAC address. We call the relationship between the two values a map. This simply means that you can locate an IPv4 address in the table and discover the corresponding MAC address. The ARP table temporarily saves (caches) the mapping for the devices on the LAN.

If the device locates the IPv4 address, its corresponding MAC address is used as the destination MAC address in the frame. If there is no entry is found, then the device sends an ARP request.

ARP messages are encapsulated directly within an Ethernet frame. There is no IPv4 header. The ARP request is encapsulated in an Ethernet frame using the following header information:

- **Destination MAC address** – This is a broadcast address FF-FF-FF-FF-FF-FF requiring all Ethernet NICs on the LAN to accept and process the ARP request.
- **Source MAC address** – This is MAC address of the sender of the ARP request.
- **Type** - ARP messages have a type field of 0x806. This informs the receiving NIC that the data portion of the frame needs to be passed to the ARP process.

Because ARP requests are broadcasts, they are flooded out all ports by the switch, except the receiving port. All Ethernet NICs on the LAN process broadcasts and must deliver the ARP request to its operating system for processing. Every device must process the ARP request to see if the target IPv4 address matches its own. A router will not forward broadcasts out other interfaces.

Only one device on the LAN will have an IPv4 address that matches the target IPv4 address in the ARP request. All other devices will not reply.

IPv6 uses a similar process to ARP for IPv4, known as ICMPv6 Neighbor Discovery (ND). IPv6 uses neighbor solicitation and neighbor advertisement messages, similar to IPv4 ARP requests and ARP replies.

On a Cisco router, the **show ip arp** command is used to display the ARP table, as shown in the figure.

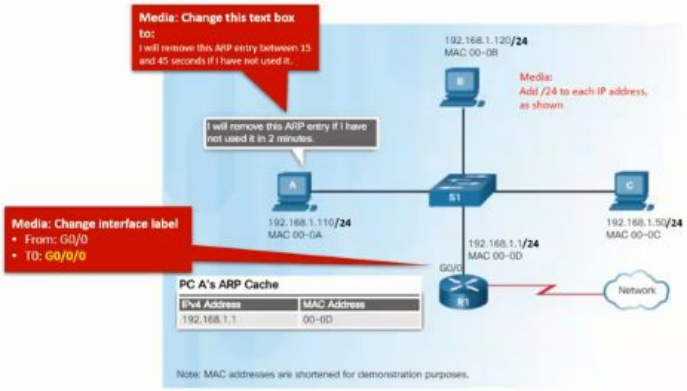
On a Windows 10 PC, the **arp –a** command is used to display the ARP table, as shown in the figure.

C:\Users\PC> **arp -a**

Enterprise level switches include mitigation techniques known as dynamic ARP inspection (DAI).

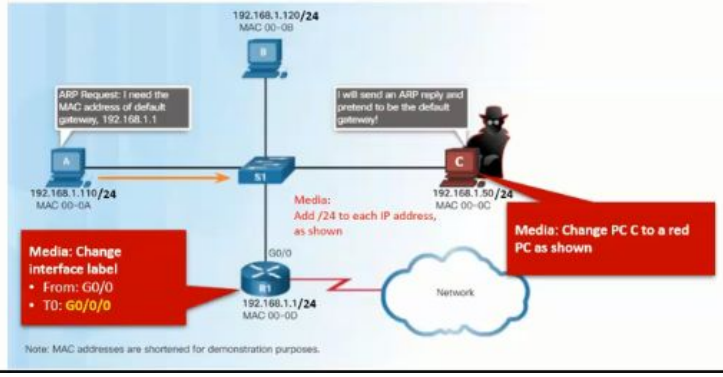
Removing Entries from an ARP Table

- Entries in the ARP table are not permanent and are removed when an ARP cache timer expires after a specified period of time.
- The duration of the ARP cache timer differs depending on the operating system.
- ARP table entries can also be removed manually by the administrator.



ARP Issues – ARP Broadcasting and ARP Spoofing

- ARP requests are received and processed by every device on the local network.
- Excessive ARP broadcasts can cause some reduction in performance.
- ARP replies can be spoofed by a threat actor to perform an ARP poisoning attack.
- Enterprise level switches include mitigation techniques to protect against ARP attacks.



Good job!

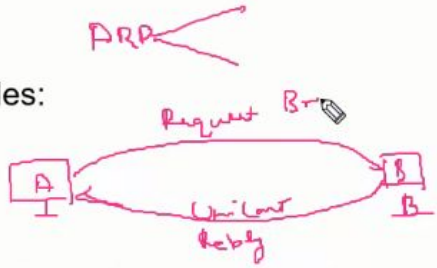
You have successfully identified the correct answers.

1. ARP has two primary functions: maintain a table of IPv4 to MAC address mappings and determine the MAC addresses of known IPv4 addresses.
2. The ARP table is cached temporarily in RAM.
3. The ARP table is cached temporarily in RAM.
4. The command **show ip arp** is used on Cisco routers to view the ARP table.
5. Two security issues with ARP Requests are that ARP messages are sent as broadcasts and can be spoofed.

You answered 5 out of 5 questions correctly.

Click **Switch1**and then the **CLI** tab. Enter the **show mac-address-table**command.

IPv6 Neighbor Discovery Messages



IPv6 Neighbor Discovery (ND) protocol provides:

- Address resolution
- Router discovery
- Redirection services
- ICMPv6 Neighbor Solicitation (NS) and Neighbor Advertisement (NA) messages are used for device-to-device messaging such as address resolution.
- ICMTPv6 Router Solicitation (RS) and Router Advertisement (RA) messages are used for messaging between devices and routers for router discovery.
- ICMPv6 redirect messages are used by routers for better next-hop selection.

IPv6 Neighbor Discovery protocol is sometimes referred to as ND or NDP. In this course, we will refer to it as ND. ND provides address resolution, router discovery, and redirection services for IPv6 using ICMPv6. ICMPv6 ND uses five ICMPv6 messages to perform these services:

- Neighbor Solicitation messages
- Neighbor Advertisement messages
- Router Solicitation messages
- Router Advertisement messages
- Redirect Message

Neighbor Solicitation and Neighbor Advertisement messages are used for device-to-device messaging such as address resolution (similar to ARP for IPv4). Devices include both host computers and routers.

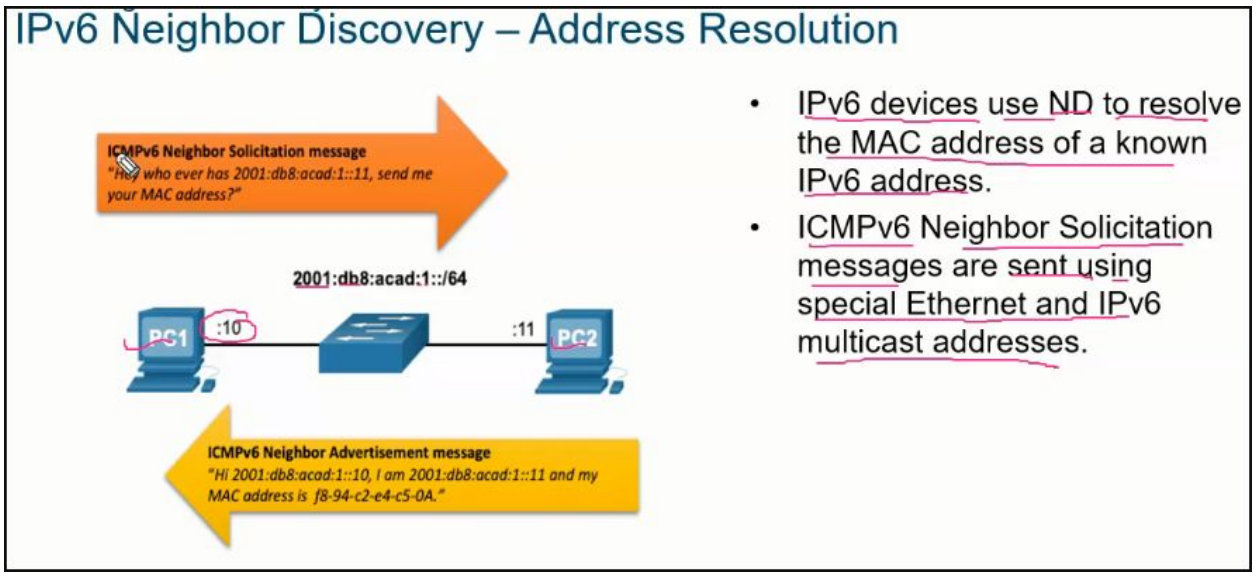
Router Solicitation and Router Advertisement messages are for messaging between devices and routers. Typically router discovery is used for dynamic address allocation and stateless address autoconfiguration (SLAAC).
ICMPv6 Neighbor Solicitation and Neighbor Advertisement messages are used for MAC address resolution. This is similar to ARP Requests and ARP Replies used by ARP for IPv4.

ICMPv6 Neighbor Solicitation messages are sent using special Ethernet and IPv6 multicast addresses. This allows the Ethernet NIC of the receiving device to determine whether the Neighbor Solicitation message is for itself without having to send it to the operating system for processing.

You have successfully identified the correct answers.

1. The two ICMPv6 messages used in SLAAC are the router solicitation and the router advertisement.
2. The two ICMPv6 messages used in determining the MAC address of a known IPv6 address are the neighbor solicitation and the neighbor advertisement.
3. ICMPv6 neighbor solicitation messages are sent as a multicast.

You answered 3 out of 3 questions correctly.



What did I learn in this module?

MAC and IP

Layer 2 physical addresses (i.e., Ethernet MAC addresses) are used to deliver the data link frame with the encapsulated IP packet from one NIC to another NIC on the same network. If the destination IP address is on the same network, the destination MAC address will be that of the destination device. When the destination IP address (IPv4 or IPv6) is on a remote network, the destination MAC address will be the address of the host default gateway (i.e., the router interface). Along each link in a path, an IP packet is encapsulated in a frame. The frame is specific to the data link technology associated that is associated with that link, such as Ethernet. If the next-hop device is the final destination, the destination MAC address will be that of the device Ethernet NIC. How are the IP addresses of the IP packets in a data flow associated with the MAC addresses on each link along the path to the destination? For IPv4 packets, this is done through a process called ARP. For IPv6 packets, the process is ICMPv6 ND.

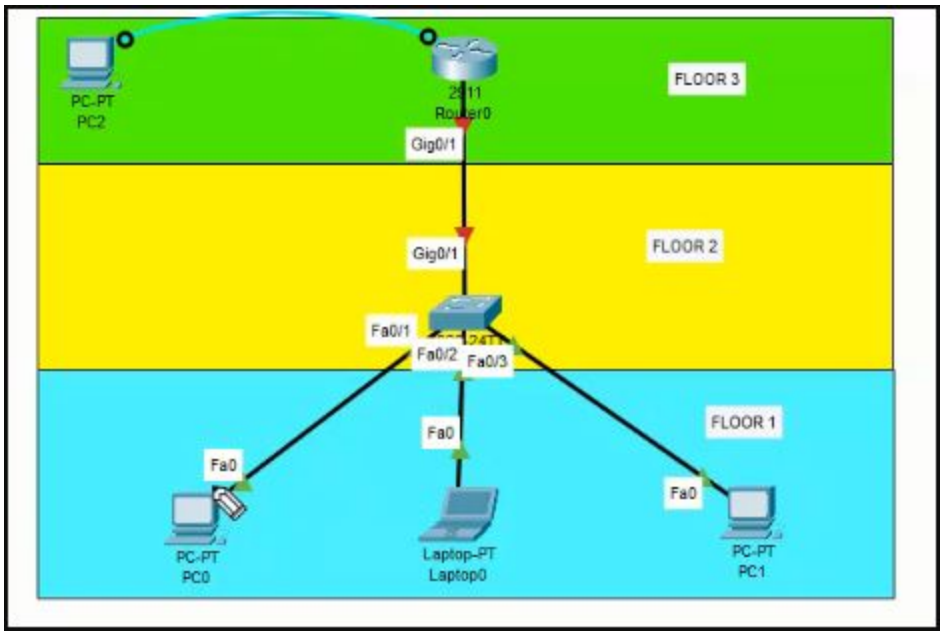
ARP

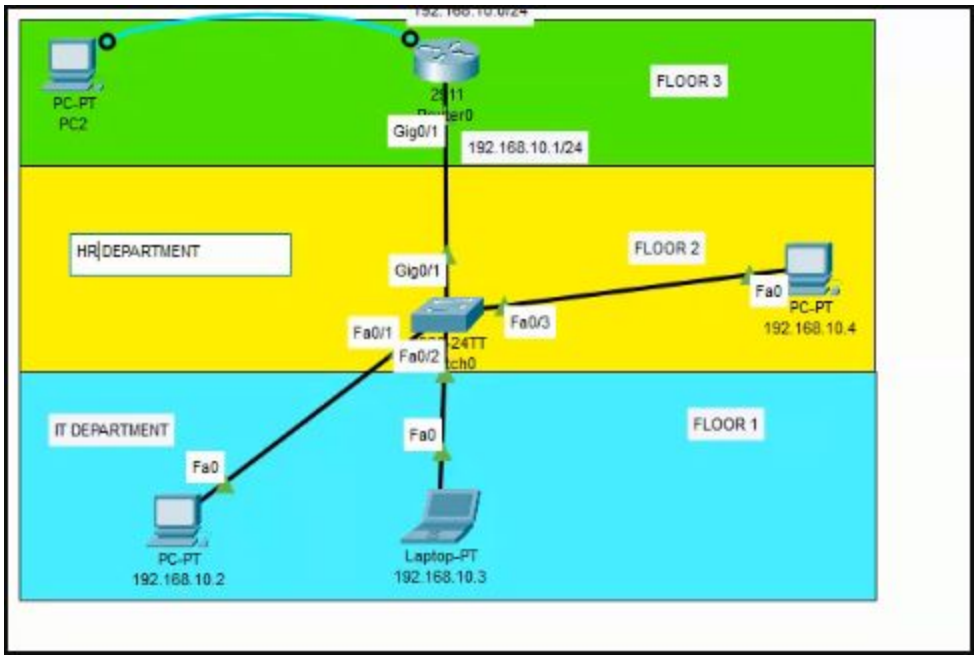
Every IP device on an Ethernet network has a unique Ethernet MAC address. When a device sends an Ethernet Layer 2 frame, it contains these two addresses: destination MAC address and source MAC address. A device uses ARP to determine the destination MAC address of a local device when it knows its IPv4 address. ARP provides two basic functions: resolving IPv4 addresses to MAC addresses and maintaining a table of IPv4 to MAC address mappings. The ARP request is encapsulated in an Ethernet frame using this header information: source and destination MAC addresses and type. Only one device on the LAN will have an IPv4 address that matches the target IPv4 address in the ARP request. All other devices will not reply. The ARP reply contains the same header fields as the request. Only the device that originally sent the ARP request will receive the unicast ARP reply. After the ARP reply is received, the device will add the IPv4 address and the corresponding MAC address to its ARP table. When the destination IPv4 address is not on the same network as the source IPv4 address, the source device needs to send the frame to its default gateway. This is the interface of the local router. For each device, an ARP cache timer removes ARP entries that have not been used for a specified period of time. Commands may also be used to manually remove some or all of the entries in the ARP table. As a broadcast frame, an ARP request is received and processed by every device on the local network, which could cause the network to slow down. A threat actor can use ARP spoofing to perform an ARP poisoning attack.

Neighbor Discovery

IPv6 does not use ARP, it uses the ND protocol to resolve MAC addresses. ND provides address resolution, router discovery, and redirection services for IPv6 using ICMPv6. ICMPv6 ND uses five ICMPv6 messages to perform these services: neighbor solicitation, neighbor advertisement, router solicitation, router advertisement, and redirect. Much like ARP for IPv4, IPv6 devices use IPv6 ND to resolve the MAC address of a device to a known IPv6 address.

CHAPTER 10


PC-PT
PC2
2811
Router0
FLOOR 3
Gig0/1
Gig0/1
FLOOR 2
Fa0/1
Fa0/2
Fa0/3
Fa0
FLOOR 1
Fa0
Fa0
PC-PT
PC0
Laptop-PT
Laptop0
PC-PT
PC1


PC-PT
PC2
FLOOR 3
Gig0/1
192.168.10.1/24
HR DEPARTMENT
Gig0/1
FLOOR 2
Fa0
PC-PT
192.168.10.4
Fa0/1
Fa0/3
Fa0/2
IT DEPARTMENT
Fa0
FLOOR 1
Fa0
PC-PT
192.168.10.2
Laptop-PT
192.168.10.3

Basic Router Configuration Example

- Commands for basic router configuration on R1.
- Configuration is saved to NVRAM.

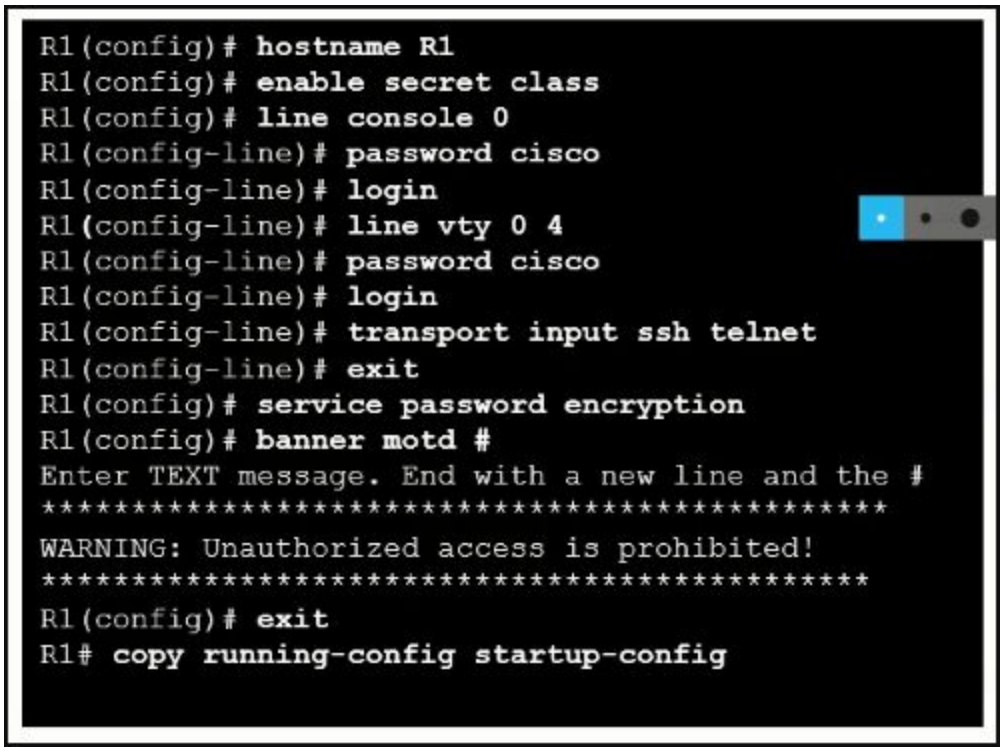
```
R1(config)# hostname R1
R1(config)# enable secret class
R1(config)# line console 0
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# transport input ssh telnet
R1(config-line)# exit
R1(config)# service password encryption
R1(config)# banner motd #
Enter TEXT message. End with a new line and the #
**********************************************
WARNING: Unauthorized access is prohibited!
**********************************************
R1(config)# exit
R1# copy running-config startup-config
```

```
Router1(config)#do show history
  hostname Router1
  ip domain-name cisco.com
  crypto key generate rsa
  username mohan password mohan
  username rohit password rohit
  line vty 0 1
  transport input ssh
  login
  exit
  do show history
Router1(config)#
```

Router(config)# **interface** *type-and-number*
Router(config-if)# **description** *description-text*
Router(config-if)# **ip address** *ipv4-address subnet-mask*
Router(config-if)# **ipv6 address** *ipv6-address/prefix-length*
Router(config-if)# **no shutdown**

Commands	Description
show ip interface brief **show ipv6 interface brief**	The output displays all interfaces, their IP addresses, and their current status. The configured and connected interfaces should display a Status of “up” and Protocol of “up”. Anything else would indicate a problem with either the configuration or the cabling.

show ip route **show ipv6 route**	Displays the contents of the IP routing tables stored in RAM.
show interfaces	Displays statistics for all interfaces on the device. However, this command will only display the IPv4 addressing information.
show ip interfaces	Displays the IPv4 statistics for all interfaces on a router.
show ipv6 interface	Displays the IPv6 statistics for all interfaces on a router.



Configure Router Interfaces

Configuring a router interface includes issuing the following commands:

```
Router(config)# interface type-and-number
Router(config-if)# description description-text
Router(config-if)# ip address ipv4-address subnet-mask
Router(config-if)# ipv6 address ipv6-address/prefix-length
Router(config-if)# no shutdown
```

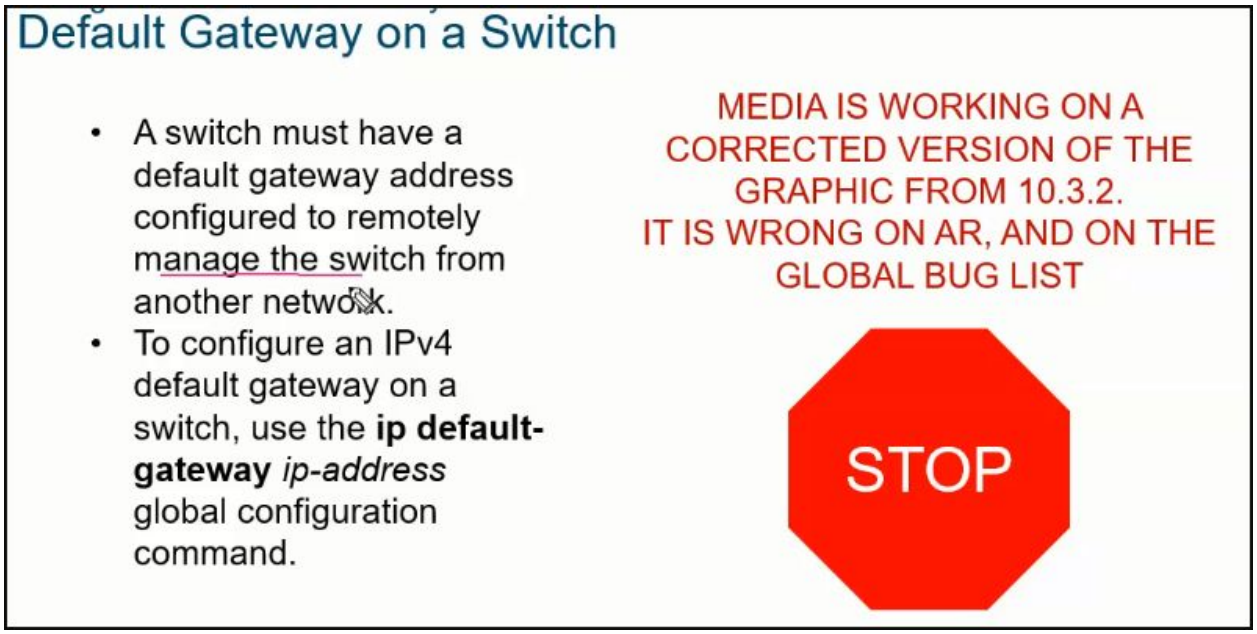
- It is a good practice to use the **description** command to add information about the network connected to the interface.
- The **no shutdown** command activates the interface.

Verify Interface Configuration

To verify interface configuration use the **show ip interface brief** and **show ipv6 interface brief** commands shown here:

```
R1#  show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0/0   192.168.10.1    YES manual up                    up
GigabitEthernet0/0/1   209.165.200.225 YES manual up                    up
Vlan1                  unassigned      YES unset  administratively down down
```

```
R1#  show ipv6 interface brief
GigabitEthernet0/0/0       [up/up]
    FE80::201:C9FF:FE89:4501
    2001:DB8:ACAD:10::1
GigabitEthernet0/0/1       [up/up]
    FE80::201:C9FF:FE89:4502
    2001:DB8:FEED:224::1
Vlan1                      [administratively down/down]
    unassigned
R1#
```



To configure an IPv4 default gateway on a switch, use the **ip default-gateway** *ip-address* global configuration command. The *ip-address* that is configured is the IPv4 address of the local router interface connected to the switch.

A workgroup switch can also be configured with an IPv6 address on an SVI. However, the switch does not require the IPv6 address of the default gateway to be configured manually. The switch will automatically receive its default gateway from the ICMPv6 Router Advertisement message from the router.

What did I learn in this module?

Configure Initial Router Settings

The following tasks should be completed when configuring initial settings on a router.

1. Configure the device name.
2. Secure privileged EXEC mode.
3. Secure user EXEC mode.
4. Secure remote Telnet / SSH access.
5. Secure all passwords in the config file.
6. Provide legal notification.
7. Save the configuration.

Configure Interfaces

For routers to be reachable, the router interfaces must be configured. The Cisco ISR 4321 router is equipped with two Gigabit Ethernet interfaces: GigabitEthernet 0/0/0 (G0/0/0) and

GigabitEthernet 0/0/1 (G0/0/1). The tasks to configure a router interface are very similar to a management SVI on a switch. Using the **no shutdown** command activates the interface. The interface must also be connected to another device, such as a switch or a router, for the physical layer to be active. There are several commands that can be used to verify interface configuration including the **show ip interface brief** and **show ipv6 interface brief**, the **show ip route** and **show ipv6 route**, as well as **show interfaces**, **show ip interface** and **show ipv6 interface.**

Configure the Default Gateway

For an end device to communicate over the network, it must be configured with the correct IP address information, including the default gateway address. The default gateway address is generally the router interface address for the router that is attached to the local network of the host. The IP address of the host device and the router interface address must be in the same network. To connect to and manage a switch over a local IP network, it must have a switch virtual interface (SVI) configured. The SVI is configured with an IPv4 address and subnet mask on the local LAN. The switch must also have a default gateway address configured to remotely manage the switch from another network. To configure an IPv4 default gateway on a switch, use the **ip default-gateway ip-address** global configuration command. Use the IPv4 address of the local router interface that is connected to the switch.

CHAPTER 11

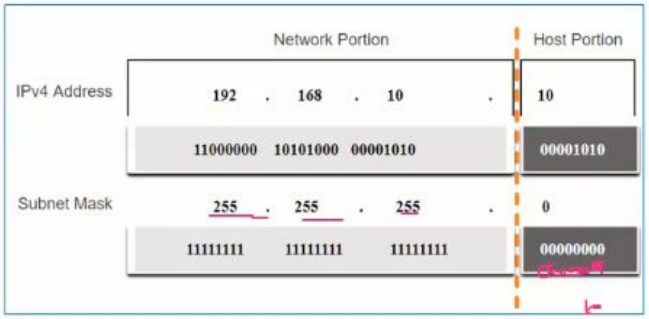
Topic Title	Topic Objective
IPv4 Address Structure	Describe the structure of an IPv4 address including the network portion, the host portion, and the subnet mask.
IPv4 Unicast, Broadcast, and Multicast	Compare the characteristics and uses of the unicast, broadcast and multicast IPv4 addresses.
Types of IPv4 Addresses	Explain public, private, and reserved IPv4 addresses.
Network Segmentation	Explain how subnetting segments a network to enable better communication.
Subnet an IPv4 Network	Calculate IPv4 subnets for a /24 prefix.
Subnet a /16 and a /8 Prefix	Calculate IPv4 subnets for a /16 and /8 prefix.
Subnet To Meet Requirements	Given a set of requirements for subnetting, implement an IPv4 addressing scheme.
Variable Length Subnet Masking	Explain how to create a flexible addressing scheme using variable length subnet masking (VLSM).
Structured Design	Implement a VLSM addressing scheme.

Note that the subnet mask does not actually contain the network or host portion of an IPv4 address, it just tells the computer where to look for the part of the IPv4 address that is the network portion and which part is the host portion.

The actual process used to identify the network portion and host portion is called ANDing.

The Subnet Mask

- To identify the network and host portions of an IPv4 address, the subnet mask is compared to the IPv4 address bit for bit, from left to right.
- The actual process used to identify the network and host portions is called ANDing.



The Prefix Length

- A prefix length is a less cumbersome method used to identify a subnet mask address.
- The prefix length is the number of bits set to 1 in the subnet mask.
- It is written in "slash notation" therefore, count the number of bits in the subnet mask and prepend it with a slash.

Subnet Mask	32-bit Address	Prefix Length
255.0.0.0	11111111.00000000.00000000.00000000	/8
255.255.0.0	11111111.11111111.00000000.00000000	/16
255.255.255.0	11111111.11111111.11111111.00000000	/24
255.255.255.128	11111111.11111111.11111111.10000000	/25
255.255.255.192	11111111.11111111.11111111.11000000	/26
255.255.255.224	11111111.11111111.11111111.11100000	/27
255.255.255.240	11111111.11111111.11111111.11110000	/28
255.255.255.248	11111111.11111111.11111111.11111000	/29
255.255.255.252	11111111.11111111.11111111.11111100	/30

Expressing network addresses and host addresses with the dotted decimal subnet mask address can become cumbersome. Fortunately, there is an alternative method of identifying a subnet mask, a method called the prefix length.

Within each network are three types of IP addresses:

- Network address
- Host addresses
- Broadcast address

A network address is an address that represents a specific network. A device belongs to this network if it meets three criteria:

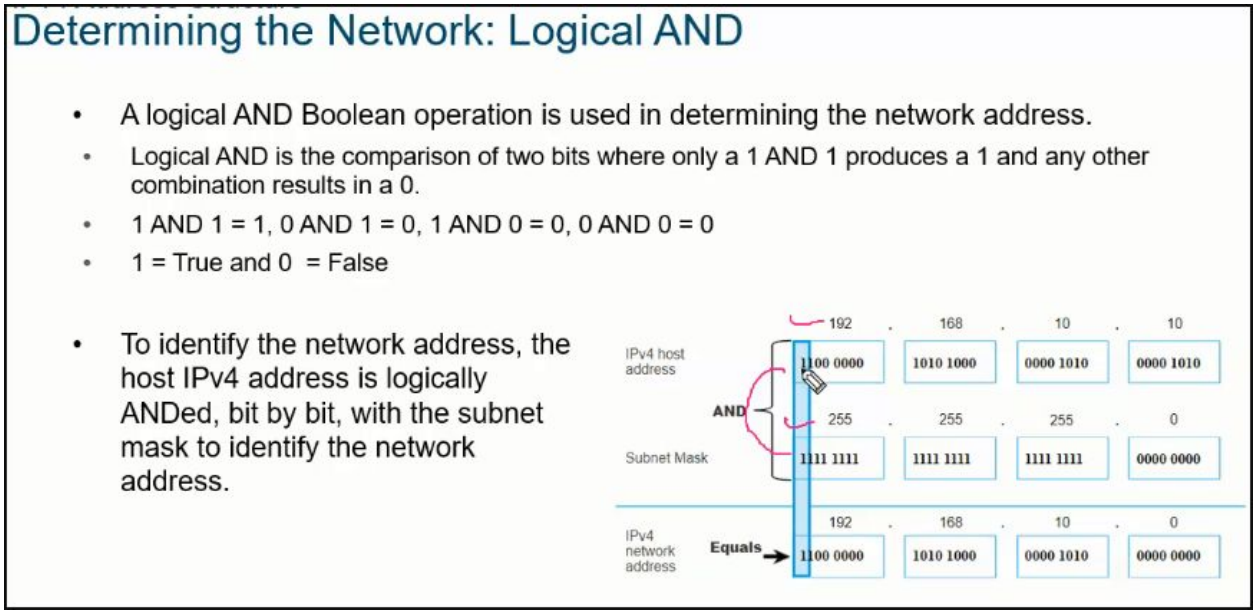
- It has the same subnet mask as the network address.
- It has the same network bits as the network address, as indicated by the subnet mask.
- It is located on the same broadcast domain as other hosts with the same network address.

A host determines its network address by performing an AND operation between its IPv4 address and its subnet mask.

Host addresses

Host addresses are addresses that can be assigned to a device such as a host computer, laptop, smart phone, web camera, printer, router, etc. The host portion of the address is the bits indicated by 0 bits in the subnet mask. Host addresses can have any combination of bits in the host portion except for all 0 bits (this would be a network address) or all 1 bits (this would be a broadcast address).

All devices within the same network, must have the same subnet mask and the same network bits. Only the host bits will differ and must be unique.



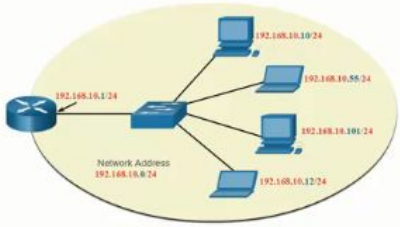
Video – Network, Host and Broadcast Addresses

This video will cover the following:

- Network address
- Broadcast Address
- First usable host
- Last usable host

Network, Host, and Broadcast Addresses

- Within each network are three types of IP addresses:
  - Network address
  - Host addresses
  - Broadcast address



	Network Portion			Host Portion	Host Bits
Subnet mask **255.255.255.0** or **/24**	255 11111111	255 11111111	255 11111111	0 00000000	
Network address **192.168.10.0** or **/24**	192 11000000	168 10100000	10 00001010	0 00000000	All 0s
First address **192.168.10.1** or **/24**	192 11000000	168 10100000	10 00001010	1 00000001	All 0s and a 1
Last address **192.168.10.254** or **/24**	192 11000000	168 10100000	10 00001010	254 11111110	All 1s and a 0
Broadcast address **192.168.10.255** or **/24**	192 11000000	168 10100000	10 00001010	255 11111111	All 1s

Good job!

You have successfully identified the best answers.

1. The network address for 10.5.4.100 with a subnet mask of 255.255.255.0 is 10.5.4.0.
2. The network address for 172.16.4.100 with a subnet mask of 255.255.0.0 is 172.16.0.0.
3. Host A is on network 10.5.4.0. Therefore, devices with the IPv4 addresses 10.5.4.1 and 10.5.4.99 are on the same network.
4. Host A is on network 172.16.0.0. Therefore, devices with the IPv4 addresses 172.16.4.99 and 172.16.0.1 are on the same network.
5. Host A is on network 192.168.1.0. Therefore, devices with the IPv4 addresses 192.168.1.1 and 192.168.1.100 are on the same network.

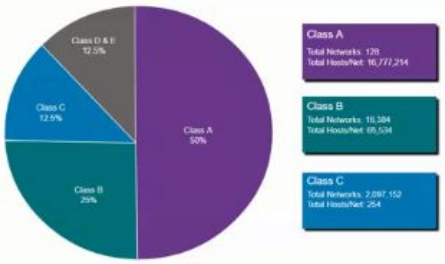
You answered 5 out of 5 questions correctly.

IPv4 unicast host addresses are in the address range of 1.1.1.1 to 223.255.255.255. However, within this range are many addresses that are reserved for special purposes.
Because of security concerns and prior abuse from malicious users, directed broadcasts are turned off by default starting with Cisco IOS Release 12.0 with the global configuration command **no ip directed-broadcasts**.

Legacy Classful Addressing

RFC 790 (1981) allocated IPv4 addresses in classes

- Class A (0.0.0.0/8 to 127.0.0.0/8)
- Class B (128.0.0.0 /16 – 191.255.255.255 /16)
- Class C (192.0.0.0 /24 – 223. .255.255.255 /24)
- Class D (224.0.0.0 to 239.255.255.255)
- Class E (240.0.0.0 – 255.255.255.255)



- Classful addressing wasted many IPv4 addresses.

Classful address allocation was replaced with classless addressing which ignores the rules of classes (A, B, C).

cisco

A multicast packet is a packet with a destination IP address that is a multicast address. IPv4 has reserved the 224.0.0.0 to 239.255.255.255 addresses as a multicast range.

Hosts that receive particular multicast packets are called multicast clients. The multicast clients use services requested by a client program to subscribe to the multicast group.

Each multicast group is represented by a single IPv4 multicast destination address. When an IPv4 host subscribes to a multicast group, the host processes packets addressed to this multicast address, and packets addressed to its uniquely allocated unicast address.

Routing protocols such as OSPF use multicast transmissions. For example, routers enabled with OSPF communicate with each other using the reserved OSPF multicast address 224.0.0.5. Only devices enabled with OSPF will process these packets with 224.0.0.5 as the destination IPv4 address. All other devices will ignore these packets.

Before the ISP can forward this packet, it must translate the source IPv4 address, which is a private address, to a public IPv4 address using Network Address Translation (NAT). NAT is used to translate between private IPv4 and public IPv4 addresses. This is usually done on the router that connects the internal network to the ISP network. Private IPv4 addresses in the organization's intranet will be translated to public IPv4 addresses before routing to the internet.

Note: Although, a device with a private IPv4 address is not directly accessible from another device across the internet, the IETF does not consider private IPv4 addresses or NAT as effective security measures.

Organizations that have resources available to the internet, such as a web server, will also have devices that have public IPv4 addresses. As shown in the figure, this part of the network is known as the DMZ (demilitarized zone). The router in the figure not only performs routing, it also performs NAT and acts as a firewall for security.

The diagram is a network topology showing a router in the center with three connections; one to the company Intranet, one to a DMZ, and one to the Internet. On the left is the Intranet with devices using private IPv4 addresses. At the top, is the DMZ with two servers using public IPv4 addresses. On the right is the Internet cloud. The router is acting as a firewall and performing NAT.

Loopback addresses

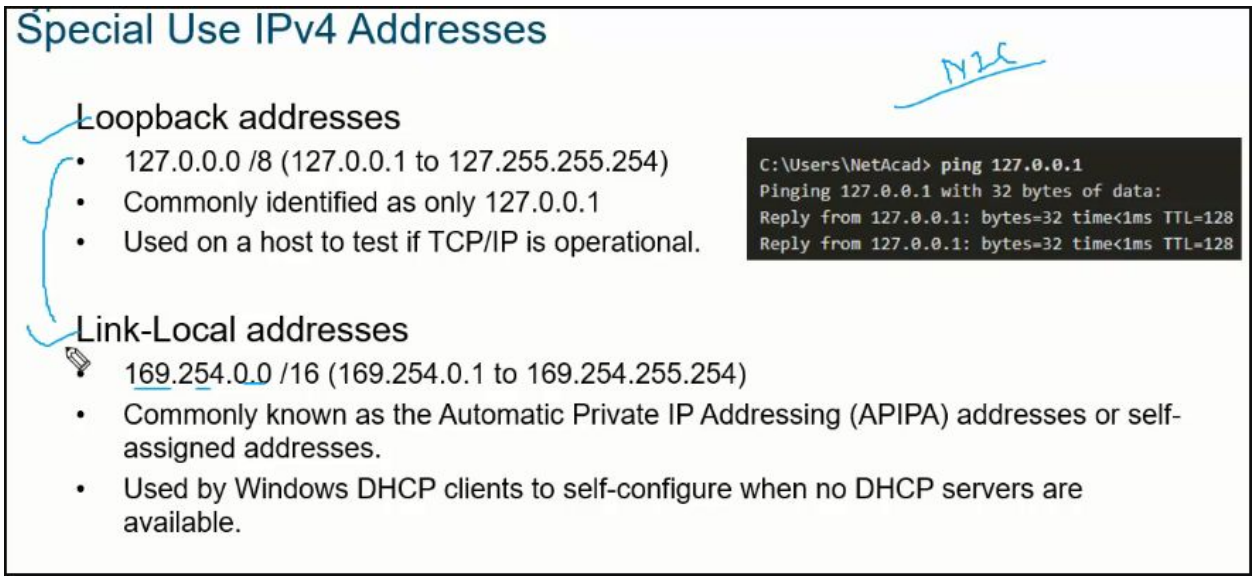
Loopback addresses (127.0.0.0 /8 or 127.0.0.1 to 127.255.255.254) are more commonly identified as only 127.0.0.1, these are special addresses used by a host to direct traffic to itself. For example, it can be used on a host to test if the TCP/IP configuration is operational,

Link-Local addresses

Link-local addresses (169.254.0.0 /16 or 169.254.0.1 to 169.254.255.254) are more commonly known as the Automatic Private IP Addressing (APIPA) addresses or self-assigned addresses. They are used by a Windows DHCP client to self-configure in the event that there are no DHCP servers available. Link-local addresses can be used in a peer-to-peer connection but are not commonly used for this purpose.

- **Class A (0.0.0.0/8 to 127.0.0.0/8)** - Designed to support extremely large networks with more than 16 million host addresses. Class A used a fixed /8 prefix with the first octet to indicate the network address and the remaining three octets for host addresses (more than 16 million host addresses per network).
- **Class B (128.0.0.0 /16 - 191.255.0.0 /16)** - Designed to support the needs of moderate to large size networks with up to approximately 65,000 host addresses. Class B used a fixed /16 prefix with the two high-order octets to indicate the network address and the remaining two octets for host addresses (more than 65,000 host addresses per network).
- **Class C (192.0.0.0 /24 - 223.255.255.0 /24)** - Designed to support small networks with a maximum of 254 hosts. Class C used a fixed /24 prefix with the first three octets to indicate the network and the remaining octet for the host addresses (only 254 host addresses per network).

Note: There is also a Class D multicast block consisting of 224.0.0.0 to 239.0.0.0 and a Class E experimental address block consisting of 240.0.0.0 - 255.0.0.0.



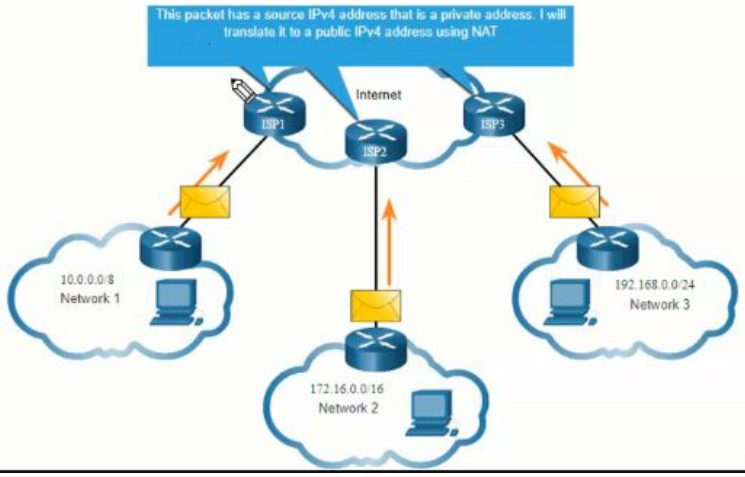
Public and Private IPv4 Addresses

- As defined in in RFC 1918, public IPv4 addresses are globally routed between internet service provider (ISP) routers.
- Private addresses are common blocks of addresses used by most organizations to assign IPv4 addresses to internal hosts.
- Private IPv4 addresses are not unique and can be used internally within any network.
- However, private addresses are not globally routable.

Network Address and Prefix	RFC 1918 Private Address Range
10.0.0.0/8	10.0.0.0 - 10.255.255.255
172.16.0.0/12	172.16.0.0 - 172.31.255.255
192.168.0.0/16	192.168.0.0 - 192.168.255.255

Routing to the Internet

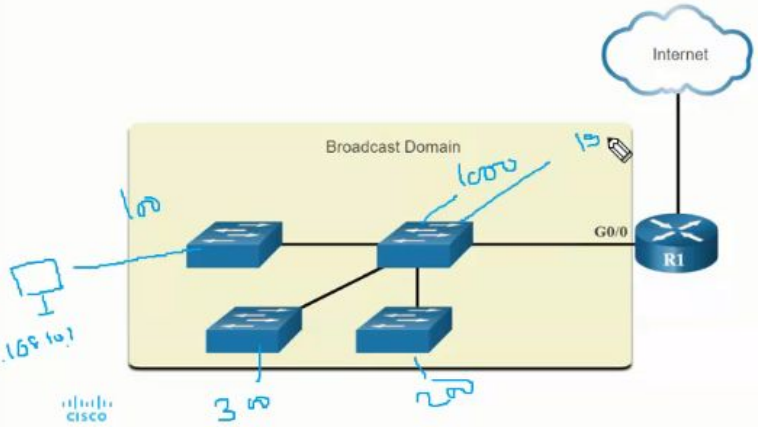
- Network Address Translation (NAT) translates private IPv4 addresses to public IPv4 addresses.
- NAT is typically enabled on the edge router connecting to the internet.
- It translates the internal private address to a public global IP address.



Both IPv4 and IPv6 addresses are managed by the Internet Assigned Numbers Authority (IANA). The IANA manages and allocates blocks of IP addresses to the Regional Internet Registries (RIRs).RIRs are responsible for allocating IP addresses to ISPs who provide IPv4 address blocks to organizations and smaller ISPs. Organizations can also get their addresses directly from an RIR (subject to the policies of that RIR)

Broadcast Domains and Segmentation

- Many protocols use broadcasts or multicasts (e.g., ARP use broadcasts to locate other devices, hosts send DHCP discover broadcasts to locate a DHCP server.)
- Switches propagate broadcasts out all interfaces except the interface on which it was received.



- The only device that stops broadcasts is a router.
- Routers do not propagate broadcasts.
- Each router interface connects to a broadcast domain and broadcasts are only propagated within that specific broadcast domain.

You have successfully identified the correct answers.

1. Private IPv4 addresses are assigned to devices within an organization's intranet (internal network) and any organization (home, school, office, company) can use the 10.0.0.0/8 address.
2. To access a device over the internet, the destination IPv4 address must be a public address. Public IPv4 address exhaustion is a reason why there are private IPv4 address and why organizations are transitioning to IPv6.
3. RIRs receive IP addresses from IANA and are responsible for allocating these addresses to ISPs and some other organizations.

You answered 3 out of 3 questions correctly.

In an Ethernet LAN, devices use broadcasts and the Address Resolution Protocol (ARP) to locate other devices.. ARP sends Layer 2 broadcasts to a known IPv4 address on the local network to discover the associated MAC address. Devices on Ethernet LANs also locate other devices using services. A host typically acquires its IPv4 address configuration using the Dynamic Host Configuration Protocol (DHCP) which sends broadcasts on the local network to locate a DHCP server.

The solution is to reduce the size of the network to create smaller broadcast domains in a process called subnetting. These smaller network spaces are called subnets.

This is the basis of subnetting: using host bits to create additional subnets.

Note: The terms subnet and network are often used interchangeably. Most networks are a subnet of some larger address block.

Subnetting reduces overall network traffic and improves network performance. It also enables an administrator to implement security policies such as which subnets are allowed or not allowed to communicate together. Another reason is that it reduces the number of devices affected by abnormal broadcast traffic due to misconfigurations, hardware/software problems, or malicious intent.
For each bit borrowed in the fourth octet, the number of subnetworks available is doubled, while reducing the number of host addresses per subnet:

- **/25 row** - Borrowing 1 bit from the fourth octet creates 2 subnets supporting 126 hosts each.
- **/26 row** - Borrowing 2 bits creates 4 subnets supporting 62 hosts each.
- **/27 row** - Borrowing 3 bits creates 8 subnets supporting 30 hosts each.
- **/28 row** - Borrowing 4 bits creates 16 subnets supporting 14 hosts each.
- **/29 row** - Borrowing 5 bits creates 32 subnets supporting 6 hosts each.
- **/30 row** - Borrowing 6 bits creates 64 subnets supporting 2 hosts each.

The figure shows a typical enterprise network:

- **Intranet -** This is the internal part of a company's network, accessible only within the organization. Devices in the intranet use private IPv4 addresses.
- **DMZ -** This is part of the company's network containing resources available to the internet such as a web server. Devices in the DMZ use public IPv4 addresses.

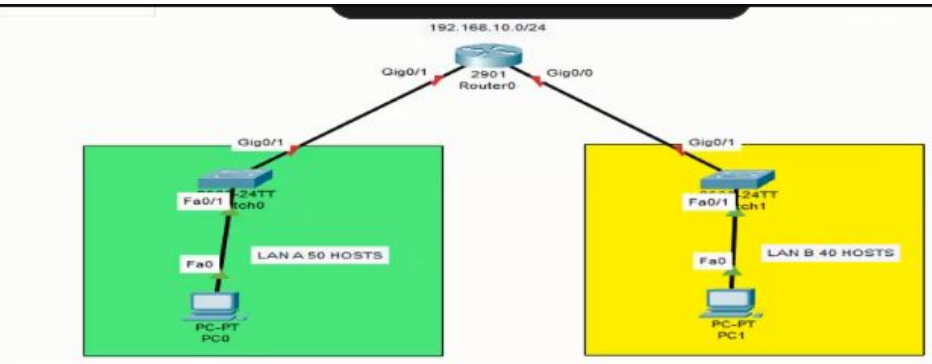
Subnet on an Octet Boundary

$2^{H} - 2$

- Networks are most easily subnetted at the octet boundary of /8, /16, and /24.
- Notice that using longer prefix lengths decreases the number of hosts per subnet.

Prefix Length	Subnet Mask	Subnet Mask in Binary (n = network, h = host)	# of hosts
/8	255.0.0.0	nnnnnnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh 11111111.00000000.00000000.00000000	16,777,214
/16	255.255.0.0	nnnnnnnn.nnnnnnnn.hhhhhhhh.hhhhhhhh 11111111.11111111.00000000.00000000	65,534
/24	255.255.255.0	nnnnnnnn.nnnnnnnn.nnnnnnnn.hhhhhhhh 11111111.11111111.11111111.00000000	254

2^{24}



In this scenario, you are a network technician assigned to install a new network for a customer. You must create multiple subnets out of the 192.168.0.0/24 network address space to meet the following requirements:

a. The first subnet is the LAN-A network. You need a minimum of 50 host IP addresses.

b. The second subnet is the LAN-B network. You need a minimum of 40 host IP addresses.

c. You also need at least two additional unused subnets for future network expansion.

Note: Variable length subnet masks will not be used. All of the device subnet masks should be the same length

NO.OF NETWORKS REQUIRED : 2^N WHERE N IS NO. OF HOST BITS BORROWED

NO.OF HOSTS IN EACH NETWORK/SUBNET : 2^H WHERE H IS NO. OF HOSTS BITS

NO.OF VALID HOSTS : 2^H-2 BECAUSE FIRST AND LAST IP ADDRESS IS RESERVED AS A NETWORK AND BROADCAST ADDRESS

BLOCK SIZE OR RANGE : 256- NEW SUBNET MASK

	Hosts	Netmask	Number of Subnets
/30	4	255.255.255.252	64
/29	8	255.255.255.248	32
/28	16	255.255.255.240	16
/27	32	255.255.255.224	8
/26	64	255.255.255.192	4
/25	128	255.255.255.128	2
/24	256	255.255.255.0	1
/23	512	255.255.254.0	2
/22	1024	255.255.252.0	4
/21	2048	255.255.248.0	8
/20	4096	255.255.240.0	16
/19	8192	255.255.224.0	32
/18	16384	255.255.192.0	64
/17	32768	255.255.128.0	128
/16	65536	255.255.0.0	256

Q1. DIVIDE 192.168.10.0/24 INTO 4 SUBNETS

Q2. HOW MANY HOSTS IN EACH NETWORK

Q3. WHAT IS BLOCK SIZE

Q4. WHAT IS THE BROADCAST AND NETWORK ADDRESS IN EACH RANGE

CLASS C SUBNETTING USING (FLSM) FIXED LENTH SUBNET MASK

192.168.10.0 255.255.255.0 /24 NEW SM: 255.255.255.192/26

PLACE VALUE	128	64	32	16	8	4	2	1
SUBNET MASK	128	192	224	240	248	252	254	255
BITS	1	1	0	0	0	0	0	0

NO.OF NETWORK REQUIRED (4) = 2^N =2^2=4
NO.OF HOSTS IN EACH SUBNET = 2^H=2^6=64
BLOCK SIZE OR RANGE = 256 – NEW SUBNET MASK=256-192=64

PREFIX LENGTH	NETWORK ADDRESS	FIRST USABLE IP ADDRESS	LAST USABLE IP ADDRESS	BROADCAST ADDRESS
/26	192.168.10.0			192.168.10.63
/26	192.168.10.64			
/26	192.168.10.128			
/26	192.168.10.192			

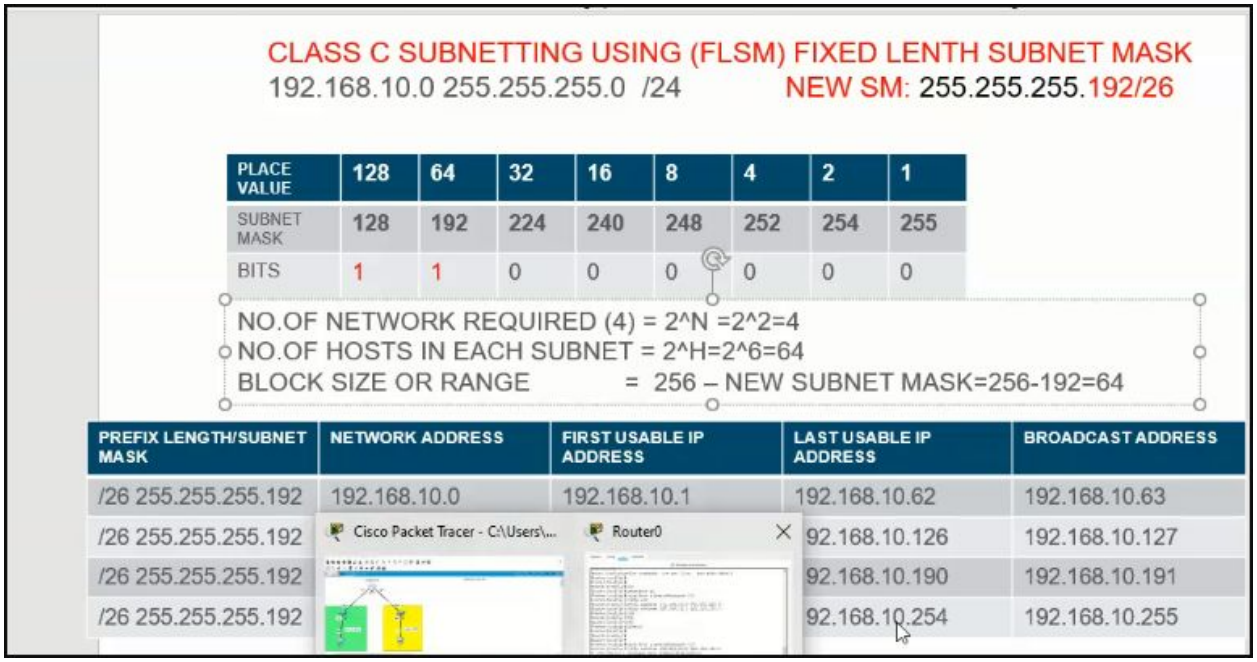
CLASS C SUBNETTING USING (FLSM) FIXED LENTH SUBNET MASK

192.168.10.0 255.255.255.0 /24 NEW SM: 255.255.255.192/26

PLACE VALUE	128	64	32	16	8	4	2	1
SUBNET MASK	128	192	224	240	248	252	254	255
BITS	1	1	0	0	0	0	0	0

NO.OF NETWORK REQUIRED (4) = 2^N =2^2=4
NO.OF HOSTS IN EACH SUBNET = 2^H=2^6=64
BLOCK SIZE OR RANGE = 256 – NEW SUBNET MASK=256-192=64

PREFIX LENGTH/SUBNET MASK	NETWORK ADDRESS	FIRST USABLE IP ADDRESS	LAST USABLE IP ADDRESS	BROADCAST ADDRESS
/26 255.255.255.192	192.168.10.0	192.168.10.1	192.168.10.62	192.168.10.63
/26	192.168.10.64			192.168.10.127
/26	192.168.10.128			192.168.10.191
/26	192.168.10.192			192.168.10.255



CLASS C SUBNETTING USING (FLSM) FIXED LENTH SUBNET MASK

192.168.10.0 255.255.255.0 /24 NEW SM: 255.255.255.192/26

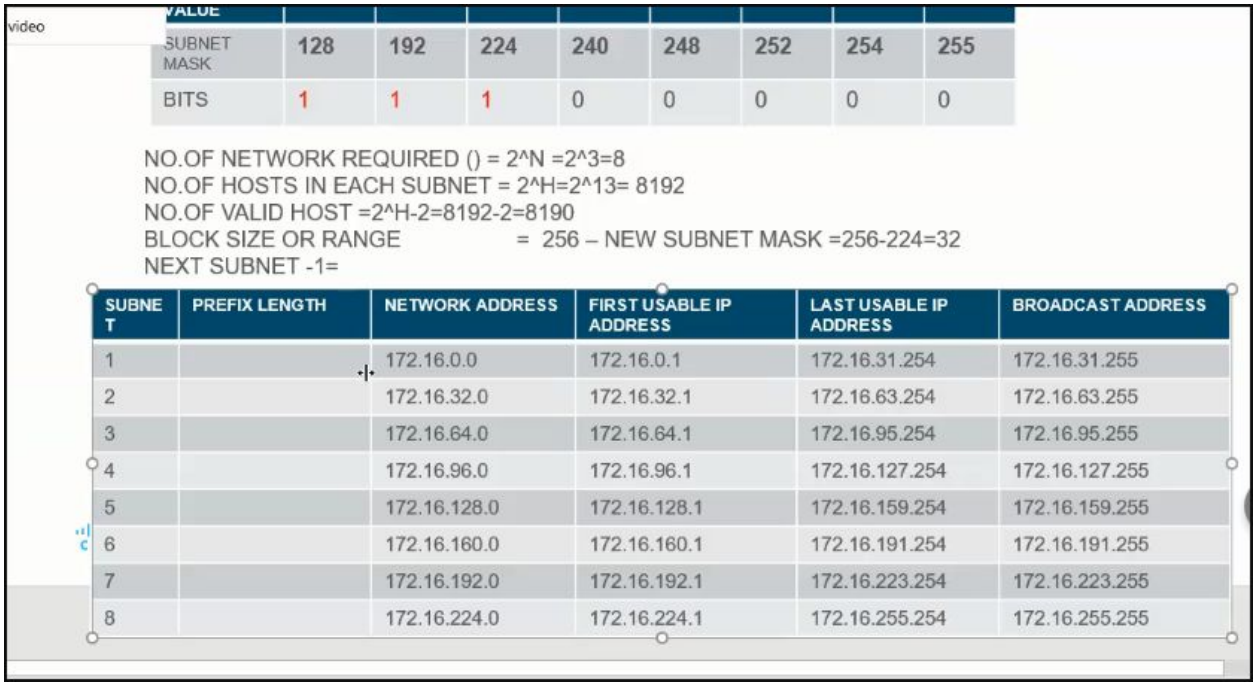
PLACE VALUE	128	64	32	16	8	4	2	1
SUBNET MASK	128	192	224	240	248	252	254	255
BITS	1	1	0	0	0	0	0	0

NO.OF NETWORK REQUIRED (4) = 2^N =2^2=4
NO.OF HOSTS IN EACH SUBNET = 2^H=2^6=64
BLOCK SIZE OR RANGE = 256 – NEW SUBNET MASK=256-192=64

PREFIX LENGTH/SUBNET MASK	NETWORK ADDRESS	FIRST USABLE IP ADDRESS	LAST USABLE IP ADDRESS	BROADCAST ADDRESS
/26 255.255.255.192	192.168.10.0	192.168.10.1	192.168.10.62	192.168.10.63
/26 255.255.255.192	192.168.10.64	192.168.10.65	192.168.10.126	192.168.10.127
/26 255.255.255.192	192.168.10.128	192.168.10.129	192.168.10.190	192.168.10.191
/26 255.255.255.192	192.168.10.192	192.168.10.193	192.168.10.254	192.168.10.255

Cisco Packet Tracer - C:\Users\... Router0

CLASS B subnetting



video

VALUE								
SUBNET MASK	128	192	224	240	248	252	254	255
BITS	1	1	1	0	0	0	0	0

NO.OF NETWORK REQUIRED () = 2^N =2^3=8
NO.OF HOSTS IN EACH SUBNET = 2^H=2^13= 8192
NO.OF VALID HOST =2^H-2=8192-2=8190
BLOCK SIZE OR RANGE = 256 – NEW SUBNET MASK =256-224=32
NEXT SUBNET -1=

SUBNET	PREFIX LENGTH	NETWORK ADDRESS	FIRST USABLE IP ADDRESS	LAST USABLE IP ADDRESS	BROADCAST ADDRESS
1		172.16.0.0	172.16.0.1	172.16.31.254	172.16.31.255
2		172.16.32.0	172.16.32.1	172.16.63.254	172.16.63.255
3		172.16.64.0	172.16.64.1	172.16.95.254	172.16.95.255
4		172.16.96.0	172.16.96.1	172.16.127.254	172.16.127.255
5		172.16.128.0	172.16.128.1	172.16.159.254	172.16.159.255
6		172.16.160.0	172.16.160.1	172.16.191.254	172.16.191.255
7		172.16.192.0	172.16.192.1	172.16.223.254	172.16.223.255
8		172.16.224.0	172.16.224.1	172.16.255.254	172.16.255.255

MASK								
BITS	1	1	1	0	0	0	0	0

NO.OF NETWORK REQUIRED () = 2^N =2^3=8
NO.OF HOSTS IN EACH SUBNET = 2^H=2^13= 8192
NO.OF VALID HOST =2^H-2=8192-2=8190
BLOCK SIZE OR RANGE = 256 – NEW SUBNET MASK =256-224=32
NEXT SUBNET -1=

SUBNET	PREFIX LENGTH	NETWORK ADDRESS	FIRST USABLE IP ADDRESS	LAST USABLE IP ADDRESS	BROADCAST ADDRESS
1	/19 255.255.224.0	172.16.0.0	172.16.0.1	172.16.31.254	172.16.31.255
2	/19 255.255.224.0	172.16.32.0	172.16.32.1	172.16.63.254	172.16.63.255
3	/19 255.255.224.0	172.16.64.0	172.16.64.1	172.16.95.254	172.16.95.255
4	/19 255.255.224.0	172.16.96.0	172.16.96.1	172.16.127.254	172.16.127.255
5	/19 255.255.224.0	172.16.128.0	172.16.128.1	172.16.159.254	172.16.159.255
6	/19 255.255.224.0	172.16.160.0	172.16.160.1	172.16.191.254	172.16.191.255
7	/19 255.255.224.0	172.16.192.0	172.16.192.1	172.16.223.254	172.16.223.255
8	/19 255.255.224.0	172.16.224.0	172.16.224.1	172.16.255.254	172.16.255.255

Requirement = 100, 60, 30, 10, 2

Subnetting using VLSM

128	64	32	16	8	4	2	1	128	64	32	16	8	4	2	1	128	64	32	16	8	4	2	1	128	64	32	16	8	4	2	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
255								255								255								240							
8								8								8								4							

Requirement (10)= $2^h - 2 = 2^4$ - 2 = 16 -2 = 14

Block size or range = $2^h = 2^4$ = 16

192.168.10.0 255.255.255.0

Prefix length	Network address	First usable ip address	Last usable ip address	Broadcast addresss	Subnet mask
/25	192.168.10.0	192.168.10.1	192.168.10.126	192.168.10.127	255.255.255.128
/26	192.168.10.128	192.168.10.129	192.168.10.190	192.168.10.191	255.255.255.192
/27	192.168.10.192	192.168.10.193	192.168.10.222	192.168.10.223	255.255.255.224
/28	192.168.10.224	192.168.10.225	192.168.10.238	192.168.10.239	255.255.255.240
	192.168.10.240				

Hosts Needed	Subnet Mask (binary)	Subnet Mask (decimal)	Prefix Notation(/x)
250	11111111.11111111.11111111.00000000	255.255.255.0	/24
25	11111111 . 11111111 . 11111111 . 11100000	255.255.255.224	/ 27
1000	11111111 . 11111111 . 11111100 . 00000000	255.255.252.0	/ 22
75	11111111 . 11111111 . 11111111 . 10000000	255.255.255.128	/ 25
10	11111111 . 11111111 . 11111111 . 11110000	255.255.255.240	/ 28
500	11111111 . 11111111 . 11111110 . 00000000	255.255.254.0	/ 23

- **End user clients** - Most networks allocate IPv4 addresses to client devices dynamically, using Dynamic Host Configuration Protocol (DHCP). This reduces the burden on network support staff and virtually eliminates entry errors. With DHCP, addresses are only leased for a period of time, and can be reused when the lease expires. This is an important feature for networks that support transient users and wireless devices. Changing the subnetting scheme means that the DHCP server needs to be reconfigured, and the clients must renew their IPv4 addresses. IPv6 clients can obtain address information using DHCPv6 or SLAAC.
- **Servers and peripherals** - These should have a predictable static IP address. Use a consistent numbering system for these devices.
- **Servers that are accessible from the internet** - Servers that need to be publicly available on the internet must have a public IPv4 address, most often accessed using NAT. In some organizations, internal servers (not publicly available) must be made available to the remote users. In most cases, these servers are assigned private addresses internally, and the user is required to create a virtual private network (VPN) connection to access the server. This has the same effect as if the user is accessing the server from a host within the intranet.
- **Intermediary devices** - These devices are assigned addresses for network management, monitoring, and security. Because we must know how to communicate with intermediary devices, they should have predictable, statically assigned addresses.
- **Gateway** - Routers and firewall devices have an IP address assigned to each interface which serves as the gateway for the hosts in that network. Typically, the router interface uses either the lowest or highest address in the network.

When developing an IP addressing scheme, it is generally recommended that you have a set pattern of how addresses are allocated to each type of device. This benefits administrators when adding and removing devices, filtering traffic based on IP, as well as simplifying documentation.

What did I learn in this module?

IPv4 Addressing Structure

An IPv4 address is a 32-bit hierarchical address that is made up of a network portion and a host portion. The bits within the network portion of the address must be identical for all devices that reside in the same network. The bits within the host portion of the address must be unique to identify a specific host within a network. A host requires a unique IPv4 address and a subnet mask to show the network/host portions of the address. The prefix length is the number of bits set to 1 in the subnet mask. It is written in “slash notation”, which is a “/” followed by the number of bits set to 1. Logical AND is the comparison of two bits. Only a 1 AND 1 produces a 1 and all other combination results in a 0. Any other combination results in a 0. Within each network there are network addresses, host addresses, and a broadcast address.

IPv4 Unicast, Broadcast, and Multicast

Unicast transmission refers to a device sending a message to one other device in one-to-one communications. A unicast packet is a packet with a destination IP address that is a unicast address which is the address of a single recipient. Broadcast transmission refers to a device sending a message to all the devices on a network in one-to-all communications. A broadcast packet has a destination IP address with all ones (1s) in the host portion, or 32 one (1) bits. Multicast transmission reduces traffic by allowing a host to send a single packet to a selected set of hosts that subscribe to a multicast group. A multicast packet is a packet with a destination IP address that is a multicast address. IPv4 has reserved the 224.0.0.0 to 239.255.255.255 addresses as a multicast range.

Types of IPv4 Addresses

Public IPv4 addresses are globally routed between ISP routers. Not all available IPv4 addresses can be used on the internet. There are blocks of addresses called private addresses that are used by most organizations to assign IPv4 addresses to internal hosts. Most internal networks use private IPv4 addresses for addressing all internal devices (intranet); however, these private addresses are not globally routable. Loopback addresses used by a host to direct traffic back to itself. Link-local addresses are more commonly known as APIPA addresses, or self-assigned addresses. In 1981, IPv4 addresses were assigned using classful addressing: A, B, or C. Public IPv4 addresses must be unique, and are globally routed over the internet. Both IPv4 and IPv6 addresses are managed by the IANA, which allocates blocks of IP addresses to the RIRs.

Network Segmentation

In an Ethernet LAN, devices broadcast to locate other devices using ARP. Switches propagate broadcasts out all interfaces except the interface on which it was received. Routers do not propagate broadcasts, instead each router interface connects a broadcast domain and broadcasts are only propagated within that specific domain. A large broadcast domain is a network that connects many hosts. A problem with a large broadcast domain is that these hosts can generate excessive broadcasts and negatively affect the network. The solution is to reduce the size of the network to create smaller broadcast domains in a process called subnetting.

These smaller network spaces are called subnets. Subnetting reduces overall network traffic and improves network performance. An administrator may subnet by location, between networks, or by device type.

Subnet an IPv4 Network

IPv4 subnets are created by using one or more of the host bits as network bits. This is done by extending the subnet mask to borrow some of the bits from the host portion of the address to create additional network bits. The more host bits that are borrowed, the more subnets that can be defined. The more bits that are borrowed to increase the number of subnets also reduces the number of hosts per subnet. Networks are most easily subnetted at the octet boundary of /8, /16, and /24. Subnets can borrow bits from any host bit position to create other masks.

Subnet a /16 and a /8 Prefix

In a situation requiring a larger number of subnets, an IPv4 network is required that has more hosts bits available to borrow. To create subnets, you must borrow bits from the host portion of the IPv4 address of the existing internetwork. Starting from the left to the right with the first available host bit, borrow a single bit at a time until you reach the number of bits necessary to create the number of subnets required. When borrowing bits from a /16 address, start borrowing bits in the third octet, going from left to right. The first address is reserved for the network address and the last address is reserved for the broadcast address.

Subnet to Meet Requirements

A typical enterprise network contains an intranet and a DMZ. Both have subnetting requirements and challenges. The intranet uses private IPv4 addressing space. The 10.0.0.0/8 can also be subnetted using any other number of prefix lengths, such as /12, /18, /20, etc., giving the network administrator many options. Because these devices need to be publicly accessible from the internet, the devices in the DMZ require public IPv4 addresses. Organizations must maximize their own limited number of public IPv4 addresses. To reduce the number of unused host addresses per subnet, the network administrator must subnet their public address space into subnets with different subnet masks. This is known as Variable Subnet Length Masking (VLSM). Administrators must consider how many host addresses are required for each network, and how many subnets are needed.

Variable Length Subnet Masking

Traditional subnetting might meet an organization’s needs for its largest LAN and divide the address space into an adequate number of subnets. But it likely also results in significant waste of unused addresses. VLSM allows a network space to be divided into unequal parts. With VLSM, the subnet mask will vary depending on how many bits have been borrowed for a particular subnet (this is the “variable” part of the VLSM). VLSM is just subnetting a subnet. When using VLSM, always begin by satisfying the host requirements of the largest subnet. Continue subnetting until the host requirements of the smallest subnet are satisfied. Subnets always need to be started on an appropriate bit boundary.

Structured Design

A network administrator should study the network requirements to better plan how the IPv4 network subnets will be structured. This means looking at the entire network, both the intranet and the DMZ, and determining how each area will be segmented. The address plan includes determining where address conservation is needed (usually within the DMZ), and where there is more flexibility (usually within the intranet). Where address conservation is required the plan should determine how many subnets are needed and how many hosts per subnet. This is usually required for public IPv4 address space within the DMZ. This will most likely include using VLSM. The address plan includes how host addresses will be assigned, which hosts will require static IPv4 addresses, and which hosts can use DHCP for obtaining their addressing information. Within a network, there are different types of devices that require addresses: end user clients, servers and peripherals, servers that are accessible from the internet, intermediary devices, and gateways. When developing an IP addressing scheme, have a set pattern of how addresses are allocated to each type of device. This helps when adding and removing devices, filtering traffic based on IP, as well as simplifying documentation.

CHAPTER 12

IPv6 is designed to be the successor to IPv4. IPv6 has a larger 128-bit address space, providing 340 undecillion (i.e., 340 followed by 36 zeroes) possible addresses. However, IPv6 is more than just larger addresses.

When the IETF began its development of a successor to IPv4, it used this opportunity to fix the limitations of IPv4 and include enhancements. One example is Internet Control Message Protocol version 6 (ICMPv6), which includes address resolution and address autoconfiguration not found in ICMP for IPv4 (ICMPv4).

The depletion of IPv4 address space has been the motivating factor for moving to IPv6. As Africa, Asia and other areas of the world become more connected to the internet, there are not enough IPv4 addresses to accommodate this growth. As shown in the figure, four out of the five RIRs have run out of IPv4 addresses.

The IETF has created various protocols and tools to help network administrators migrate their networks to IPv6. The migration techniques can be divided into three categories:
1. Dual stack allows IPv4 and IPv6 to coexist on the same network segment. Dual stack devices run both IPv4 and IPv6 protocol stacks simultaneously. Known as native IPv6, this means the customer network has an IPv6 connection to their ISP and is able to access content found on the internet over IPv6.
2. Tunneling is a method of transporting an IPv6 packet over an IPv4 network. The IPv6 packet is encapsulated inside an IPv4 packet, similar to other types of data.
3. Network Address Translation 64 (NAT64) allows IPv6-enabled devices to communicate with

IPv4-enabled devices using a translation technique similar to NAT for IPv4. An IPv6 packet is translated to an IPv4 packet and an IPv4 packet is translated to an IPv6 packet.

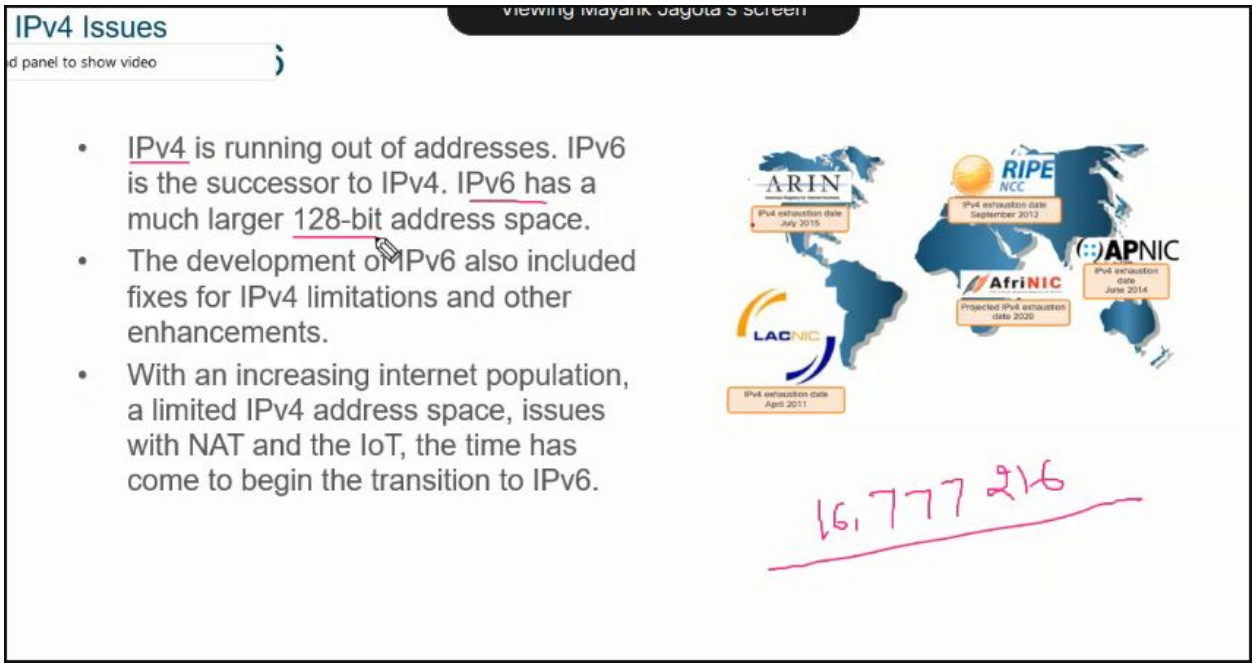
Tunneling and translation are for transitioning to native IPv6 and should only be used where needed. The goal should be native IPv6 communications from source to destination.

Good job!

You have successfully identified the correct answers.

1. The main driver or most important factor for IPv6 is the depletion of the IPv4 address space.
2. The correct answer is True. Four of the five RIRs, ARIN, APNIC, LACNIC, and RIPENCC have exhausted their IPv4 address pools. Only AfriNIC has remaining IPv4 address space to allocate to customers.
3. Only dual stack uses native IPv6 connectivity.

You answered 3 out of 3 questions correctly.



Rule 1 – Omit Leading Zeros

Rule 2- Double Colon

The second rule to help reduce the notation of IPv6 addresses is that a double colon (::) can replace any single, contiguous string of one or more 16-bit hextets consisting of all zeros. For example, 2001:db8:cafe:1:0:0:0:1 (leading 0s omitted) could be represented as 2001:db8:cafe:1::1. The double colon (::) is used in place of the three all-0 hextets (0:0:0).

The double colon (::) can only be used once within an address, otherwise there would be more than one possible resulting address. When used with the omitting leading 0s technique, the notation of IPv6 address can often be greatly reduced. This is commonly known as the compressed format.

there are three broad categories of IPv6 addresses:

- **Unicast** - An IPv6 unicast address uniquely identifies an interface on an IPv6-enabled device.
- **Multicast** - An IPv6 multicast address is used to send a single IPv6 packet to multiple destinations.
- **Anycast** - An IPv6 anycast address is any IPv6 unicast address that can be assigned to multiple devices. A packet sent to an anycast address is routed to the nearest device having that address. Anycast addresses are beyond the scope of this course.

Unlike IPv4, IPv6 does not have a broadcast address. However, there is an IPv6 all-nodes multicast address that essentially gives the same result.

It is strongly recommended to use a 64-bit Interface ID for most networks. This is because stateless address autoconfiguration (SLAAC) uses 64 bits for the Interface ID. It also makes subnetting easier to create and manage.

RS and RA Messages

Devices obtain GUA addresses dynamically through Internet Control Message Protocol version 6 (ICMPv6) messages.

- Router Solicitation (RS) messages are sent by host devices to discover IPv6 routers
- Router Advertisement (RA) messages are sent by routers to inform hosts on how to obtain an IPv6 GUA and provide useful network information such as:
  - Network prefix and prefix length
  - Default gateway address
  - DNS addresses and domain name
- The RA can provide three methods for configuring an IPv6 GUA :
  - SLAAC
  - SLAAC with stateless DHCPv6 server
  - Stateful DHCPv6 (no SLAAC)

IPv4 and IPv6 Coexistence

Both IPv4 and IPv6 will coexist in the near future and the transition will take several years.

The IETF has created various protocols and tools to help network administrators migrate their networks to IPv6. These migration techniques can be divided into three categories:

- **Dual stack** -The devices run both IPv4 and IPv6 protocol stacks simultaneously.
- **Tunneling** – A method of transporting an IPv6 packet over an IPv4 network. The IPv6 packet is encapsulated inside an IPv4 packet.
- **Translation** - Network Address Translation 64 (NAT64) allows IPv6-enabled devices to communicate with IPv4-enabled devices using a translation technique similar to NAT for IPv4.

Note: Tunneling and translation are for transitioning to native IPv6 and should only be used where needed. The goal should be native IPv6 communications from source to destination.

IPv6 Addressing Formats

- IPv6 addresses are 128 bits in length and written in hexadecimal.
- IPv6 addresses are not case-sensitive and can be written in either lowercase or uppercase.
- The preferred format for writing an IPv6 address is x:x:x:x:x:x:x:x, with each "x" consisting of four hexadecimal values.
- In IPv6, a hextet is the unofficial term used to refer to a segment of 16 bits, or four hexadecimal values.
- Examples of IPv6 addresses in the preferred format:

```
2001:0db8:0000:1111:0000:0000:0000:0200
2001:0db8:0000:00a3:abcd:0000:0000:1234
```


Rule 1 – Omit Leading Zero

The first rule to help reduce the notation of IPv6 addresses is to omit any leading 0s (zeros).

Examples:

- 01ab can be represented as 1ab
- 09f0 can be represented as 9f0
- 0a00 can be represented as a00
- 00ab can be represented as ab

Note: This rule only applies to leading 0s, NOT to trailing 0s, otherwise the address would be ambiguous.

Type	Format
Preferred	2001 : 0db8 : 0000 : 1111 : 0000 : 0000 : 0000 : 0200
No leading zeros	2001 : db8 : 0 : 1111 : 0 : 0 : 0 : 200

Rule 2 – Double Colon

A double colon (::) can replace any single, contiguous string of one or more 16-bit hextets consisting of all zeros.

Example:

- 2001:db8:cafe:1:0:0:0:1 (leading 0s omitted) could be represented as 2001:db8:cafe:1::1

Note: The double colon (::) can only be used once within an address, otherwise there would be more than one possible resulting address.

Type	Format
Preferred	2001 : 0db8 : 0000 : 1111 : 0000 : 0000 : 0000 : 0200
Compressed	2001:db8:0:1111::200

Unicast, Multicast, Anycast

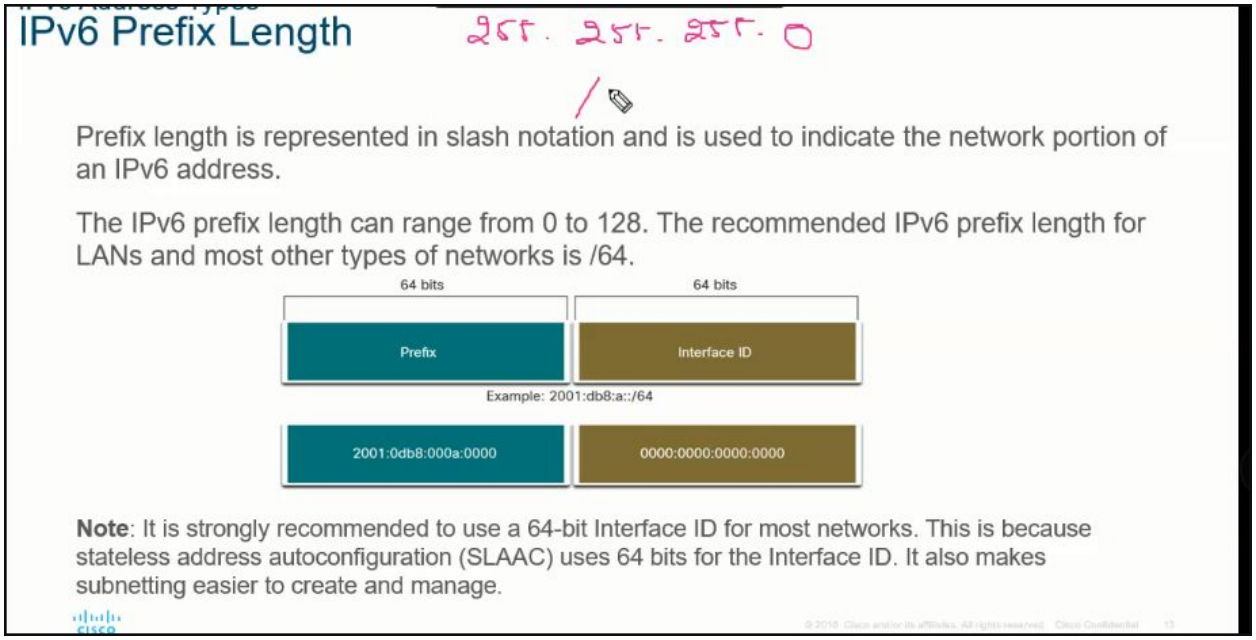
1 to all

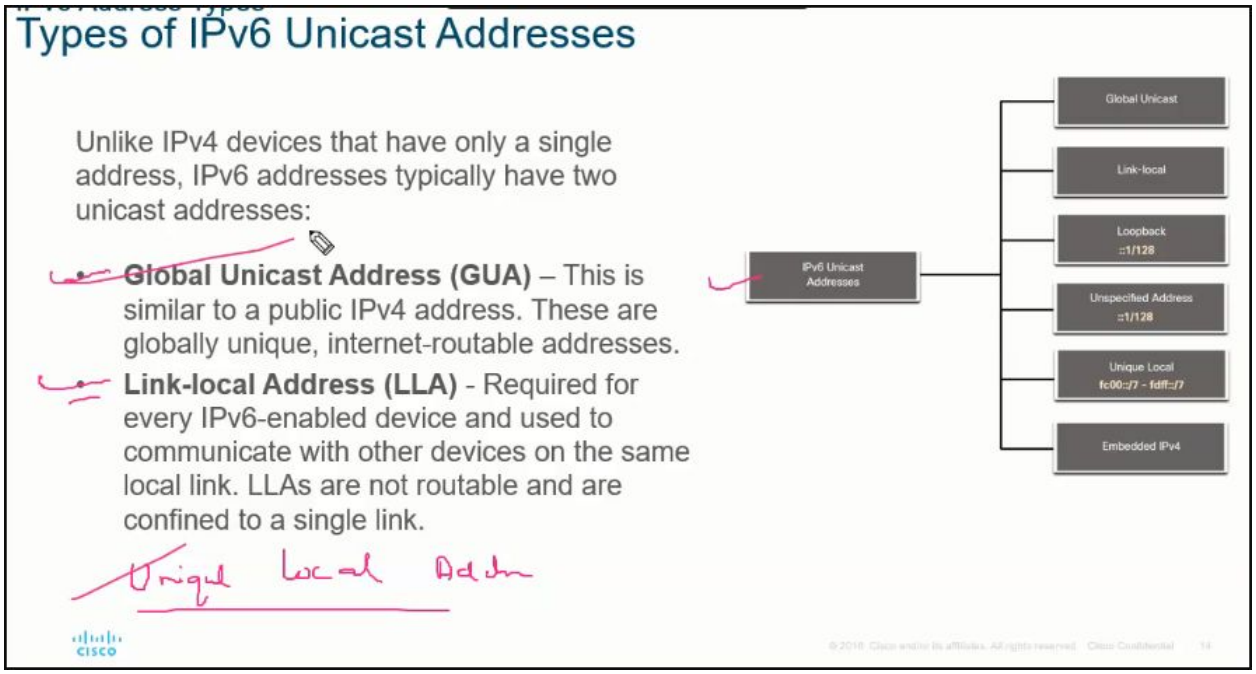
1 to nearest

There are three broad categories of IPv6 addresses:

- **Unicast** – Unicast uniquely identifies an interface on an IPv6-enabled device.
- **Multicast** – Multicast is used to send a single IPv6 packet to multiple destinations.
- **Anycast** – This is any IPv6 unicast address that can be assigned to multiple devices. A packet sent to an anycast address is routed to the nearest device having that address.

Note: Unlike IPv4, IPv6 does not have a broadcast address. However, there is an IPv6 all-nodes multicast address that essentially gives the same result.

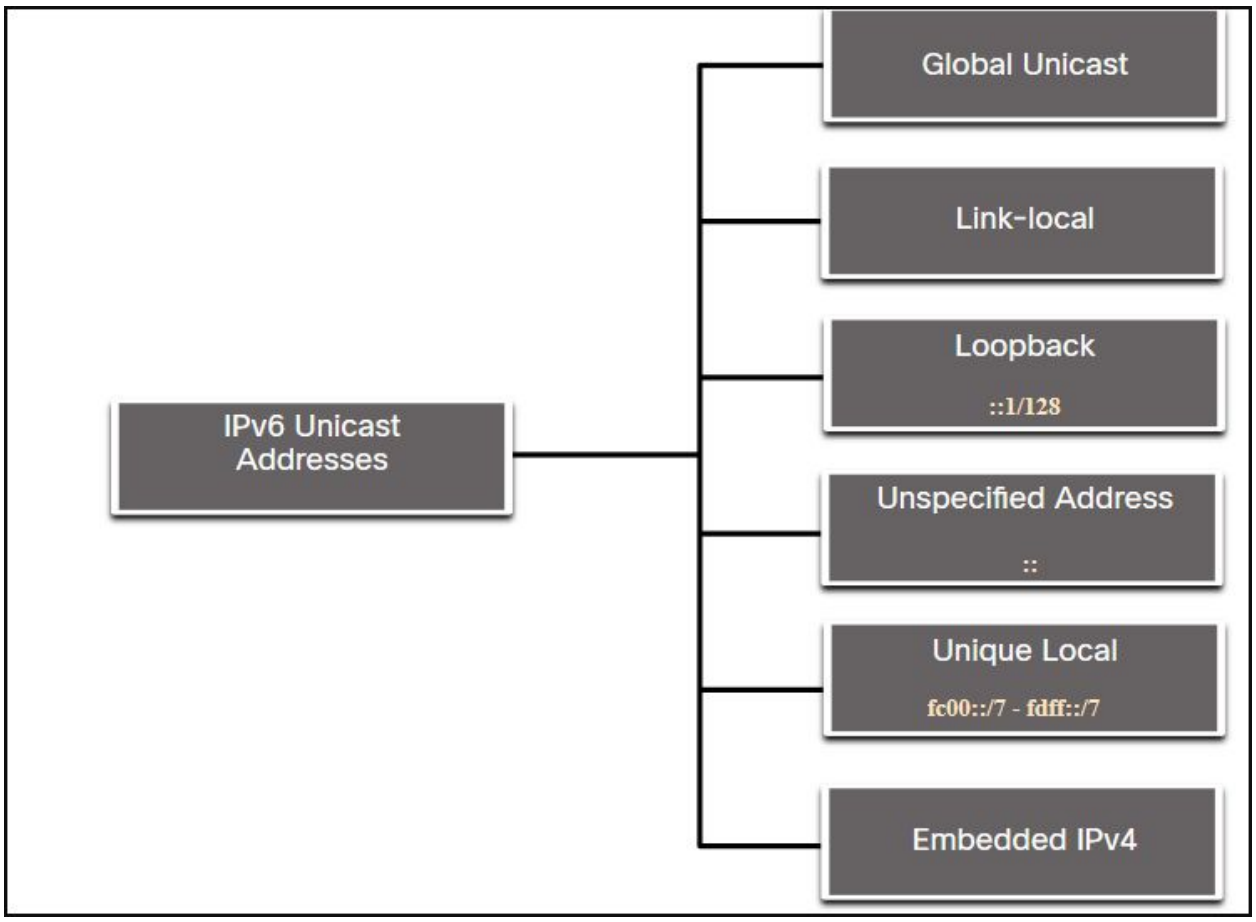




Unlike IPv4 devices that have only a single address, IPv6 addresses typically have two unicast addresses:

- **Global Unicast Address (GUA)** - This is similar to a public IPv4 address. These are globally unique, internet-routable addresses. GUAs can be configured statically or assigned dynamically.
- **Link-local Address (LLA)** - This is required for every IPv6-enabled device. LLAs are used to communicate with other devices on the same local link. With IPv6, the term link refers to a subnet. LLAs are confined to a single link. Their uniqueness must only be

confirmed on that link because they are not routable beyond the link. In other words, routers will not forward packets with a link-local source or destination address.



A Note About the Unique Local Address

Unique local addresses (range fc00::/7 to fdff::/7) are not yet commonly implemented. Therefore, this module only covers GUA and LLA configuration. However, unique local addresses may eventually be used to address devices that should not be accessible from the outside, such as internal servers and printers.

The IPv6 unique local addresses have some similarity to RFC 1918 private addresses for IPv4, but there are significant differences:

- Unique local addresses are used for local addressing within a site or between a limited number of sites.
- Unique local addresses can be used for devices that will never need to access another network.
- Unique local addresses are not globally routed or translated to a global IPv6 address.

Note: Many sites also use the private nature of RFC 1918 addresses to attempt to secure or hide their network from potential security risks. However, this was never the intended use of these technologies, and the IETF has always recommended that sites take the proper security precautions on their internet-facing router.

A Note About the Unique Local Address

The IPv6 unique local addresses (range fc00::/7 to fdff::/7) have some similarity to RFC 1918 private addresses for IPv4, but there are significant differences:

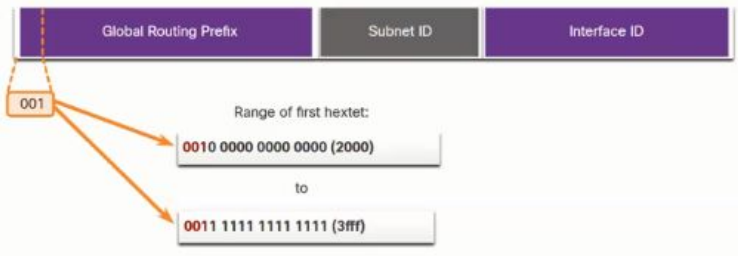
- Unique local addresses are used for local addressing within a site or between a limited number of sites.
- Unique local addresses can be used for devices that will never need to access another network.
- Unique local addresses are not globally routed or translated to a global IPv6 address.

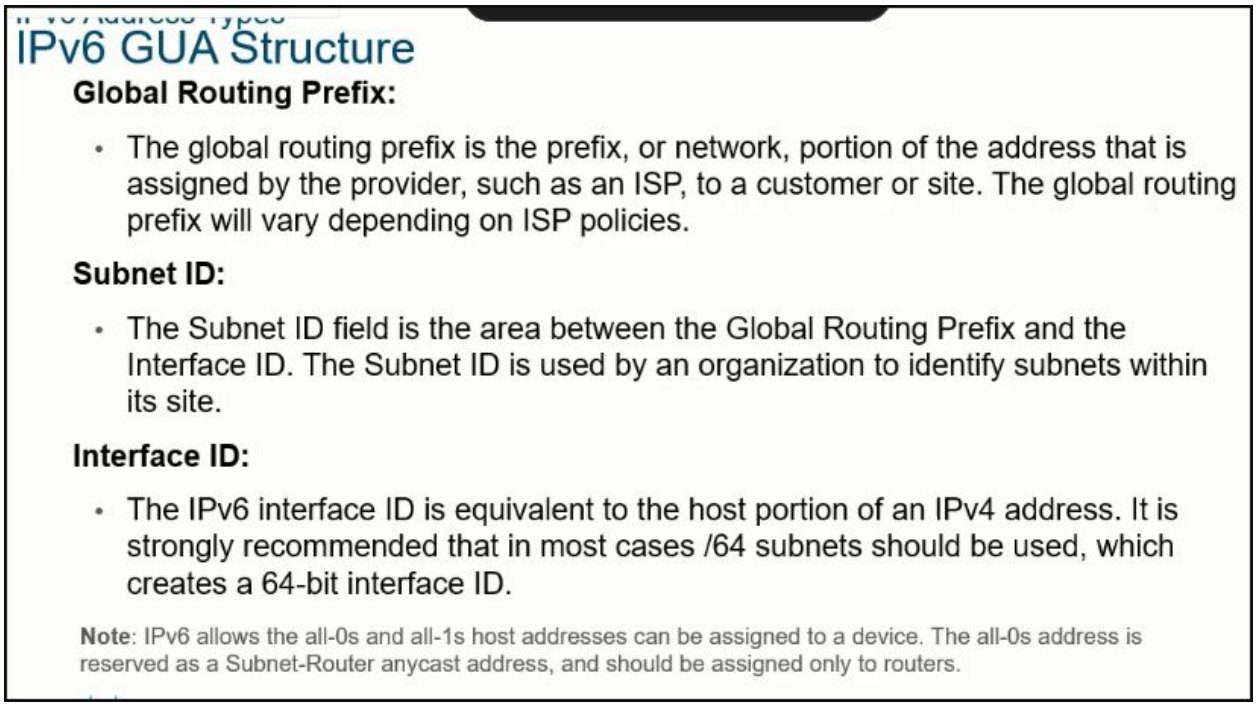
Note: Many sites use the private nature of RFC 1918 addresses to attempt to secure or hide their network from potential security risks. This was never the intended use of ULAs.

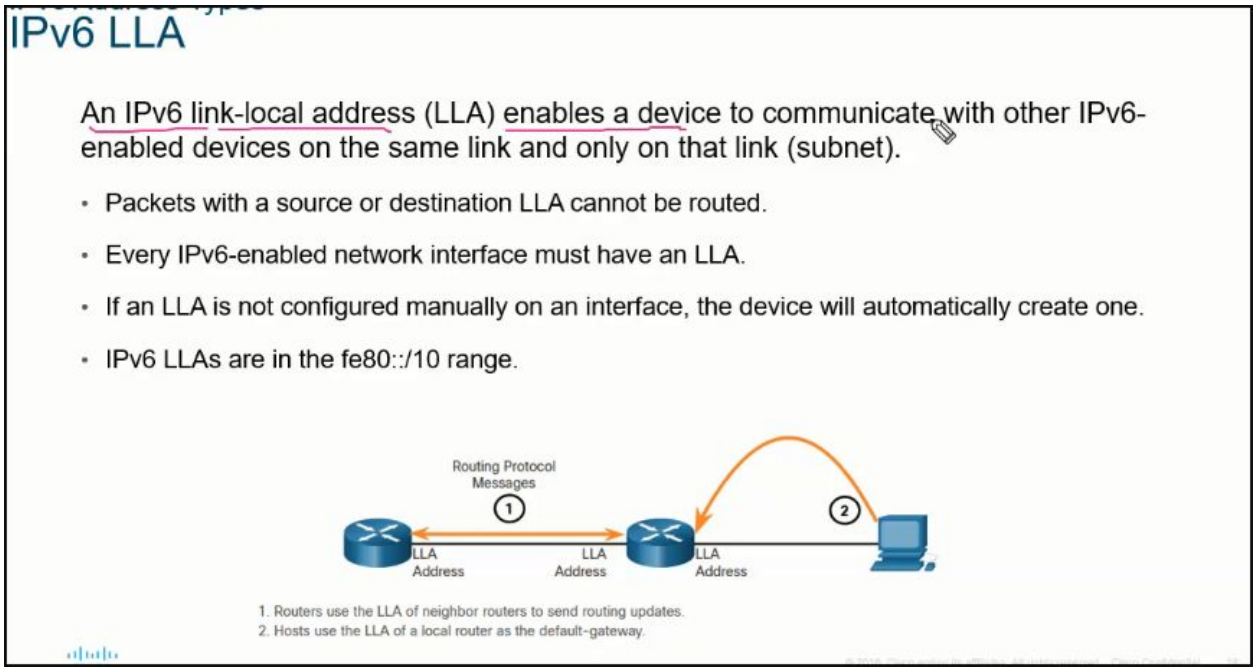
IPv6 GUA

IPv6 global unicast addresses (GUAs) are globally unique and routable on the IPv6 internet.

- Currently, only GUAs with the first three bits of 001 or 2000::/3 are being assigned.
- Currently available GUAs begins with a decimal 2 or a 3 (This is only 1/8th of the total available IPv6 address space).







IPv6 GUA Structure

Global Routing Prefix

The global routing prefix is the prefix, or network, portion of the address that is assigned by the provider, such as an ISP, to a customer or site. For example, it is common for ISPs to assign a

/48 global routing prefix to its customers. The global routing prefix will usually vary depending on the policies of the ISP.

The previous figure shows a GUA using a /48 global routing prefix. /48 prefixes are a common global routing prefix that is assigned and will be used in most of the examples throughout this course.

For example, the IPv6 address 2001:db8:acad::/48 has a global routing prefix that indicates that the first 48 bits (3 hextets) (2001:db8:acad) is how the ISP knows of this prefix (network). The double colon (::) following the /48 prefix length means the rest of the address contains all 0s. The size of the global routing prefix determines the size of the subnet ID.

Subnet ID

The Subnet ID field is the area between the Global Routing Prefix and the Interface ID. Unlike IPv4 where you must borrow bits from the host portion to create subnets, IPv6 was designed with subnetting in mind. The Subnet ID is used by an organization to identify subnets within its site. The larger the subnet ID, the more subnets available.

Note: Many organizations are receiving a /32 global routing prefix. Using the recommended /64 prefix in order to create a 64-bit Interface ID, leaves a 32 bit Subnet ID. This means an organization with a /32 global routing prefix and a 32-bit Subnet ID will have 4.3 billion subnets, each with 18 quintillion devices per subnet. That is as many subnets as there are public IPv4 addresses!

The IPv6 address in the previous figure has a /48 Global Routing Prefix, which is common among many enterprise networks. This makes it especially easy to examine the different parts of the address. Using a typical /64 prefix length, the first four hextets are for the network portion of the address, with the fourth hextet indicating the Subnet ID. The remaining four hextets are for the Interface ID.

Interface ID

The IPv6 interface ID is equivalent to the host portion of an IPv4 address. The term Interface ID is used because a single host may have multiple interfaces, each having one or more IPv6 addresses. The figure shows an example of the structure of an IPv6 GUA. It is strongly recommended that in most cases /64 subnets should be used, which creates a 64-bit interface ID. A 64-bit interface ID allows for 18 quintillion devices or hosts per subnet.

A /64 subnet or prefix (Global Routing Prefix + Subnet ID) leaves 64 bits for the interface ID. This is recommended to allow SLAAC-enabled devices to create their own 64-bit interface ID. It also makes developing an IPv6 addressing plan simple and effective.

Note: Unlike IPv4, in IPv6, the all-0s and all-1s host addresses can be assigned to a device. The all-1s address can be used because broadcast addresses are not used within IPv6. The

all-0s address can also be used, but is reserved as a Subnet-Router anycast address, and should be assigned only to routers.

12.3.7

IPv6 LLA

An IPv6 link-local address (LLA) enables a device to communicate with other IPv6-enabled devices on the same link and only on that link (subnet). Packets with a source or destination LLA cannot be routed beyond the link from which the packet originated.

The GUA is not a requirement. However, every IPv6-enabled network interface must have an LLA.

If an LLA is not configured manually on an interface, the device will automatically create its own without communicating with a DHCP server. IPv6-enabled hosts create an IPv6 LLA even if the device has not been assigned a global unicast IPv6 address. This allows IPv6-enabled devices to communicate with other IPv6-enabled devices on the same subnet. This includes communication with the default gateway (router).

IPv6 LLAs are in the fe80::/10 range. The /10 indicates that the first 10 bits are 1111 1110 10xx xxxx. The first hextet has a range of 1111 1110 10**00 0000** (fe80) to 1111 1110 10**11 1111** (febf).

Typically, it is the LLA of the router, and not the GUA, that is used as the default gateway for other devices on the link.

There are two ways that a device can obtain an LLA:

- **Statically** - This means the device has been manually configured.
- **Dynamically** - This means the device creates its own interface ID by using randomly generated values or using the Extended Unique Identifier (EUI) method, which uses the client MAC address along with additional bits.

Good job!

You have successfully identified the correct answers.

1. Most IPv6 subnets will have a prefix length of /64.
2. The global routing prefix is the part of a GUA that is assigned by an ISP.
3. Link-local IPv6 addresses are for link only communication and are not routable.
4. The correct answer is False. GUAs do not use a bit from the interface ID to create subnets.
5. Link-local IPv6 addresses start with the prefix fe80.

You answered 5 out of 5 questions correctly.

For example, the Cisco IOS command to configure an IPv4 address on an interface is **ip address** *ip-address subnet-mask.* In contrast, the command to configure an IPv6 GUA on an interface is **ipv6 address** *ipv6-address/prefix-length.*

There are two ways in which a device can obtain an IPv6 GUA automatically:

- *Stateless Address Autoconfiguration (SLAAC)*
- *Stateful DHCPv6*

When DHCPv6 or SLAAC is used, the LLA of the router will automatically be specified as the default gateway address.

Static GUA Configuration on a Router

Most IPv6 configuration and verification commands in the Cisco IOS are similar to their IPv4 counterparts. In many cases, the only difference is the use of **ipv6** in place of **ip** within the commands.

- The command to configure an IPv6 GUA on an interface is: **ipv6 address** *ipv6-address/prefix-length.*
- The example shows commands to configure a GUA on the G0/0/0 interface on R1:

```
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# ipv6 address 2001:db8:acad:1::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
```

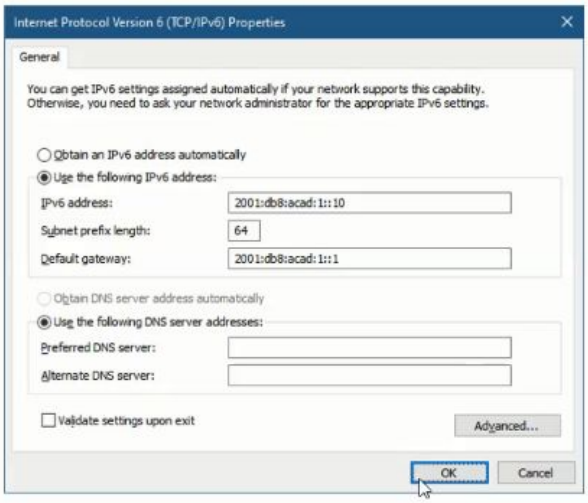
16 16 16 16

2001:db8:acad:1 :: 1

Static GUA Configuration on a Windows Host

- Manually configuring the IPv6 address on a host is similar to configuring an IPv4 address.
- The GUA or LLA of the router interface can be used as the default gateway. Best practice is to use the LLA.

Note: When DHCPv6 or SLAAC is used, the LLA of the router will automatically be specified as the default gateway address.



Static GUA Configuration of a Link-Local Unicast Address

Configuring the LLA manually lets you create an address that is recognizable and easier to remember.

- LLAs can be configured manually using the **ipv6 address** *ipv6-link-local-address* **link-local** command.
- The example shows commands to configure a LLA on the G0/0/0 interface on R1

```
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# ipv6 address fe80::1:1 link-local
R1(config-if)# no shutdown
R1(config-if)# exit
```

Note: The same LLA can be configured on each link as long as it is unique on that link. Common practice is to create a different LLA on each interface of the router to make it easy to identify the router and the specific interface.

Static Configuration of a Link-Local Unicast Address

Configuring the LLA manually lets you create an address that is recognizable and easier to remember. Typically, it is only necessary to create recognizable LLAs on routers. This is beneficial because router LLAs are used as default gateway addresses and in routing advertisement messages.

LLAs can be configured manually using the **ipv6 address** *ipv6-link-local-address* **link-local** command. When an address begins with this hextet within the range of fe80 to febf, the **link-local** parameter must follow the address.

Configure and activate IPv6 on the Gigabit Ethernet 0/0/0 interface with the following addresses:

- Use **g0/0/0** as the interface name
- LLA - fe80::1:1
- GUA - 2001:db8:acad:1::1/64
- Activate the interface
- Exit interface configuration mode

```
R1(config)#int g0/0/0
You must enter the exact and full command.
R1(config)#interface gigabitEthernet0/0/0
You must enter the exact and full command.
R1(config)#interface gigbitEthernet0/0/0
You must enter the exact and full command.
R1(config)#interface gigbitEthernet 0/0/0
You must enter the exact and full command.
R1(config)#interface g0/0/0
R1(config-if)#ip address fe80::1:1 link local
You must enter the exact and full command.
R1(config-if)#ip address fe80::1:1 link-local
You must enter the exact and full command.
R1(config-if)#ipv6 address fe80::1:1 link-local
R1(config-if)#ipv6 address 2001:db8:acad:1::1/64
R1(config-if)#no shutdown
%LINK-3-UPDOWN: Interface GigabitEthernet0/0/0, changed state to up
R1(config-if)#exit
```

Configure and activate IPv6 on the Gigabit Ethernet 0/0/1 interface with the following addresses:

- Use **g0/0/1** as the interface name
- LLA - fe80::2:1
- GUA - 2001:db8:acad:2::1/64
- Activate the interface
- Exit interface configuration mode

```
R1(config)#interface g0/0/1
R1(config-if)#ipv6 address fe80::2:1 link-local
R1(config-if)#ipv6 address 2001:db8:acad:2::1/64
R1(config-if)#no shutdown
%LINK-3-UPDOWN: Interface GigabitEthernet0/0/1, changed state to up
R1(config-if)#exit
```

Configure and activate IPv6 on the serial 0/1/0 interface with the following addresses:

- Use **s0/1/0** as the interface name
- LLA - fe80::3:1
- GUA - 2001:db8:acad:3::1/64
- Activate the interface
- Exit interface configuration mode

R1(config)#interface s0/1/0
R1(config-if)#ipv6 address fe80::3:1 link-local
R1(config-if)#ipv6 address 2001:db8:acad:3::1/64
R1(config-if)#no shutdown
%LINK-3-UPDOWN: Interface Serial0/1/0, changed state to up
R1(config-if)#exit
R1(config)#
You successfully configured IPv6 GUAs on the interfaces of router R1.

RS and RA Messages

If you do not want to statically configure IPv6 GUAs, no need to worry. Most devices obtain their IPv6 GUAs dynamically. This topic explains how this process works using Router Advertisement (RA) and Router Solicitation (RS) messages. This topic gets rather technical, but when you understand the difference between the three methods that a router advertisement can use, as well as how the EUI-64 process for creating an interface ID differs from a randomly generated process, you will have made a huge leap in your IPv6 expertise!

For the GUA, a device obtains the address dynamically through Internet Control Message Protocol version 6 (ICMPv6) messages. IPv6 routers periodically send out ICMPv6 RA messages, every 200 seconds, to all IPv6-enabled devices on the network. An RA message will

also be sent in response to a host sending an ICMPv6 RS message, which is a request for an RA message

RA messages are on IPv6 router Ethernet interfaces. The router must be enabled for IPv6 routing, which is not enabled by default. To enable a router as an IPv6 router, the **ipv6 unicast-routing** global configuration command must be used.

The ICMPv6 RA message is a suggestion to a device on how to obtain an IPv6 GUA. The ultimate decision is up to the device operating system. The ICMPv6 RA message includes the following:

- **Network prefix and prefix length** - This tells the device which network it belongs to.
- **Default gateway address** - This is an IPv6 LLA, the source IPv6 address of the RA message.
- **DNS addresses and domain name** - These are the addresses of DNS servers and a domain name.

There are three methods for RA messages:

- **Method 1: SLAAC** - “I have everything you need including the prefix, prefix length, and default gateway address.”
- **Method 2: SLAAC with a stateless DHCPv6 server** - “Here is my information but you need to get other information such as DNS addresses from a stateless DHCPv6 server.”
- **Method 3: Stateful DHCPv6 (no SLAAC)** - “I can give you your default gateway address. You need to ask a stateful DHCPv6 server for all your other information.”

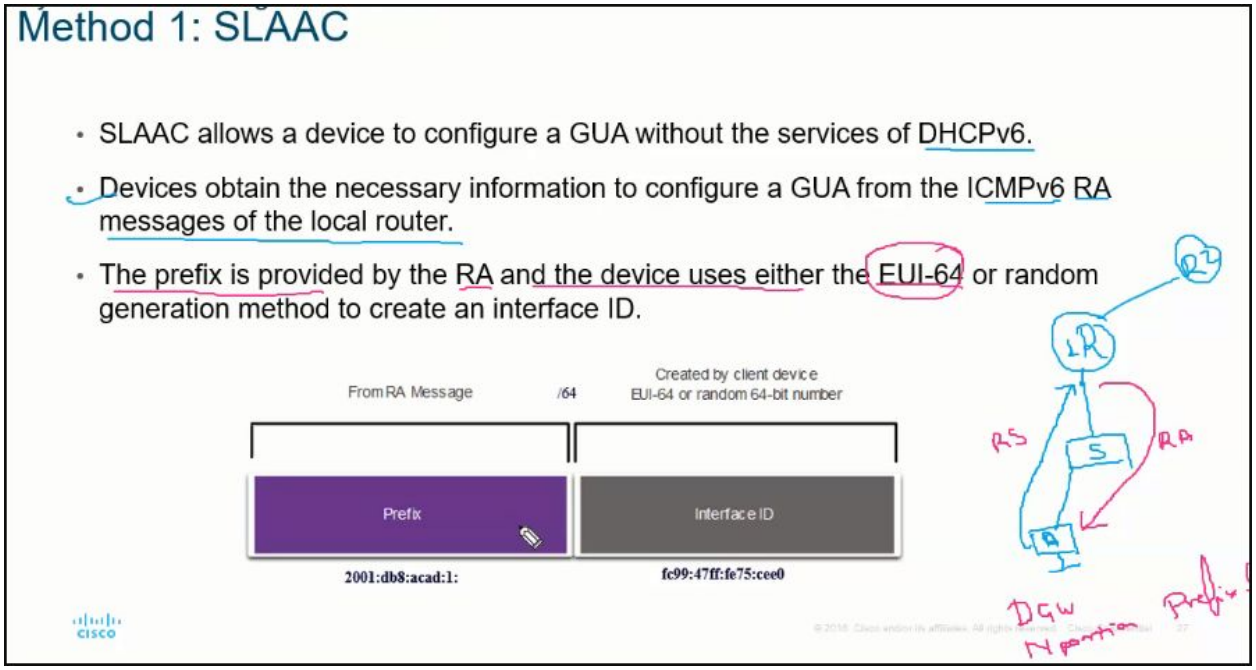
Method 1: SLAAC

SLAAC is a method that allows a device to create its own GUA without the services of DHCPv6. Using SLAAC, devices rely on the ICMPv6 RA messages of the local router to obtain the necessary information.

By default, the RA message suggests that the receiving device use the information in the RA message to create its own IPv6 GUA and all other necessary information. The services of a DHCPv6 server are not required.

SLAAC is stateless, which means there is no central server (for example, a stateful DHCPv6 server) allocating GUAs and keeping a list of devices and their addresses. With SLAAC, the client device uses the information in the RA message to create its own GUA. As shown in the figure, the two parts of the address are created as follows:

- **Prefix** - This is advertised in the RA message.
- **Interface ID** - This uses the EUI-64 process or by generating a random 64-bit number, depending on the device operating system.



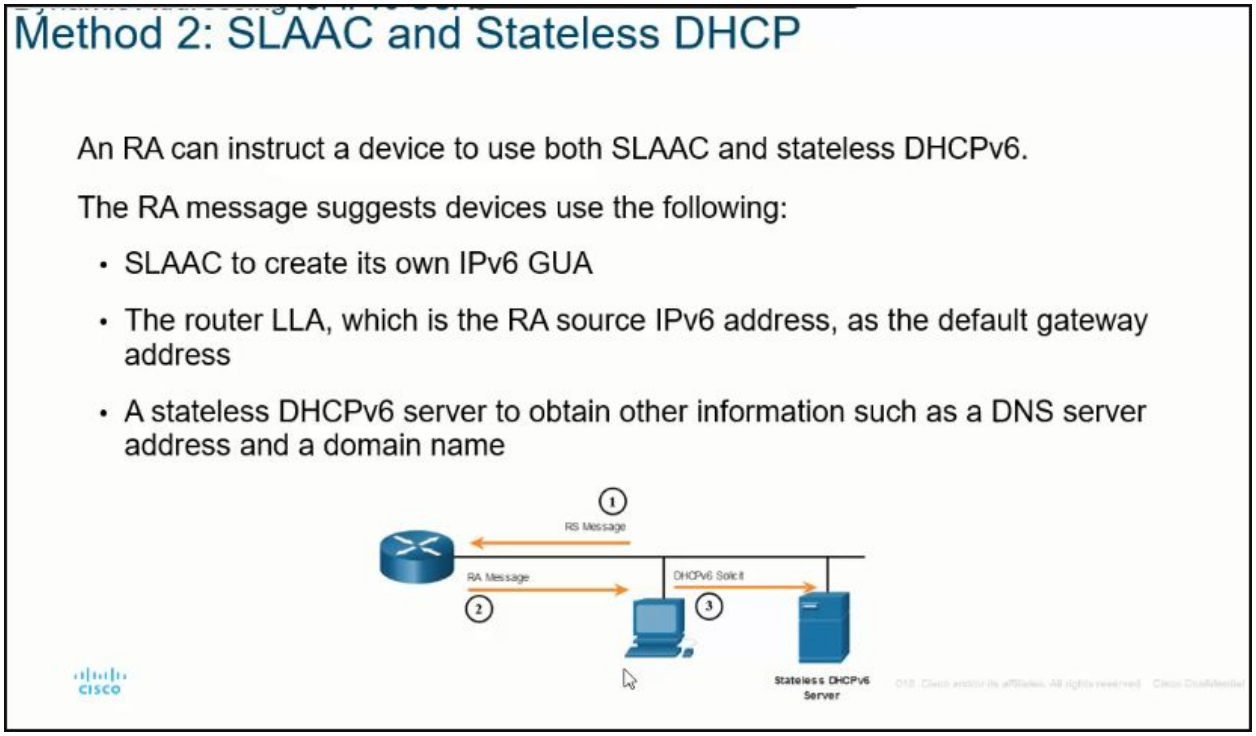
Method 2: SLAAC and Stateless DHCPv6

A router interface can be configured to send a router advertisement using SLAAC and stateless DHCPv6.

As shown in the figure, with this method, the RA message suggests devices use the following:

- SLAAC to create its own IPv6 GUA
- The router LLA, which is the RA source IPv6 address, as the default gateway address
- A stateless DHCPv6 server to obtain other information such as a DNS server address and a domain name

Note: A stateless DHCPv6 server distributes DNS server addresses and domain names. It does not allocate GUAs.



Method 3: Stateful DHCPv6

A router interface can be configured to send an RA using stateful DHCPv6 only.

Stateful DHCPv6 is similar to DHCP for IPv4. A device can automatically receive its addressing information including a GUA, prefix length, and the addresses of DNS servers from a stateful DHCPv6 server.

As shown in the figure, with this method, the RA message suggests devices use the following:

- The router LLA, which is the RA source IPv6 address, for the default gateway address.
- A stateful DHCPv6 server to obtain a GUA, DNS server address, domain name and other necessary information.

A stateful DHCPv6 server allocates and maintains a list of which device receives which IPv6 address. DHCP for IPv4 is stateful.

Note: The default gateway address can only be obtained dynamically from the RA message. The stateless or stateful DHCPv6 server does not provide the default gateway address.

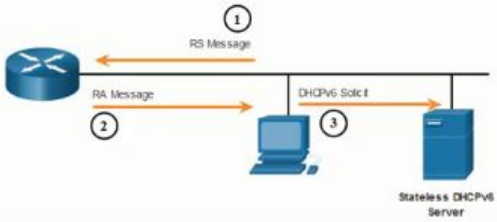
Method 3: Stateful DHCPv6

An RA can instruct a device to use stateful DHCPv6 only.

Stateful DHCPv6 is similar to DHCP for IPv4. A device can automatically receive a GUA, prefix length, and the addresses of DNS servers from a stateful DHCPv6 server.

The RA message suggests devices use the following:

- The router LLA, which is the RA source IPv6 address, for the default gateway address.
- A stateful DHCPv6 server to obtain a GUA, DNS server address, domain name and other necessary information.



EUI-64 Process

The IEEE defined the Extended Unique Identifier (EUI) or modified EUI-64 process which performs the following:

- A 16 bit value of fffe (in hexadecimal) is inserted into the middle of the 48-bit Ethernet MAC address of the client.
- The 7th bit of the client MAC address is reversed from binary 0 to 1.
- Example:

48-bit MAC	fc:99:47:75:ce:e0
EUI-64 Interface ID	fe:99:47:ff:fe:75:ce:e0

EUI-64 Process

IEEE defined the Extended Unique Identifier (EUI) or modified EUI-64 process. This process uses the 48-bit Ethernet MAC address of a client, and inserts another 16 bits in the middle of the 48-bit MAC address to create a 64-bit interface ID.

Ethernet MAC addresses are usually represented in hexadecimal and are made up of two parts:

- **Organizationally Unique Identifier (OUI)** - The OUI is a 24-bit (6 hexadecimal digits) vendor code assigned by IEEE.
- **Device Identifier** - The device identifier is a unique 24-bit (6 hexadecimal digits) value within a common OUI.

An EUI-64 Interface ID is represented in binary and is made up of three parts:

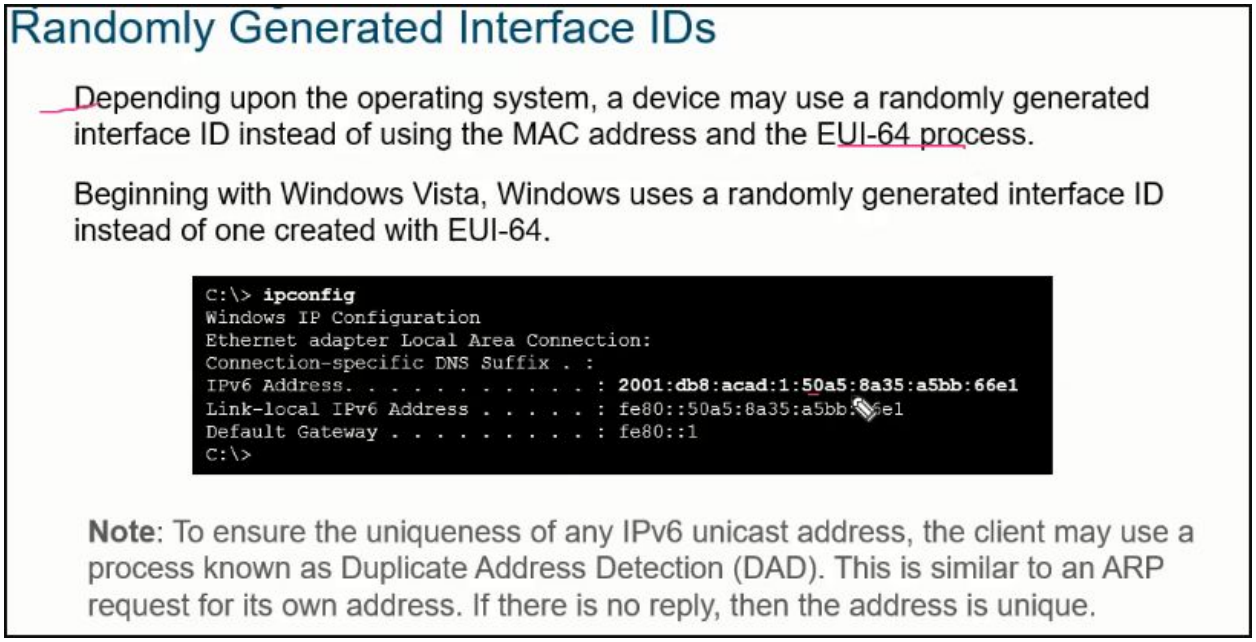
- 24-bit OUI from the client MAC address, but the 7th bit (the Universally/Locally (U/L) bit) is reversed. This means that if the 7th bit is a 0, it becomes a 1, and vice versa.
- The inserted 16-bit value fffe (in hexadecimal).
- 24-bit Device Identifier from the client MAC address.

However, this has caused privacy concerns among many users who worried that their packets could be traced to the actual physical computer. Due to these concerns, a randomly generated interface ID may be used instead.

Randomly Generated Interface IDs

Depending upon the operating system, a device may use a randomly generated interface ID instead of using the MAC address and the EUI-64 process. Beginning with Windows Vista, Windows uses a randomly generated interface ID instead of one created with EUI-64. Windows XP and previous Windows operating systems used EUI-64.

To ensure the uniqueness of any IPv6 unicast address, the client may use a process known as **Duplicate Address Detection (DAD)**. This is similar to an ARP request for its own address. If there is no reply, then the address is unique.

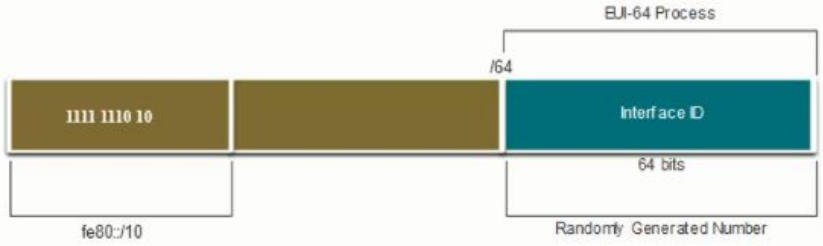


1. The correct answer is False. Router Advertisement (RA) messages are sent to all IPv6 nodes. If Method 1 (SLAAC only) is used, the RA includes network prefix, prefix-length, and default-gateway information.
2. SLAAC is a method where devices create their own GUA without the services of DHCPv6. Using SLAAC, devices rely on the local router ICMPv6 RA messages to obtain the necessary information.
3. Stateful DHCPv6 is a method where devices automatically receive their addressing information including a GUA, prefix length, and the addresses of DNS servers from a stateful DHCPv6 server.
4. SLAAC and stateless DHCPv6 is a method where devices use SLAAC for the GUA and default gateway address. The devices then use a stateless DHCPv6 server for DNS servers and other addressing information.
5. When the RA message is either SLAAC or SLAAC with stateless DHCPv6, the client must generate its own interface ID using the EUI-64 process or a randomly generated 64-bit number.

You answered 5 out of 5 questions correctly.

Dynamic LLAs

- All IPv6 interfaces must have an IPv6 LLA.
- Like IPv6 GUAs, LLAs can be configured dynamically.
- The figure shows the LLA is dynamically created using the fe80::/10 prefix and the interface ID using the EUI-64 process, or a randomly generated 64-bit number.



Dynamic LLAs on Windows

Operating systems, such as Windows, will typically use the same method for both a SLAAC-created GUA and a dynamically assigned LLA.

EUI-64 Generated Interface ID:

```
C:\> ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix . :
IPv6 Address. . . . . . . . . . . : 2001:db8:acad:1:fc99:47ff:fe75:cee0
Link-local IPv6 Address . . . . . : fe80::fc99:47ff:fe75:cee0
Default Gateway . . . . . . . . . : fe80::1
C:\>
```

Random 64-bit Generated Interface ID:

```
C:\> ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . : 2001:db8:acad:1:50a5:8a35:a5bb:66e1
   Link-local IPv6 Address . . . . . : fe80::50a5:8a35:a5bb:66e1
   Default Gateway . . . . . . . . . : fe80::1
C:\>
```

Dynamic LLAs on Cisco Routers

Cisco routers automatically create an IPv6 LLA whenever a GUA is assigned to the interface. By default, Cisco IOS routers use EUI-64 to generate the interface ID for all LLAs on IPv6 interfaces.

Here is an example of a LLA dynamically configured on the G0/0/0 interface of R1:

```
R1# show interface gigabitEthernet 0/0/0
GigabitEthernet0/0/0 is up, line protocol is up
Hardware is ISR4221-2x1GE, address is 7079.b392.3640 (bia 7079.b392.3640)
(Output omitted)
R1# show ipv6 interface brief
GigabitEthernet0/0/0 [up/up]
FE80::7279:B3FF:FE92:3640
2001:DB8:ACAD:1::1
```

By default, Cisco IOS routers use EUI-64 to generate the interface ID for all LLAs on IPv6 interfaces. For serial interfaces, the router will use the MAC address of an Ethernet interface. Recall that an LLA must be unique only on that link or network. However, a drawback to using the dynamically assigned LLA is its long interface ID, which makes it challenging to identify and remember assigned addresses.

Serial interfaces do not have Ethernet MAC addresses, so Cisco IOS uses the MAC address of the first available Ethernet interface. This is possible because link-local interfaces only have to be unique on that link.

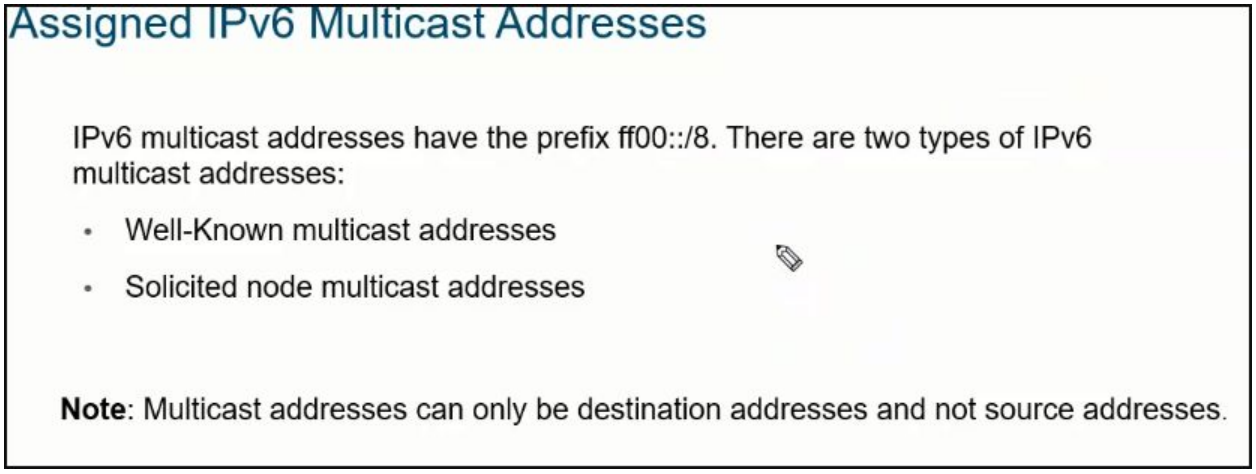
The **show ipv6 interface brief** command displays the MAC address of the Ethernet interfaces. EUI-64 uses this MAC address to generate the interface ID for the LLA. Additionally, the **show ipv6 interface brief** command displays abbreviated output for each of the interfaces. The

[up/up] output on the same line as the interface indicates the Layer 1/Layer 2 interface state. This is the same as the Status and Protocol columns in the equivalent IPv4 command.
As shown in the example, the **show ipv6 route** command can be used to verify that IPv6 networks and specific IPv6 interface addresses have been installed in the IPv6 routing table. The **show ipv6 route** command will only display IPv6 networks, not IPv4 networks.

Within the route table, a **C** next to a route indicates that this is a directly connected network. When the router interface is configured with a GUA and is in the “up/up” state, the IPv6 prefix and prefix length is added to the IPv6 routing table as a connected route.

Note: The **L** indicates a Local route, the specific IPv6 address assigned to the interface. This is not an LLA. LLAs are not included in the routing table of the router because they are not routable addresses.

The IPv6 GUA configured on the interface is also installed in the routing table as a local route. The local route has a /128 prefix. Local routes are used by the routing table to efficiently process packets with a destination address of the router interface address.



Well-Known IPv6 Multicast Addresses

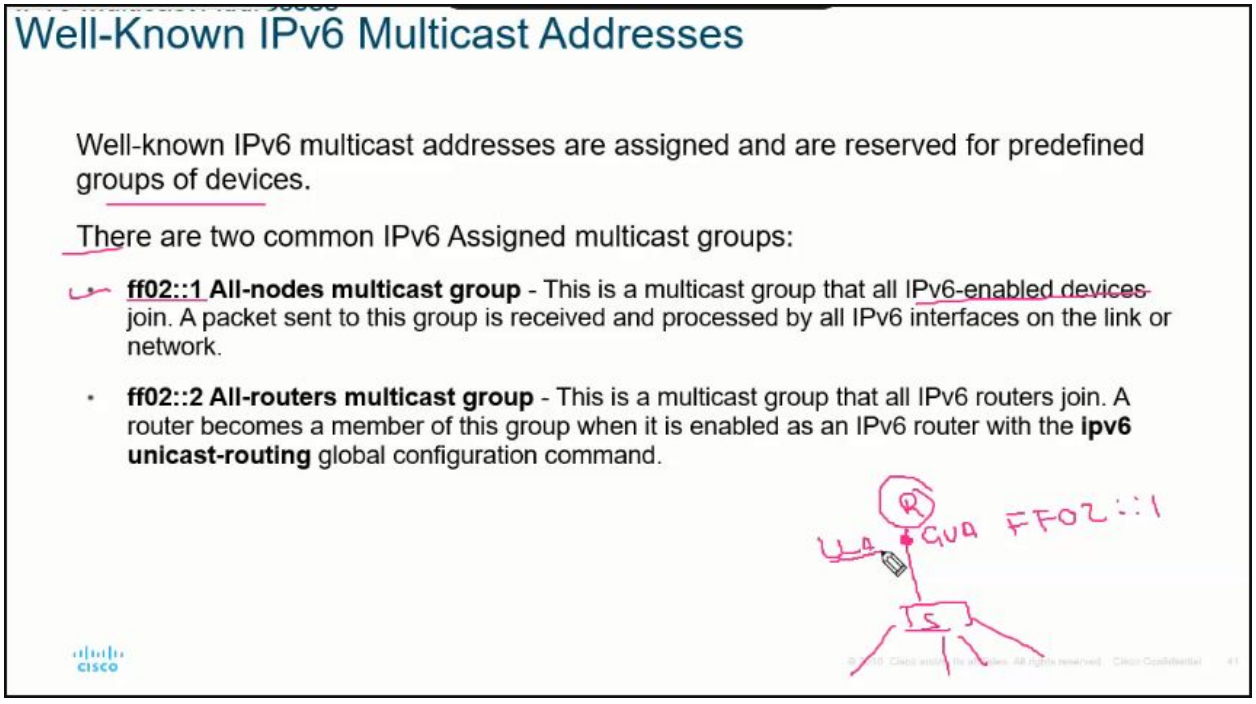
Well-known IPv6 multicast addresses are assigned. Assigned multicast addresses are reserved multicast addresses for predefined groups of devices. An assigned multicast address is a single address used to reach a group of devices running a common protocol or service. Assigned multicast addresses are used in context with specific protocols such as DHCPv6.

These are two common IPv6 assigned multicast groups:

- **ff02::1 All-nodes multicast group** - This is a multicast group that all IPv6-enabled devices join. A packet sent to this group is received and processed by all IPv6 interfaces on the link or network. This has the same effect as a broadcast address in IPv4. The

figure shows an example of communication using the all-nodes multicast address. An IPv6 router sends ICMPv6 RA messages to the all-node multicast group.

- **ff02::2 All-routers multicast group** - This is a multicast group that all IPv6 routers join. A router becomes a member of this group when it is enabled as an IPv6 router with the **ipv6 unicast-routing** global configuration command. A packet sent to this group is received and processed by all IPv6 routers on the link or network

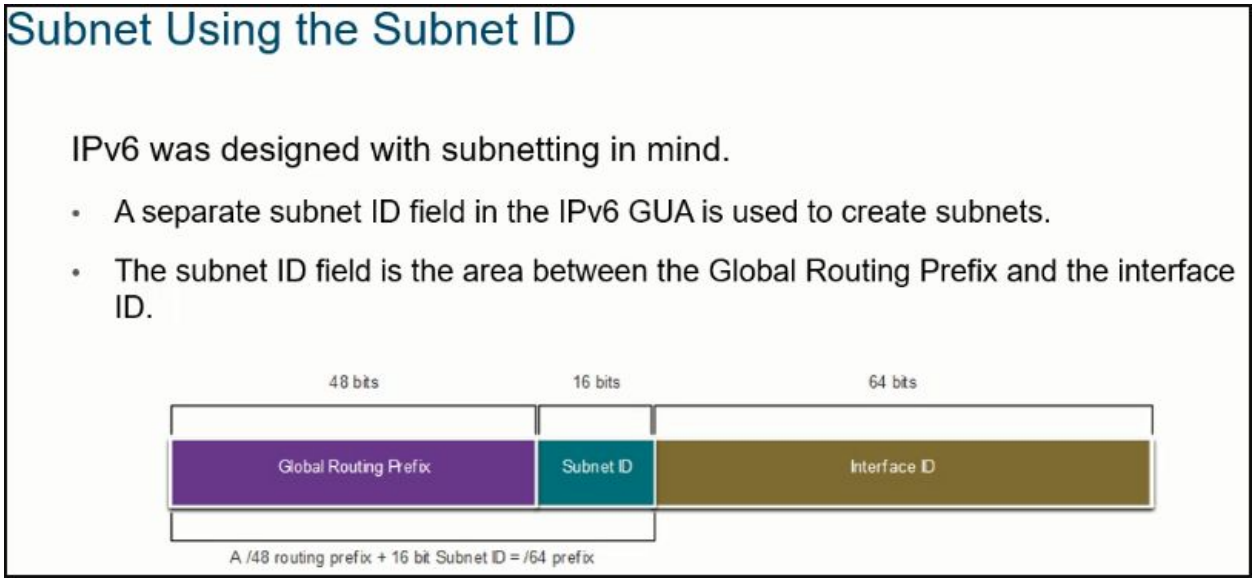


Solicited-Node IPv6 Multicast Addresses

A solicited-node multicast address is similar to the all-nodes multicast address. The advantage of a solicited-node multicast address is that it is mapped to a special Ethernet multicast address. This allows the Ethernet NIC to filter the frame by examining the destination MAC address without sending it to the IPv6 process to see if the device is the intended target of the IPv6 packet.

The graphic shows three PCs recieving a message from a router. Each PC has the following informational text: My Ethernet NIC determined this multicast is not for me. Above the graphic is indicated that the Destination MAC address is a multicast and the Destination IPv6 address is a Solicited-Node multicast.

IPv6EthernetDestination IPv6: Solicited-Node MulticastDestination MAC: MulticastMy Ethernet NIC determined this multicast is not for me.My Ethernet NIC determined this multicast is not for me.My Ethernet NIC determined this multicast is for me!



The benefit of a 128-bit address is that it can support more than enough subnets and hosts per subnet, for each network. Address conservation is not an issue. For example, if the global routing prefix is a /48, and using a typical 64 bits for the interface ID, this will create a 16-bit subnet ID:

- **16-bit subnet ID** - Creates up to 65,536 subnets.
- **64-bit interface ID** - Supports up to 18 quintillion host IPv6 addresses per subnet (i.e., 18,000,000,000,000,000,000).

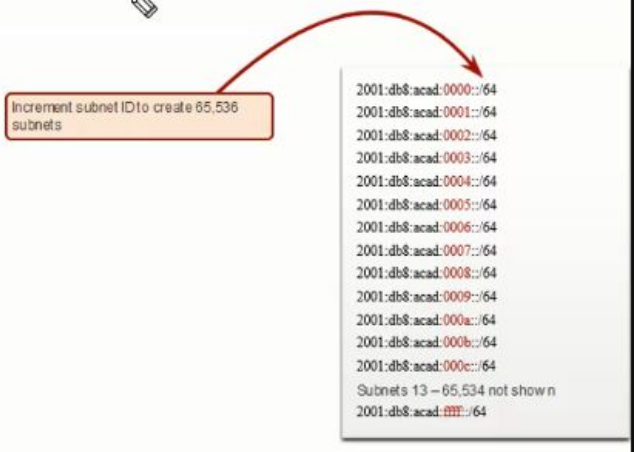
Note: Subnetting into the 64-bit interface ID (or host portion) is also possible but it is rarely required.

IPv6 subnetting is also easier to implement than IPv4, because there is no conversion to binary required. To determine the next available subnet, just count up in hexadecimal.

IPv6 Subnetting Example

Given the 2001:db8:acad::/48 global routing prefix with a 16 bit subnet ID.

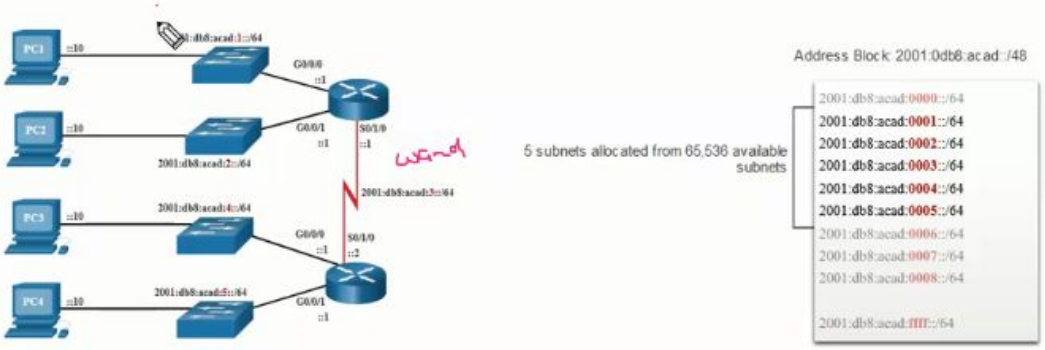
- Allows 65,536 /64 subnets
- The global routing prefix is the same for all subnets.
- Only the subnet ID hextet is incremente in hexadecimal for each subnet.



IPv6 Subnet Allocation

The example topology requires five subnets, one for each LAN as well as for the serial link between R1 and R2.

The five IPv6 subnets were allocated, with the subnet ID field 0001 through 0005. Each /64 subnet will provide more addresses than will ever be needed.



You have successfully identified the correct answers.

1. The correct answer is True. IPv6 has a separate subnet ID field in the network prefix portion of the address that can be used to create subnets.
2. The Subnet ID field, which is between the Global Routing Prefix field and the Interface ID field, is used for subnetting.
3. The subnet portion of the address is the 16 bits between the /48 and /64 prefixes, which are 2222.
4. The subnet portion of the address consists of the 32 bits between the /32 and /64 prefixes.

You answered 4 out of 4 questions correctly.

Router Configured with IPv6 Subnets

The example shows that each of the router interfaces on R1 has been configured to be on a different IPv6 subnet.

```
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# ipv6 address 2001:db8:acad:1::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# ipv6 address 2001:db8:acad:2::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface serial 0/1/0
R1(config-if)# ipv6 address 2001:db8:acad:3::1/64
R1(config-if)# no shutdown
```

What did I learn in this module?

IPv4 Issues

IPv4 has a theoretical maximum of 4.3 billion addresses. Private addresses in combination with NAT have helped to slow the depletion of IPv4 address space. With an increasing internet population, a limited IPv4 address space, issues with NAT and the IoT, the time has come to begin the transition to IPv6. Both IPv4 and IPv6 will coexist in the near future and the transition will take several years. The IETF has created various protocols and tools to help network

administrators migrate their networks to IPv6. The migration techniques can be divided into three categories: dual stack, tunneling, and translation.

IPv6 Address Representation

IPv6 addresses are 128 bits in length and written as a string of hexadecimal values. Every 4 bits is represented by a single hexadecimal digit; for a total of 32 hexadecimal values. The preferred format for writing an IPv6 address is x:x:x:x:x:x:x:x, with each “x” consisting of four hexadecimal values. For example: 2001:0db8:0000:1111:0000:0000:0000:0200. Two rules that help to reduce the number of digits needed to represent an IPv6 address. The first rule to help reduce the notation of IPv6 addresses is to omit any leading 0s (zeros) in any hextet. For example: 2001:db8:0:1111:0:0:0:200. The second rule to help reduce the notation of IPv6 addresses is that a double colon (::) can replace any single, contiguous string of one or more 16-bit hextets consisting of all zeros. For example: 2001:db8:0:1111::200.

IPv6 Address Types

There are three types of IPv6 addresses: unicast, multicast, and anycast. IPv6 does not use the dotted-decimal subnet mask notation. Like IPv4, the prefix length is represented in slash notation and is used to indicate the network portion of an IPv6 address. An IPv6 unicast address uniquely identifies an interface on an IPv6-enabled device. IPv6 addresses typically have two unicast addresses: GUA and LLA. IPv6 unique local addresses have the following uses: they are used for local addressing within a site or between a limited number of sites, they can be used for devices that will never need to access another network, and they are not globally routed or translated to a global IPv6 address. IPv6 global unicast addresses (GUAs) are globally unique and routable on the IPv6 internet. These addresses are equivalent to public IPv4 addresses. A GUA has three parts: a global routing prefix, a subnet ID, and an interface ID. An IPv6 link-local address (LLA) enables a device to communicate with other IPv6-enabled devices on the same link and only on that link (subnet). Devices can obtain an LLA either statically or dynamically.

GUA and LLA Static Configuration

The Cisco IOS command to configure an IPv4 address on an interface is **ip address** *ip-address subnet-mask*. In contrast, the command to configure an IPv6 GUA on an interface is **ipv6 address** *ipv6-address/prefix-length*. Just as with IPv4, configuring static addresses on clients does not scale to larger environments. For this reason, most network administrators in an IPv6 network will enable dynamic assignment of IPv6 addresses. Configuring the LLA manually lets you create an address that is recognizable and easier to remember. Typically, it is only necessary to create recognizable LLAs on routers. LLAs can be configured manually using the **ipv6 address** *ipv6-link-local-address* **link-local** command.

Dynamic Addressing for IPv6 GUAs

A device obtains a GUA dynamically through ICMPv6 messages. IPv6 routers periodically send out ICMPv6 RA messages, every 200 seconds, to all IPv6-enabled devices on the network. An

RA message will also be sent in response to a host sending an ICMPv6 RS message, which is a request for an RA message. The ICMPv6 RA message includes: network prefix and prefix length, default gateway address, and the DNS addresses and domain name. RA messages have three methods: SLAAC, SLAAC with a stateless DHCPv6 server, and stateful DHCPv6 (no SLAAC). With SLAAC, the client device uses the information in the RA message to create its own GUA because the message contains the prefix and the interface ID. With SLAAC with stateless DHCPv6 the RA message suggests devices use SLAAC to create their own IPv6 GUA, use the router LLA as the default gateway address, and use a stateless DHCPv6 server to obtain other necessary information. With stateful DHCPv6 the RA suggests that devices use the router LLA as the default gateway address, and the stateful DHCPv6 server to obtain a GUA, a DNS server address, domain name and all other necessary information. The interface ID can be created using the EUI-64 process or a randomly generated 64-bit number. The EUIs process uses the 48-bit Ethernet MAC address of the client and inserts another 16 bits in the middle of MAC address to create a 64-bit interface ID. Depending upon the operating system, a device may use a randomly generated interface ID.

Dynamic Addressing for IPv6 LLAs

All IPv6 devices must have an IPv6 LLA. An LLA can be configured manually or created dynamically. Operating systems, such as Windows, will typically use the same method for both a SLAAC-created GUA and a dynamically assigned LLA. Cisco routers automatically create an IPv6 LLA whenever a GUA is assigned to the interface. By default, Cisco IOS routers use EUI-64 to generate the Interface ID for all LLAs on IPv6 interfaces. For serial interfaces, the router will use the MAC address of an Ethernet interface. To make it easier to recognize and remember these addresses on routers, it is common to statically configure IPv6 LLAs on routers. To verify IPv6 address configuration use the following three commands: **show ipv6 interface brief**, **show ipv6 route**, and **ping**.

IPv6 Multicast Addresses

There are two types of IPv6 multicast addresses: well-known multicast addresses and solicited node multicast addresses. Assigned multicast addresses are reserved multicast addresses for predefined groups of devices. Well-known multicast addresses are assigned. Two commonIPv6 assigned multicast groups are: ff02::1 All-nodes multicast group and ff02::2 All-routers multicast group. A solicited-node multicast address is similar to the all-nodes multicast address. The advantage of a solicited-node multicast address is that it is mapped to a special Ethernet multicast address.

Subnet an IPv6 Network

IPv6 was designed with subnetting in mind. A separate subnet ID field in the IPv6 GUA is used to create subnets. The subnet ID field is the area between the Global Routing Prefix and the interface ID. The benefit of a 128-bit address is that it can support more than enough subnets and hosts per subnet for each network. Address conservation is not an issue. For example, if

the global routing prefix is a /48, and using a typical 64 bits for the interface ID, this will create a 16-bit subnet ID:

- 16-bit subnet ID - Creates up to 65,536 subnets.
- 64-bit interface ID - Supports up to 18 quintillion host IPv6 addresses per subnet (i.e., 18,000,000,000,000,000,000).

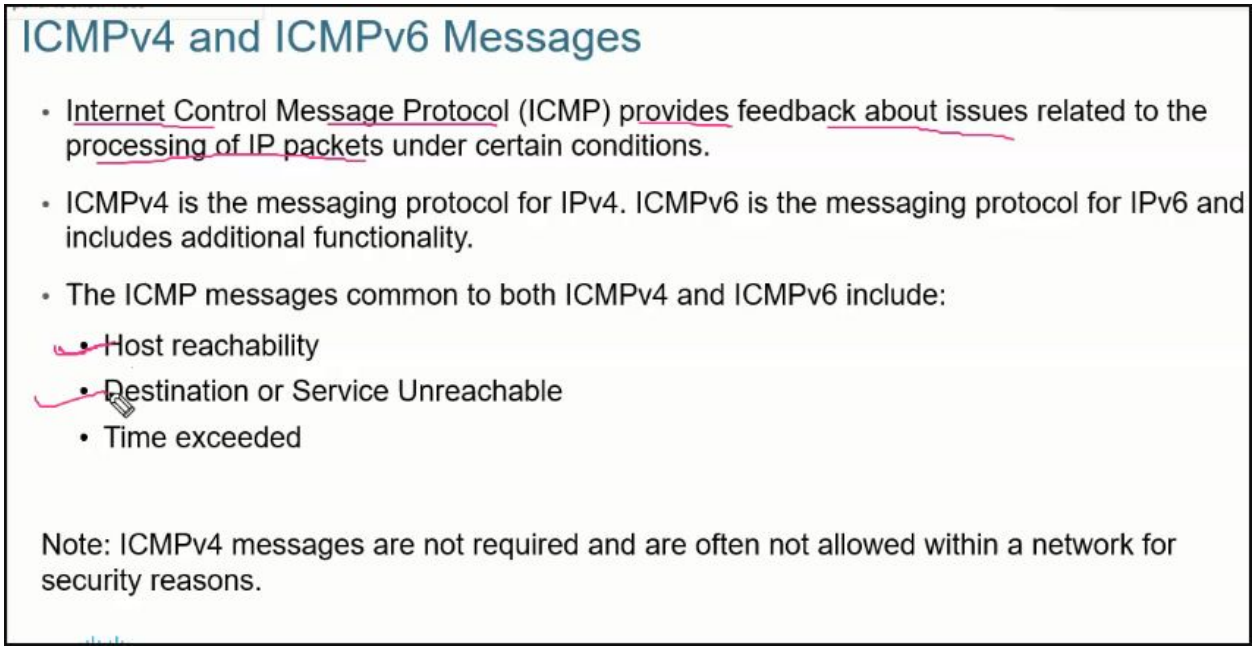
With over 65,536 subnets to choose from, the task of the network administrator becomes one of designing a logical scheme to address the network. Address conservation is not a concern when using IPv6. Similar to configuring IPv4, each router interface can be configured to be on a different IPv6 subnet.

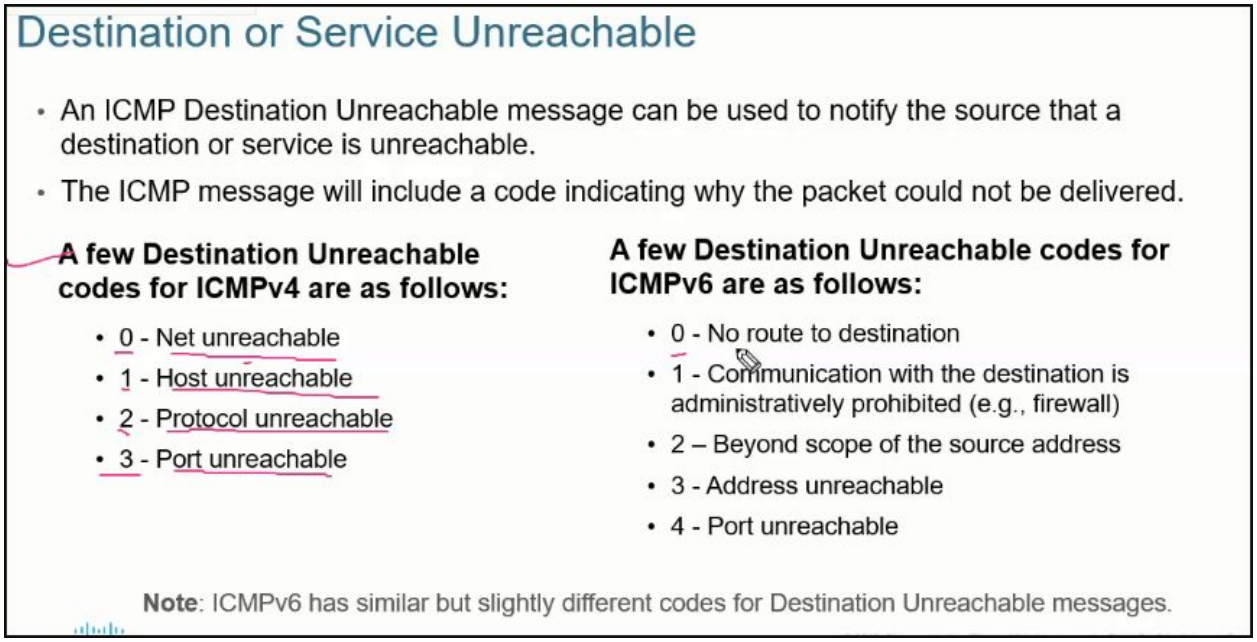
CHAPTER 13

Module Title: ICMP

Module Objective: Use various tools to test network connectivity.

Topic Title	Topic Objective
ICMP Messages	Explain how ICMP is used to test network connectivity.
Ping and Traceroute Testing	Use ping and traceroute utilities to test network connectivity.





Destination or Service Unreachable

When a host or gateway receives a packet that it cannot deliver, it can use an ICMP Destination Unreachable message to notify the source that the destination or service is unreachable. The message will include a code that indicates why the packet could not be delivered.

Some of the Destination Unreachable codes for ICMPv4 are as follows:

- 0 - Net unreachable
- 1 - Host unreachable
- 2 - Protocol unreachable
- 3 - Port unreachable

Some of the Destination Unreachable codes for ICMPv6 are as follows:

- 0 - No route to destination
- 1 - Communication with the destination is administratively prohibited (e.g., firewall)
- 2 – Beyond scope of the source address
- 3 - Address unreachable
- 4 - Port unreachable

Note: ICMPv6 has similar but slightly different codes for Destination Unreachable messages.

Time Exceeded

An ICMPv4 Time Exceeded message is used by a router to indicate that a packet cannot be forwarded because the Time to Live (TTL) field of the packet was decremented to 0. If a router receives a packet and decrements the TTL field in the IPv4 packet to zero, it discards the packet and sends a Time Exceeded message to the source host.

ICMPv6 also sends a Time Exceeded message if the router cannot forward an IPv6 packet because the packet has expired. Instead of the IPv4 TTL field, ICMPv6 uses the IPv6 Hop Limit field to determine if the packet has expired.

Note: Time Exceeded messages are used by the **traceroute** tool.

ICMPv6 Messages

ICMPv6 has new features and improved functionality not found in ICMPv4, including four new protocols as part of the Neighbor Discovery Protocol (ND or NDP).

Messaging between an IPv6 router and an IPv6 device, including dynamic address allocation are as follows:

- Router Solicitation (RS) message
- Router Advertisement (RA) message

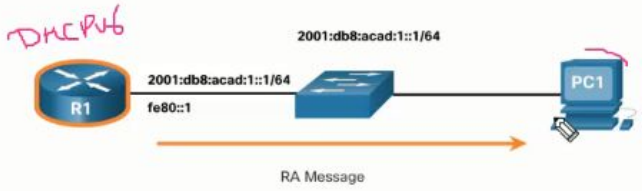
Messaging between IPv6 devices, including duplicate address detection and address resolution are as follows:

- Neighbor Solicitation (NS) message
- Neighbor Advertisement (NA) message

Note: ICMPv6 ND also includes the redirect message, which has a similar function to the redirect message used in ICMPv4.

ICMPv6 Messages (Cont.)

- RA messages are sent by IPv6-enabled routers every 200 seconds to provide addressing information to IPv6-enabled hosts.
- RA message can include addressing information for the host such as the prefix, prefix length, DNS address, and domain name.
- A host using Stateless Address Autoconfiguration (SLAAC) will set its default gateway to the link-local address of the router that sent the RA.



ICMPv6 Messages

The informational and error messages found in ICMPv6 are very similar to the control and error messages implemented by ICMPv4. However, ICMPv6 has new features and improved functionality not found in ICMPv4. ICMPv6 messages are encapsulated in IPv6.

ICMPv6 includes four new protocols as part of the Neighbor Discovery Protocol (ND or NDP).

Messaging between an IPv6 router and an IPv6 device, including dynamic address allocation are as follows:

- Router Solicitation (RS) message
- Router Advertisement (RA) message

Messaging between IPv6 devices, including duplicate address detection and address resolution are as follows:

- Neighbor Solicitation (NS) message
- Neighbor Advertisement (NA) message

Note: ICMPv6 ND also includes the redirect message, which has a similar function to the redirect message used in ICMPv4.

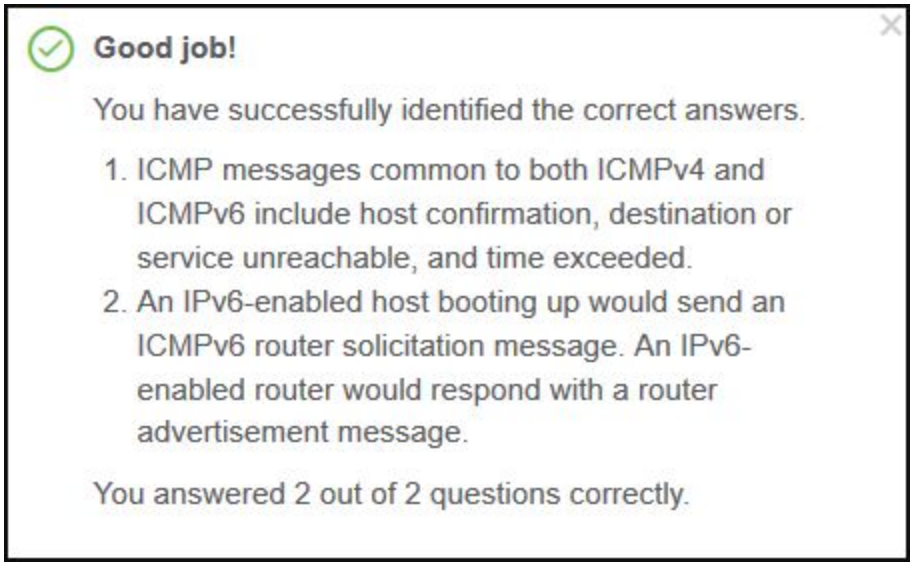
RA messages are sent by IPv6-enabled routers every 200 seconds to provide addressing information to IPv6-enabled hosts. The RA message can include addressing information for the host such as the prefix, prefix length, DNS address, and domain name. A host using Stateless Address Autoconfiguration (SLAAC) will set its default gateway to the link-local address of the router that sent the RA.
An IPv6-enabled router will also send out an RA message in response to an RS message

When a device is assigned a global IPv6 unicast or link-local unicast address, it may perform duplicate address detection (DAD) to ensure that the IPv6 address is unique. To check the uniqueness of an address, the device will send an NS message with its own IPv6 address as the targeted IPv6 address, as shown in the figure.

If another device on the network has this address, it will respond with an NA message. This NA message will notify the sending device that the address is in use. If a corresponding NA message is not returned within a certain amount of time, the unicast address is unique and acceptable for use.

Note: DAD is not required, but RFC 4861 recommends that DAD is performed on unicast addresses.

Address resolution is used when a device on the LAN knows the IPv6 unicast address of a destination but does not know its Ethernet MAC address. To determine the MAC address for the destination, the device will send an NS message to the solicited node address. The message will include the known (targeted) IPv6 address. The device that has the targeted IPv6 address will respond with an NA message containing its Ethernet MAC address.



Ping - Test Connectivity

In the previous topic, you were introduced to the **ping** and traceroute (**tracert**) tools. In this topic, you will learn about the situations in which each tool is used, and how to use them. Ping is an IPv4 and IPv6 testing utility that uses ICMP echo request and echo reply messages to test connectivity between hosts.

To test connectivity to another host on a network, an echo request is sent to the host address using the **ping** command. If the host at the specified address receives the echo request, it responds with an echo reply. As each echo reply is received, **ping** provides feedback on the time between when the request was sent and when the reply was received. This can be a measure of network performance.

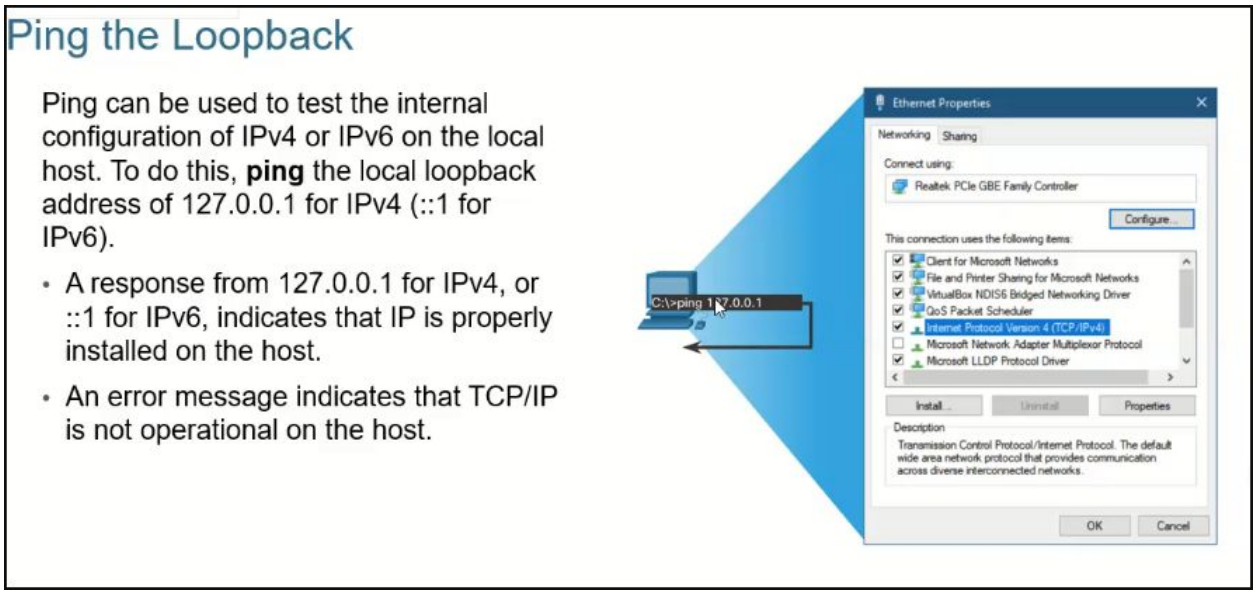
Ping has a timeout value for the reply. If a reply is not received within the timeout, ping provides a message indicating that a response was not received. This may indicate that there is a problem, but could also indicate that security features blocking ping messages have been enabled on the network. It is common for the first ping to timeout if address resolution (ARP or ND) needs to be performed before sending the ICMP Echo Request.

After all the requests are sent, the **ping** utility provides a summary that includes the success rate and average round-trip time to the destination.

Type of connectivity tests performed with **ping** include the following:

- Pinging the local loopback
- Pinging the default gateway
- Pinging the remote host

A response from 127.0.0.1 for IPv4, or ::1 for IPv6, indicates that IP is properly installed on the host



Ping the Default Gateway

You can also use **ping** to test the ability of a host to communicate on the local network. This is generally done by pinging the IP address of the default gateway of the host. A successful **ping** to the default gateway indicates that the host and the router interface serving as the default gateway are both operational on the local network.

For this test, the default gateway address is most often used because the router is normally always operational. If the default gateway address does not respond, a **ping** can be sent to the IP address of another host on the local network that is known to be operational.

If either the default gateway or another host responds, then the local host can successfully communicate over the local network. If the default gateway does not respond but another host does, this could indicate a problem with the router interface serving as the default gateway.

One possibility is that the wrong default gateway address has been configured on the host. Another possibility is that the router interface may be fully operational but have security applied to it that prevents it from processing or responding to ping requests.

Ping a Remote Host

Ping can also be used to test the ability of a local host to communicate across an internetwork. The local host can ping an operational IPv4 host of a remote network, as shown in the figure. The router uses its IP routing table to forward the packets.

If this ping is successful, the operation of a large piece of the internetwork can be verified. A successful **ping** across the internetwork confirms communication on the local network, the operation of the router serving as the default gateway, and the operation of all other routers that might be in the path between the local network and the network of the remote host.

Additionally, the functionality of the remote host can be verified. If the remote host could not communicate outside of its local network, it would not have responded.

Note: Many network administrators limit or prohibit the entry of ICMP messages into the corporate network; therefore, the lack of a **ping** response could be due to security restrictions.

Traceroute - Test the Path

Ping is used to test connectivity between two hosts but does not provide information about the details of devices between the hosts. Traceroute (**tracert**) is a utility that generates a list of hops that were successfully reached along the path. This list can provide important verification and troubleshooting information. If the data reaches the destination, then the trace lists the interface of every router in the path between the hosts. If the data fails at some hop along the way, the address of the last router that responded to the trace can provide an indication of where the problem or security restrictions are found.

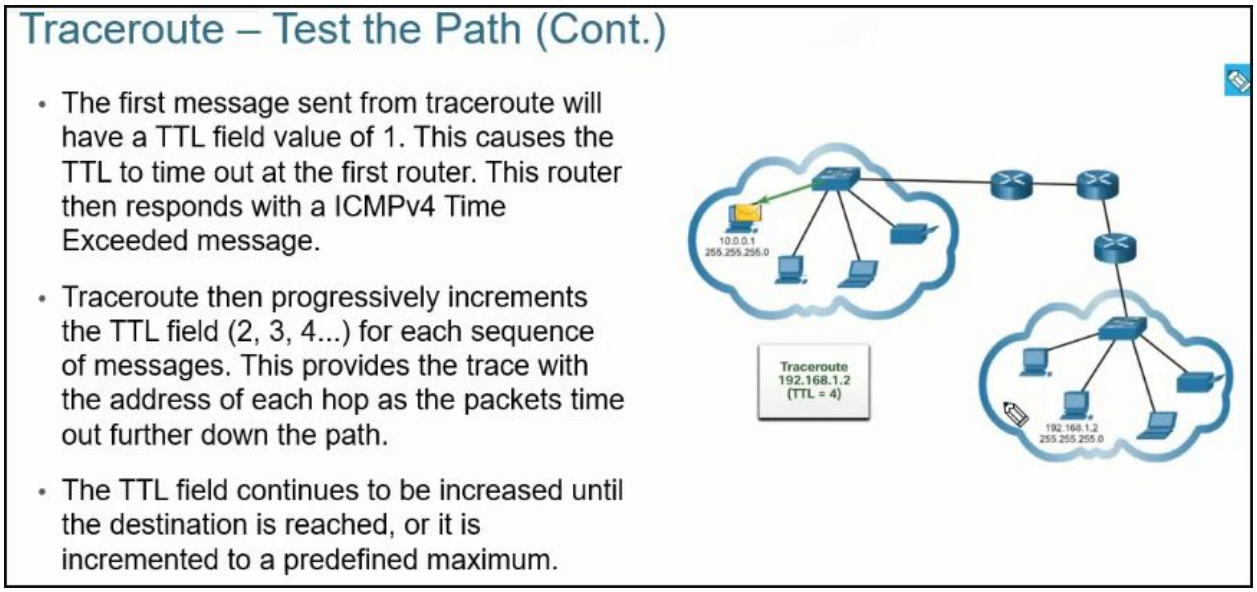
Round Trip Time (RTT)

Using traceroute provides round-trip time for each hop along the path and indicates if a hop fails to respond. The round-trip time is the time a packet takes to reach the remote host and for the response from the host to return. An asterisk (*) is used to indicate a lost or unreplied packet.

This information can be used to locate a problematic router in the path or may indicate that the router is configured not to reply. If the display shows high response times or data losses from a particular hop, this is an indication that the resources of the router or its connections may be stressed.

IPv4 TTL and IPv6 Hop Limit

Traceroute makes use of a function of the TTL field in IPv4 and the Hop Limit field in IPv6 in the Layer 3 headers, along with the ICMP Time Exceeded message.



What did I learn in this module?

ICMP Messages

The TCP/IP suite provides for error messages and informational messages when communicating with another IP device. These messages are sent using ICMP. The purpose of these messages is to provide feedback about issues related to the processing of IP packets under certain conditions. The ICMP messages common to both ICMPv4 and ICMPv6 are: Host reachability, Destination or Service Unreachable, and Time exceeded. An ICMP Echo Message tests the reachability of a host on an IP network. The local host sends an ICMP Echo Request to a host. If the host is available, the destination host responds with an Echo Reply. This is the basis of the **ping** utility. When a host or gateway receives a packet that it cannot deliver, it can use an ICMP Destination Unreachable message to notify the source. The message will include a code that indicates why the packet could not be delivered. An ICMPv4 Time Exceeded message is used by a router to indicate that a packet cannot be forwarded because the Time to Live (TTL) field of the packet was decremented to zero. If a router receives a packet and decrements the TTL field to zero, it discards the packet and sends a Time Exceeded message to the source host. ICMPv6 also sends a Time Exceeded in this situation. ICMPv6 uses the IPv6 hop limit field to determine if the packet has expired. Time Exceeded messages are used by the **traceroute** tool. The messages between an IPv6 router and an IPv6 device including dynamic address allocation include RS and RA. The messages between IPv6 devices include the redirect (similar to IPv4), NS and NA.

Ping and Traceroute Testing

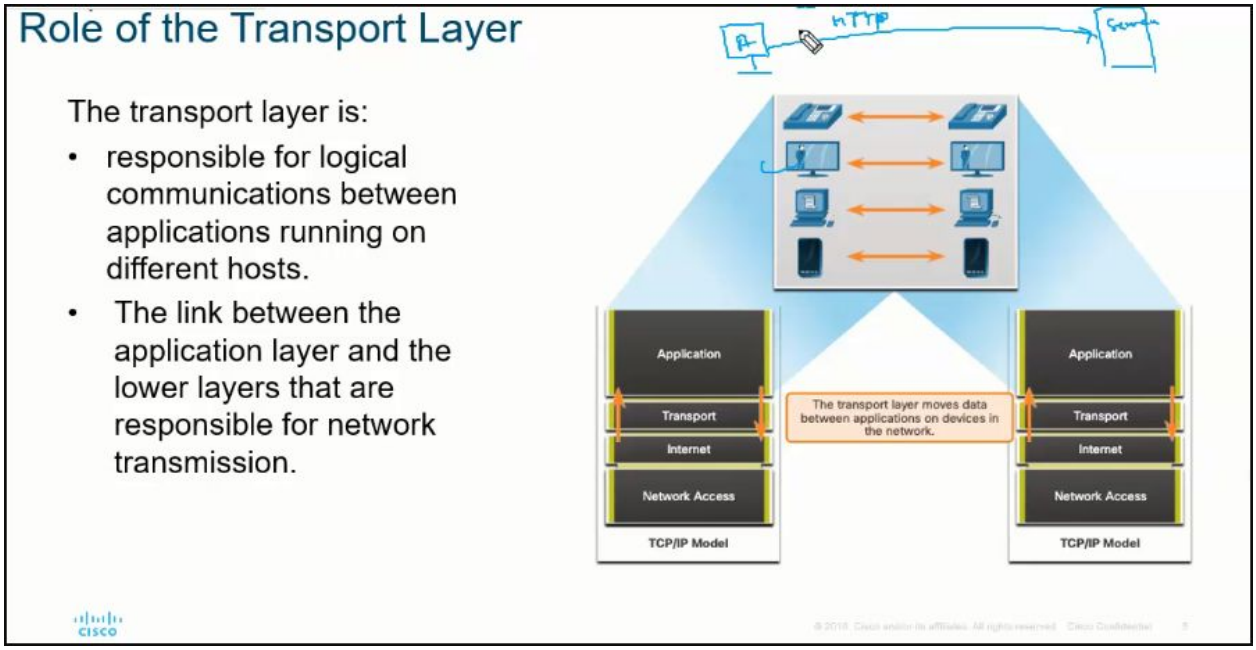
Ping (used by IPv4 and IPv6) uses ICMP echo request and echo reply messages to test connectivity between hosts. To test connectivity to another host on a network, an echo request is sent to the host address using the ping command. If the host at the specified address

receives the echo request, it responds with an echo reply. As each echo reply is received, ping provides feedback on the time between when the request was sent and when the reply was received. After all the requests are sent, the ping utility provides a summary that includes the success rate and average round-trip time to the destination. Ping can be used to test the internal configuration of IPv4 or IPv6 on the local host. Ping the local loopback address of 127.0.0.1 for IPv4 (::1 for IPv6). Use **ping** to test the ability of a host to communicate on the local network, by pinging the IP address of the default gateway of the host. A successful ping to the default gateway indicates that the host and the router interface serving as the default gateway are both operational on the local network. Ping can also be used to test the ability of a local host to communicate across an internetwork. The local host can **ping** an operational IPv4 host of a remote network. Traceroute (tracert) generates a list of hops that were successfully reached along the path. This list provides verification and troubleshooting information. If the data reaches the destination, then the trace lists the interface of every router in the path between the hosts. If the data fails at some hop along the way, the address of the last router that responded to the trace can provide an indication of where the problem or security restrictions are found. The round-trip time is the time a packet takes to reach the remote host and for the response from the host to return. Traceroute makes use of a function of the TTL field in IPv4 and the Hop Limit field in IPv6 in the Layer 3 headers, along with the ICMP time exceeded message.

When the **ping** command is issued on a router, the most common indicators are as follows:

- **!** - indicates receipt of an ICMP echo reply message
- **.** - indicates a time expired while waiting for an ICMP echo reply message
- **U** - an ICMP message of unreachability was received

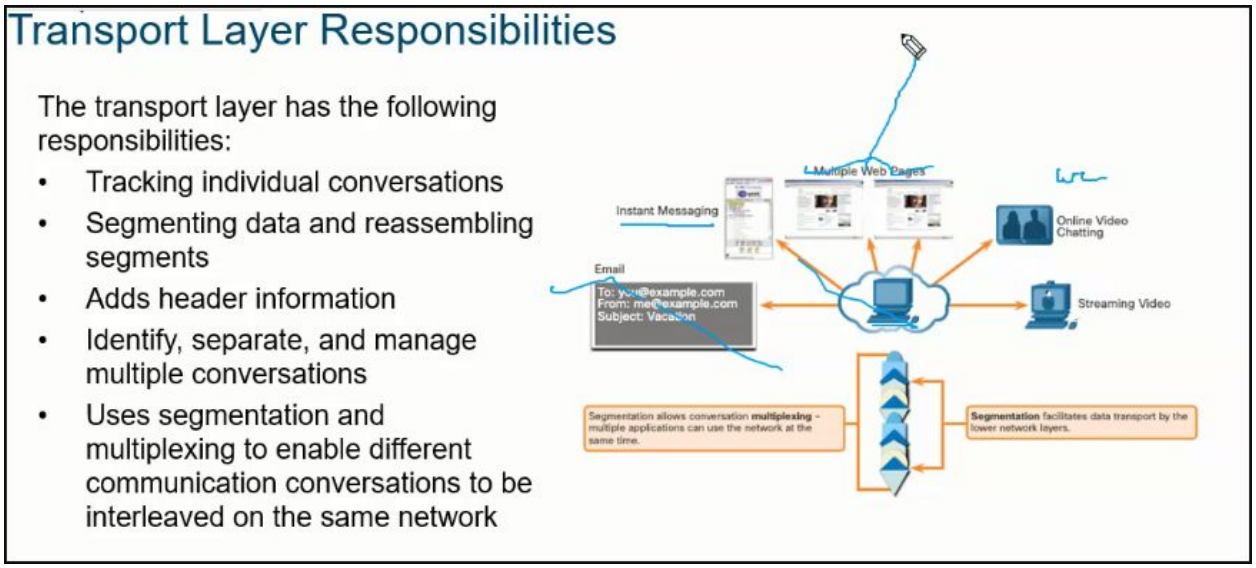
CHAPTER 14



TCP provides reliability and flow control using these basic operations:

- Number and track data segments transmitted to a specific host from a specific application
- Acknowledge received data
- Retransmit any unacknowledged data after a certain amount of time
- Sequence data that might arrive in wrong order
- Send data at an efficient rate that is acceptable by the receiver

In order to maintain the state of a conversation and track the information, TCP must first establish a connection between the sender and the receiver. This is why TCP is known as a connection-oriented protocol.



UDP is a connectionless protocol. Because UDP does not provide reliability or flow control, it does not require an established connection. Because UDP does not track information sent or received between the client and server, UDP is also known as a stateless protocol.

UDP is also known as a best-effort delivery protocol because there is no acknowledgment that the data is received at the destination. With UDP, there are no transport layer processes that inform the sender of a successful delivery.

UDP is like placing a regular, nonregistered, letter in the mail. The sender of the letter is not aware of the availability of the receiver to receive the letter. Nor is the post office responsible for tracking the letter or informing the sender if the letter does not arrive at the final destination.

The Right Transport Layer Protocol for the Right Application

Some applications can tolerate some data loss during transmission over the network, but delays in transmission are unacceptable. For these applications, UDP is the better choice because it requires less network overhead. UDP is preferable for applications such as Voice over IP (VoIP). Acknowledgments and retransmission would slow down delivery and make the voice conversation unacceptable.

UDP is also used by request-and-reply applications where the data is minimal, and retransmission can be done quickly. For example, domain name service (DNS) uses UDP for this type of transaction. The client requests IPv4 and IPv6 addresses for a known domain name from a DNS server. If the client does not receive a response in a predetermined amount of time, it simply sends the request again.

For example, if one or two segments of a live video stream fail to arrive, it creates a momentary disruption in the stream. This may appear as distortion in the image or sound, but may not be

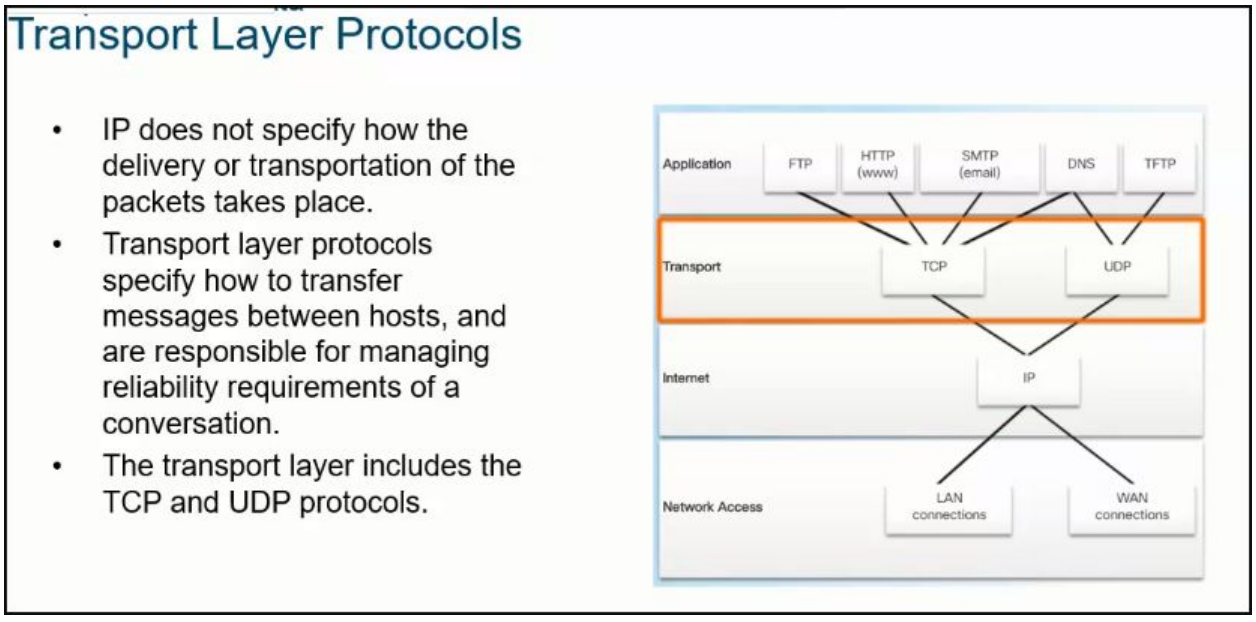
noticeable to the user. If the destination device had to account for lost data, the stream could be delayed while waiting for retransmissions, therefore causing the image or sound to be greatly degraded. In this case, it is better to render the best media possible with the segments received, and forego reliability.

For other applications it is important that all the data arrives and that it can be processed in its proper sequence. For these types of applications, TCP is used as the transport protocol. For example, applications such as databases, web browsers, and email clients, require that all data that is sent arrives at the destination in its original condition. Any missing data could corrupt a communication, making it either incomplete or unreadable. For example, it is important when accessing banking information over the web to make sure all the information is sent and received correctly.

Application developers must choose which transport protocol type is appropriate based on the requirements of the applications. Video may be sent over TCP or UDP. Applications that stream stored audio and video typically use TCP. The application uses TCP to perform buffering, bandwidth probing, and congestion control, in order to better control the user experience.

Real-time video and voice usually use UDP, but may also use TCP, or both UDP and TCP. A video conferencing application may use UDP by default, but because many firewalls block UDP, the application can also be sent over TCP.

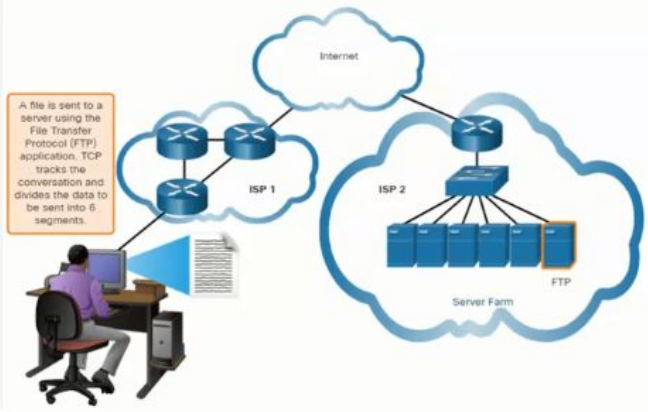
Applications that stream stored audio and video use TCP. For example, if your network suddenly cannot support the bandwidth needed to watch an on-demand movie, the application pauses the playback. During the pause, you might see a “buffering...” message while TCP works to re-establish the stream. When all the segments are in order and a minimum level of bandwidth is restored, your TCP session resumes, and the movie resumes playing.



Transmission Control Protocol

TCP provides reliability and flow control. TCP basic operations:

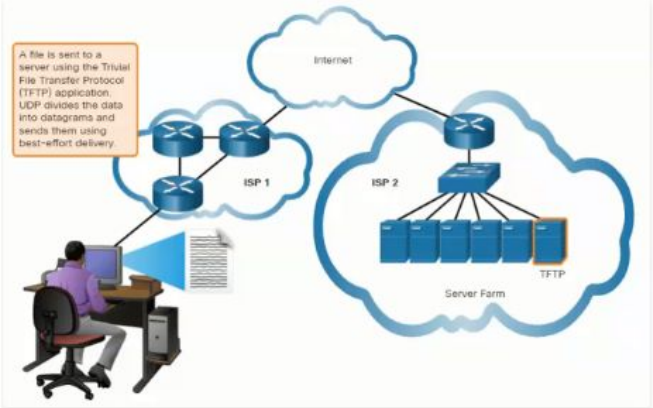
- Number and track data segments transmitted to a specific host from a specific application
- Acknowledge received data
- Retransmit any unacknowledged data after a certain amount of time
- Sequence data that might arrive in wrong order
- Send data at an efficient rate that is acceptable by the receiver



User Datagram Protocol (UDP)

UDP provides the basic functions for delivering datagrams between the appropriate applications, with very little overhead and data checking.

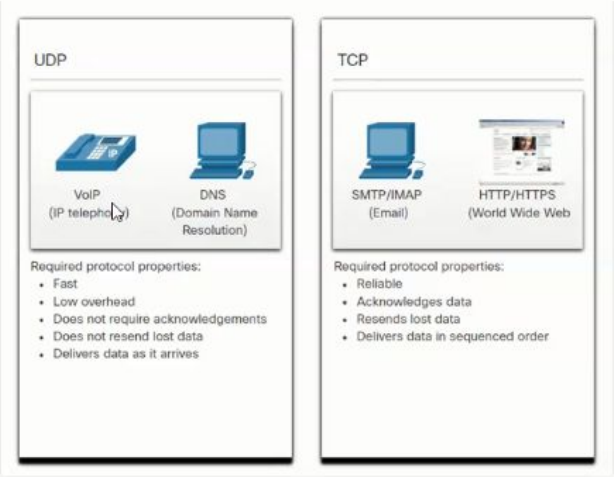
- UDP is a connectionless protocol.
- UDP is known as a best-effort delivery protocol because there is no acknowledgment that the data is received at the destination.



The Right Transport Layer Protocol for the Right Application

UDP is also used by request-and-reply applications where the data is minimal, and retransmission can be done quickly.

If it is important that all the data arrives and that it can be processed in its proper sequence, TCP is used as the transport protocol.



Good job!

You have successfully identified the correct answers.

1. The transport layer is responsible for establishing a temporary communication session between the source and destination host applications.
2. The transport layer is responsible for conversation multiplexing, segmenting data and reassembling segments, and tracking individual conversations.
3. UDP is a best-effort delivery protocol while TCP is a reliable transport protocol.
4. UDP would be used by time sensitive VoIP applications.

You answered 4 out of 4 questions correctly.

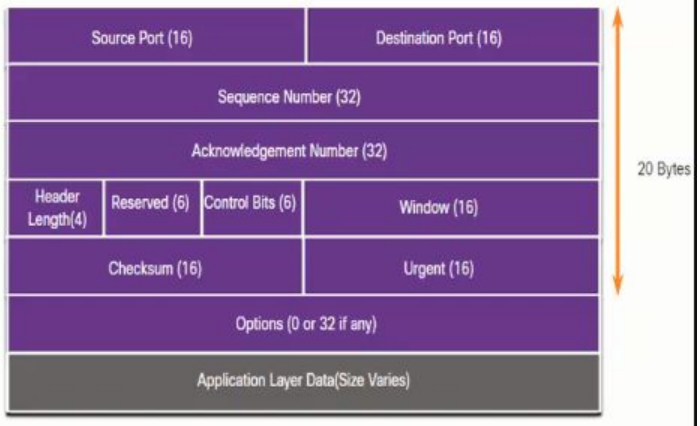
TCP Features

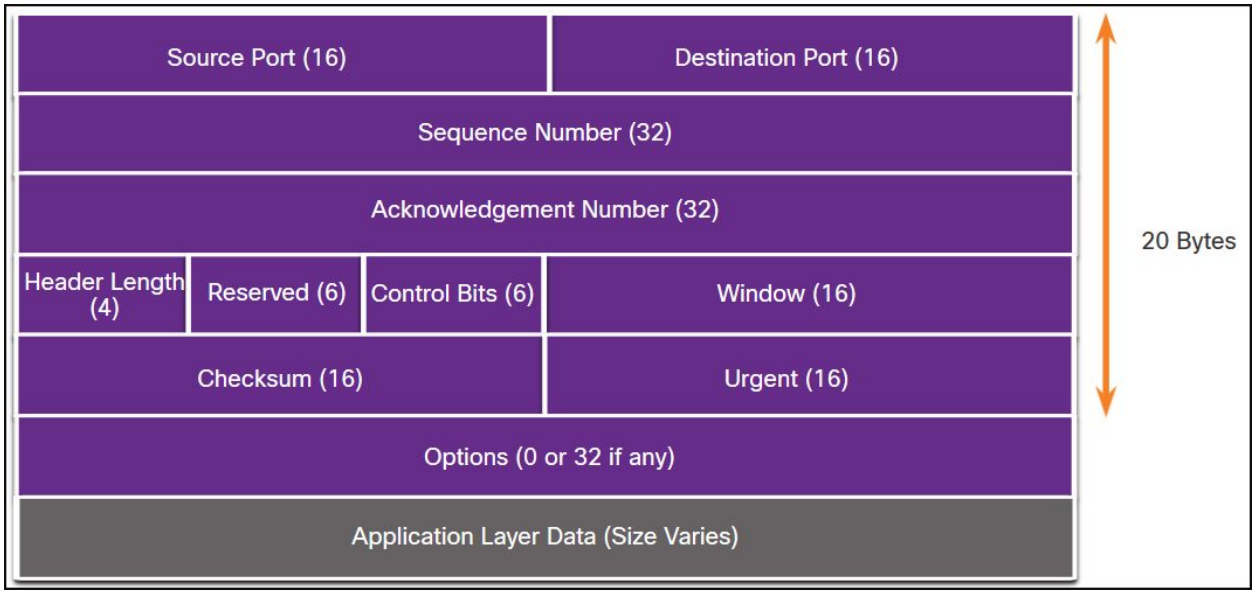
- **Establishes a Session** - TCP is a connection-oriented protocol that negotiates and establishes a permanent connection (or session) between source and destination devices prior to forwarding any traffic.
- **Ensures Reliable Delivery** - For many reasons, it is possible for a segment to become corrupted or lost completely, as it is transmitted over the network. TCP ensures that each segment that is sent by the source arrives at the destination.
- **Provides Same-Order Delivery** - Because networks may provide multiple routes that can have different transmission rates, data can arrive in the wrong order.
- **Supports Flow Control** - Network hosts have limited resources (i.e., memory and processing power). When TCP is aware that these resources are overtaxed, it can request that the sending application reduce the rate of data flow.

TCP Header

TCP is a stateful protocol which means it keeps track of the state of the communication session.

TCP records which information it has sent, and which information has been acknowledged.



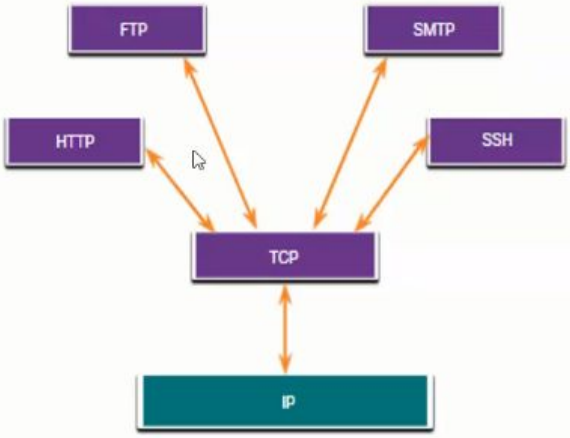


TCP Header Fields

TCP Header Field	Description
Source Port	A 16-bit field used to identify the source application by port number.
Destination Port	A 16-bit field used to identify the destination application by port number.
Sequence Number	A 32-bit field used for data reassembly purposes.
Acknowledgment Number	A 32-bit field used to indicate that data has been received and the next byte expected from the source.
Header Length	A 4-bit field known as "data offset" that indicates the length of the TCP segment header.
Reserved	A 6-bit field that is reserved for future use.
Control bits	A 6-bit field used that includes bit codes, or flags, which indicate the purpose and function of the TCP segment.
Window size	A 16-bit field used to indicate the number of bytes that can be accepted at one time.
Checksum	A 16-bit field used for error checking of the segment header and data.
Urgent	A 16-bit field used to indicate if the contained data is urgent.

Applications that use TCP

TCP handles all tasks associated with dividing the data stream into segments, providing reliability, controlling data flow, and reordering segments.



Good job!

You have successfully identified the correct answers.

1. The TCP transport layer protocol ensures reliable same-order delivery.
2. The TCP header consists of 10 fields in a 20-byte header.
3. FTP and HTTP require the use of the TCP transport layer protocol.

You answered 3 out of 3 questions correctly.

UDP Features

UDP features include the following:

- Data is reconstructed in the order that it is received.
- Any segments that are lost are not resent.
- There is no session establishment.
- The sending is not informed about resource availability.

This topic will cover UDP, what it does, and when it is a good idea to use it instead of TCP. UDP is a best-effort transport protocol. UDP is a lightweight transport protocol that offers the same data segmentation and reassembly as TCP, but without TCP reliability and flow control.

UDP is such a simple protocol that it is usually described in terms of what it does not do compared to TCP.

UDP features include the following:

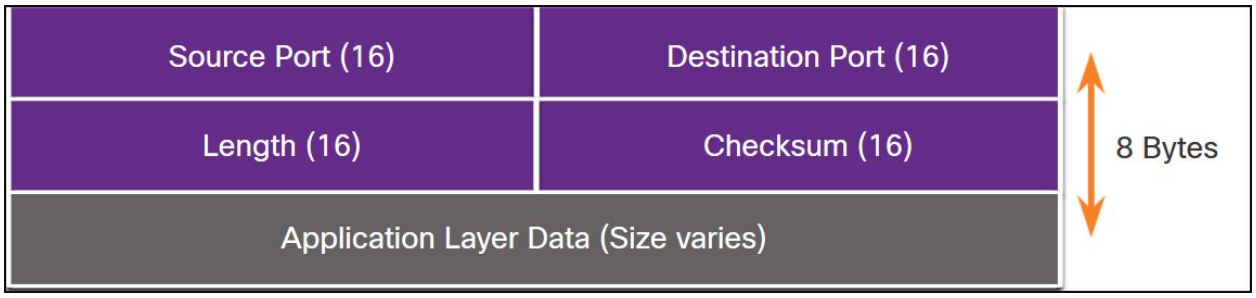
- Data is reconstructed in the order that it is received.
- Any segments that are lost are not resent.
- There is no session establishment.
- The sending is not informed about resource availability.

UDP is a stateless protocol, meaning neither the client, nor the server, tracks the state of the communication session. If reliability is required when using UDP as the transport protocol, it must be handled by the application.

One of the most important requirements for delivering live video and voice over the network is that the data continues to flow quickly. Live video and voice applications can tolerate some data loss with minimal or no noticeable effect, and are perfectly suited to UDP.

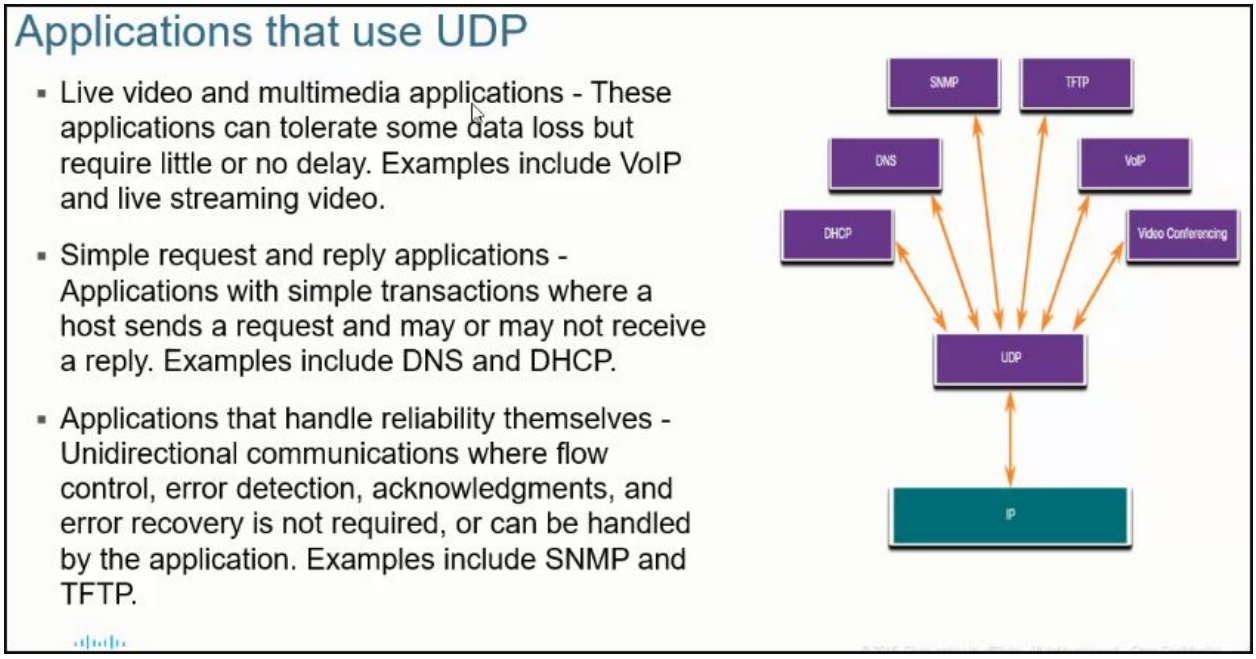
The blocks of communication in UDP are called datagrams, or segments. These datagrams are sent as best effort by the transport layer protocol.

The UDP header is far simpler than the TCP header because it only has four fields and requires 8 bytes (i.e., 64 bits).

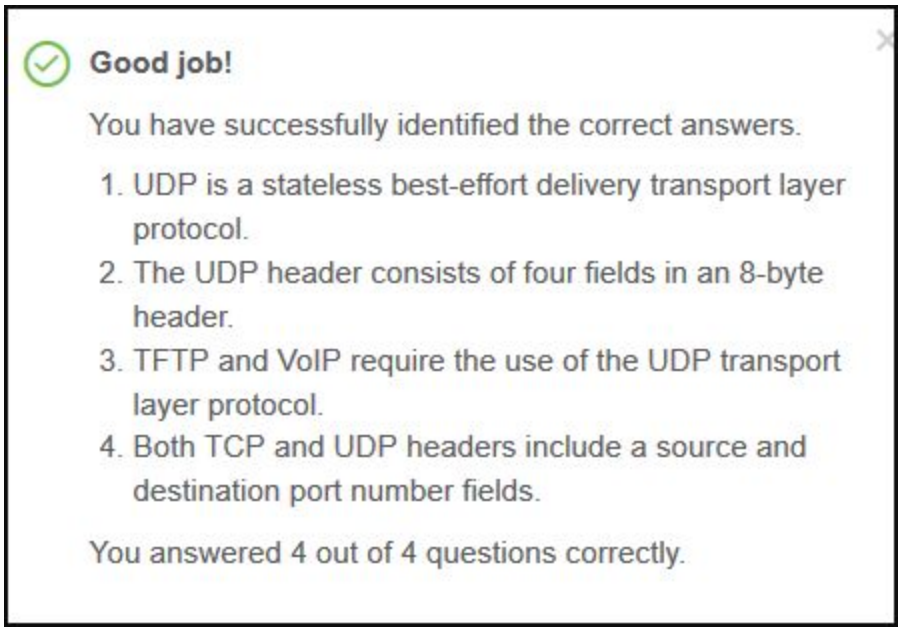


There are three types of applications that are best suited for UDP:

- **Live video and multimedia applications** - These applications can tolerate some data loss, but require little or no delay. Examples include VoIP and live streaming video.
- **Simple request and reply applications** - Applications with simple transactions where a host sends a request and may or may not receive a reply. Examples include DNS and DHCP.
- **Applications that handle reliability themselves** - Unidirectional communications where flow control, error detection, acknowledgments, and error recovery is not required, or can be handled by the application. Examples include SNMP and TFTP.



Although DNS and SNMP use UDP by default, both can also use TCP. DNS will use TCP if the DNS request or DNS response is more than 512 bytes, such as when a DNS response includes many name resolutions. Similarly, under some situations the network administrator may want to configure SNMP to use TCP.



Socket Pairs

The source and destination ports are placed within the segment. The segments are then encapsulated within an IP packet. The IP packet contains the IP address of the source and

destination. The combination of the source IP address and source port number, or the destination IP address and destination port number is known as a socket.

Port Number Groups

The Internet Assigned Numbers Authority (IANA) is the standards organization responsible for assigning various addressing standards, including the 16-bit port numbers. The 16 bits used to identify the source and destination port numbers provides a range of ports from 0 through 65535.

Port Group	Number Range	Description
Well-known Ports	**0 to 1,023**	● These port numbers are reserved for common or popular services and applications such as web browsers, email clients, and remote access clients. ● Defined well-known ports for common server applications enables clients to easily identify the associated service required.
Registered Ports	**1,024 to 49,151**	● These port numbers are assigned by IANA to a requesting entity to use with specific processes or applications. ● These processes are primarily individual applications that a user has chosen to install, rather than common applications that would receive a well-known port number. ● For example, Cisco has registered port 1812 for its RADIUS server authentication process.
Private and/or **Dynamic Ports**	**49,152 to 65,535**	● These ports are also known as *ephemeral ports*. ● The client's OS usually assign port numbers dynamically when a connection to a service is initiated. ● The dynamic port is then used to identify the client application during communication.

Port Number	Protocol	Application

20	TCP	File Transfer Protocol (FTP) - Data
21	TCP	File Transfer Protocol (FTP) - Control
22	TCP	Secure Shell (SSH)
23	TCP	Telnet
25	TCP	Simple Mail Transfer Protocol (SMTP)
53	UDP, TCP	Domain Name Service (DNS)
67	UDP	Dynamic Host Configuration Protocol (DHCP) - Server
68	UDP	Dynamic Host Configuration Protocol - Client
69	UDP	Trivial File Transfer Protocol (TFTP)
80	TCP	Hypertext Transfer Protocol (HTTP)
110	TCP	Post Office Protocol version 3 (POP3)
143	TCP	Internet Message Access Protocol (IMAP)
161	UDP	Simple Network Management Protocol (SNMP)
443	TCP	Hypertext Transfer Protocol Secure (HTTPS)

Some applications may use both TCP and UDP. For example, DNS uses UDP when clients send requests to a DNS server. However, communication between two DNS servers always uses TCP.

Netstat is an important network utility that can be used to verify those connections. As shown below, enter the command **netstat** to list the protocols in use, the local address and port numbers, the foreign address and port numbers, and the connection state.
C:\> **netstat**
By default, the **netstat** command will attempt to resolve IP addresses to domain names and port numbers to well-known applications. The **-n** option can be used to display IP addresses and port numbers in their numerical form

Multiple Separate Communications

TCP and UDP transport layer protocols use port numbers to manage multiple, simultaneous conversations.

The source port number is associated with the originating application on the local host whereas the destination port number is associated with the destination application on the remote host.

Good job!

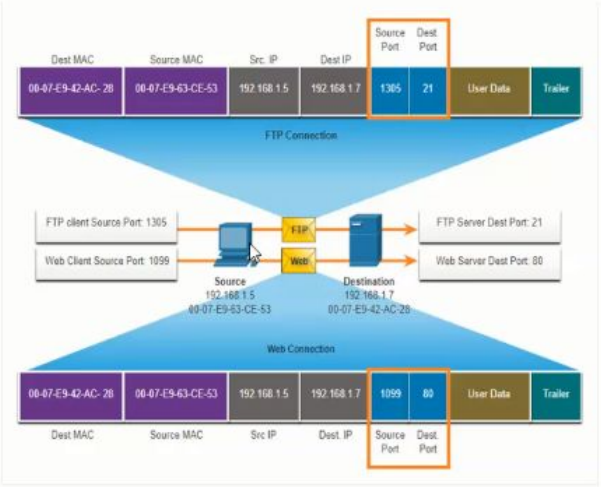
You have successfully identified the correct answers.

1. The socket pair for a host with IP address 10.1.1.10 requesting web services from a server at 10.1.1.254 would be 10.1.1.10:1099, 10.1.1.254:80.
2. FTP, HTTP, and TFTP applications port numbers are defined in the well-known port numbers group.
3. The **netstat** Windows command would display protocols in use, the local address and port numbers, the foreign address and port numbers, and the connection state.

You answered 3 out of 3 questions correctly.

Socket Pairs

- The source and destination ports are placed within the segment.
- The segments are then encapsulated within an IP packet.
- The combination of the source IP address and source port number, or the destination IP address and destination port number is known as a socket.
- Sockets enable multiple processes, running on a client, to distinguish themselves from each other, and multiple connections to a server process to be distinguished from each other.



Port Number Groups

Port Group	Number Range	Description
Well-known Ports	**0 to 1,023**	•These port numbers are reserved for common or popular services and applications such as web browsers, email clients, and remote access clients. •Defined well-known ports for common server applications enables clients to easily identify the associated service required.
Registered Ports	**1,024 to 49,151**	•These port numbers are assigned by IANA to a requesting entity to use with specific processes or applications. •These processes are primarily individual applications that a user has chosen to install, rather than common applications that would receive a well-known port number. •For example, Cisco has registered port 1812 for its RADIUS server authentication process.
Private and/or **Dynamic Ports**	**49,152 to 65,535**	•These ports are also known as *ephemeral ports*. •The client's OS usually assign port numbers dynamically when a connection to a service is initiated. •The dynamic port is then used to identify the client application during communication.

Port Number Groups (Cont.)

Well-Known Port Numbers

Port Number	Protocol	Application
20	TCP	File Transfer Protocol (FTP) - Data
21	TCP	File Transfer Protocol (FTP) - Control
22	TCP	Secure Shell (SSH)
23	TCP	Telnet
25	TCP	Simple Mail Transfer Protocol (SMTP)
53	UDP, TCP	Domain Name Service (DNS)
67	UDP	Dynamic Host Configuration Protocol (DHCP) - Server
68	UDP	Dynamic Host Configuration Protocol - Client
69	UDP	Trivial File Transfer Protocol (TFTP)
80	TCP	Hypertext Transfer Protocol (HTTP)
110	TCP	Post Office Protocol version 3 (POP3)
143	TCP	Internet Message Access Protocol (IMAP)
161	UDP	Simple Network Management Protocol (SNMP)
443	TCP	Hypertext Transfer Protocol Secure (HTTPS)

The netstat Command

Unexplained TCP connections can pose a major security threat. Netstat is an important tool to verify connections.

```
C:\> netstat
Active Connections
Proto Local Address          Foreign Address          State
TCP   192.168.1.124:3126     192.168.0.2:netbios-ssn  ESTABLISHED
TCP   192.168.1.124:3158     207.138.126.152:http     ESTABLISHED
TCP   192.168.1.124:3159     207.138.126.169:http     ESTABLISHED
TCP   192.168.1.124:3160     207.138.126.169:http     ESTABLISHED
TCP   192.168.1.124:3161     sc.msn.com:http          ESTABLISHED
TCP   192.168.1.124:3166     www.cisco.com:http       ESTABLISHED
```

TCP Connection Establishment

Step 1. SYN

The initiating client requests a client-to-server communication session with the server.

Step 2. ACK and SYN

The server acknowledges the client-to-server communication session and requests a server-to-client communication session.

PCB replies to PCA by sending its own syn and also an ack, acknowledging the syn from PCA, and incrementing the sequence number by 1.

Step 3. ACK

The initiating client acknowledges the server-to-client communication session.

The three-way handshake validates that the destination host is available to communicate.

Session Termination

To close a connection, the Finish (FIN) control flag must be set in the segment header. To end each one-way TCP session, a two-way handshake, consisting of a FIN segment and an Acknowledgment (ACK) segment, is used. Therefore, to terminate a single conversation supported by TCP, four exchanges are needed to end both sessions. Either the client or the server can initiate the termination.

Step 1. FIN

When the client has no more data to send in the stream, it sends a segment with the FIN flag set.

PCA sends a fin segment to PCB to end the session when there is no more data to send

1 A B
Send FINFIN receivedA sends FIN request to B.

Step 2. ACK

The server sends an ACK to acknowledge the receipt of the FIN to terminate the session from client to server.

Step 3. FIN

The server sends a FIN to the client to terminate the server-to-client session.

Step 4. ACK

The client responds with an ACK to acknowledge the FIN from the server.

PCA sends an ack in response to PCBs fin which finishes the connection

TCP Three-way Handshake Analysis

Hosts maintain state, track each data segment within a session, and exchange information about what data is received using the information in the TCP header. TCP is a full-duplex protocol, where each connection represents two one-way communication sessions. To establish the connection, the hosts perform a three-way handshake. As shown in the figure, control bits in the TCP header indicate the progress and status of the connection.

These are the functions of the three-way handshake:

- It establishes that the destination device is present on the network.
- It verifies that the destination device has an active service and is accepting requests on the destination port number that the initiating client intends to use.
- It informs the destination device that the source client intends to establish a communication session on that port number.

After the communication is completed the sessions are closed, and the connection is terminated. The connection and session mechanisms enable TCP reliability function.

The six bits in the Control Bits field of the TCP segment header are also known as flags. A flag is a bit that is set to either on or off.

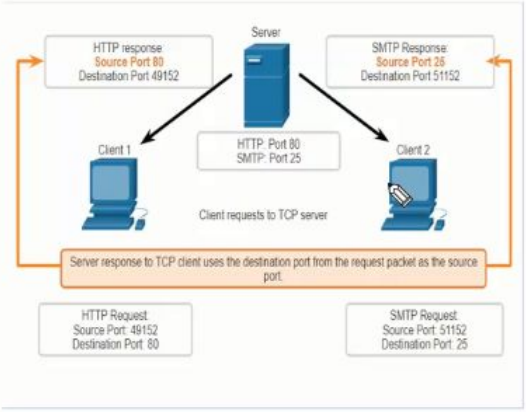
The six control bits flags are as follows:

- **URG** - Urgent pointer field significant
- **ACK** - Acknowledgment flag used in connection establishment and session termination
- **PSH** - Push function
- **RST** - Reset the connection when an error or timeout occurs
- **SYN** - Synchronize sequence numbers used in connection establishment
- **FIN** - No more data from sender and used in session termination

TCP Server Processes

Each application process running on a server is configured to use a port number.

- An individual server cannot have two services assigned to the same port number within the same transport layer services.
- An active server application assigned to a specific port is considered open, which means that the transport layer accepts, and processes segments addressed to that port.
- Any incoming client request addressed to the correct socket is accepted, and the data is passed to the server application.

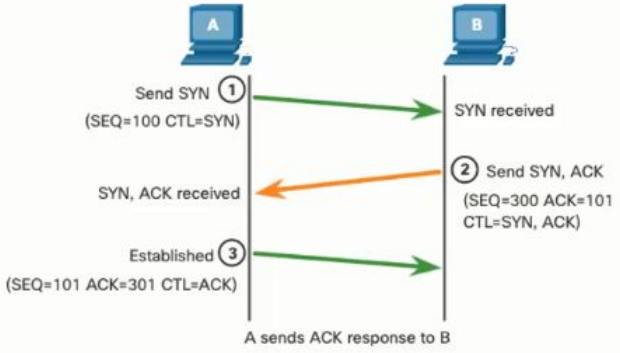


TCP Connection Establishment

Step 1: The initiating client requests a client-to-server communication session with the server.

Step 2: The server acknowledges the client-to-server communication session and requests a server-to-client communication session.

Step 3: The initiating client acknowledges the server-to-client communication session.



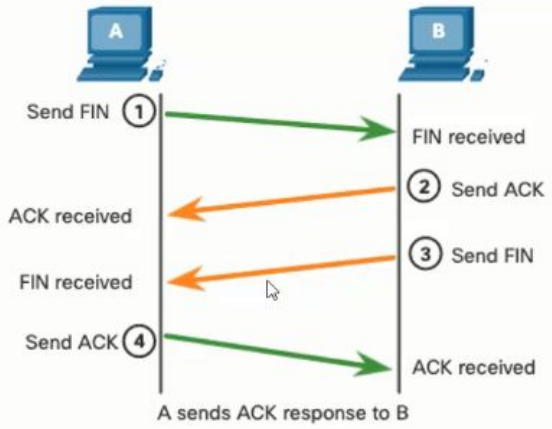
Session Termination

Step 1: When the client has no more data to send in the stream, it sends a segment with the FIN flag set.

Step 2: The server sends an ACK to acknowledge the receipt of the FIN to terminate the session from client to server.

Step 3: The server sends a FIN to the client to terminate the server-to-client session.

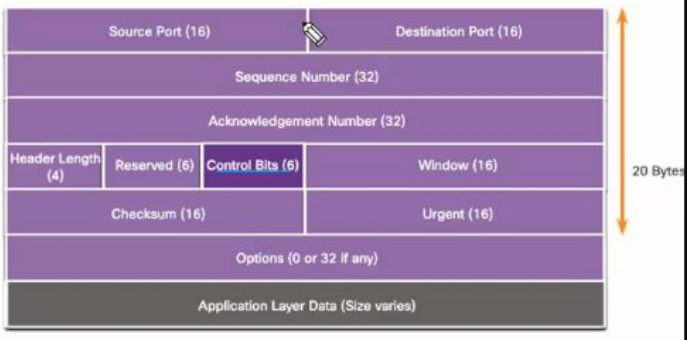
Step 4: The client responds with an ACK to acknowledge the FIN from the server.



TCP Three-Way Handshake Analysis (Cont.)

The six control bit flags are as follows:

- **URG** - Urgent pointer field significant
- **ACK** - Acknowledgment flag used in connection establishment and session termination
- **PSH** - Push function
- **RST** - Reset the connection when an error or timeout occurs
- **SYN** - Synchronize sequence numbers used in connection establishment
- **FIN** - No more data from sender and used in session termination



You have successfully identified the correct answers.

1. The destination port is the well-known port for Simple Mail Transport Protocol, which is 25. This is the port that the mail server will be listening on. The source port is dynamically selected by the requesting client and can be 49152.
2. The three-way handshake consists of a three message exchanges with the following control bit flags: SYN, SYN ACK, and ACK.
3. There are four exchanges to end both sessions between two hosts. (1) Host A sends a FIN. (2) Host B sends an ACK. (3) Host B sends a FIN. (4) Host A sends an ACK.

You answered 3 out of 3 questions correctly.

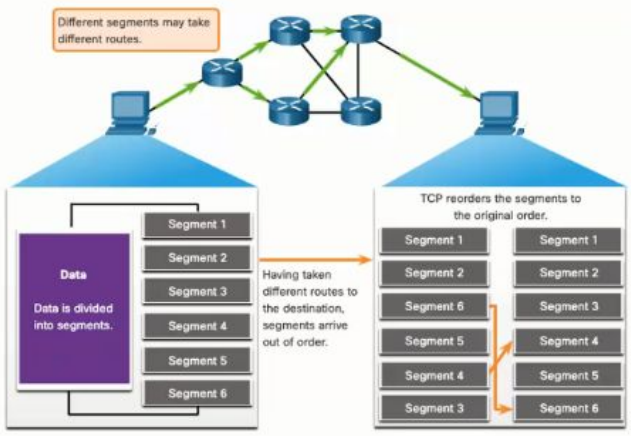
During session setup, an initial sequence number (ISN) is set. This ISN represents the starting value of the bytes that are transmitted to the receiving application. As data is transmitted during the session, the sequence number is incremented by the number of bytes that have been transmitted. This data byte tracking enables each segment to be uniquely identified and acknowledged. Missing segments can then be identified.

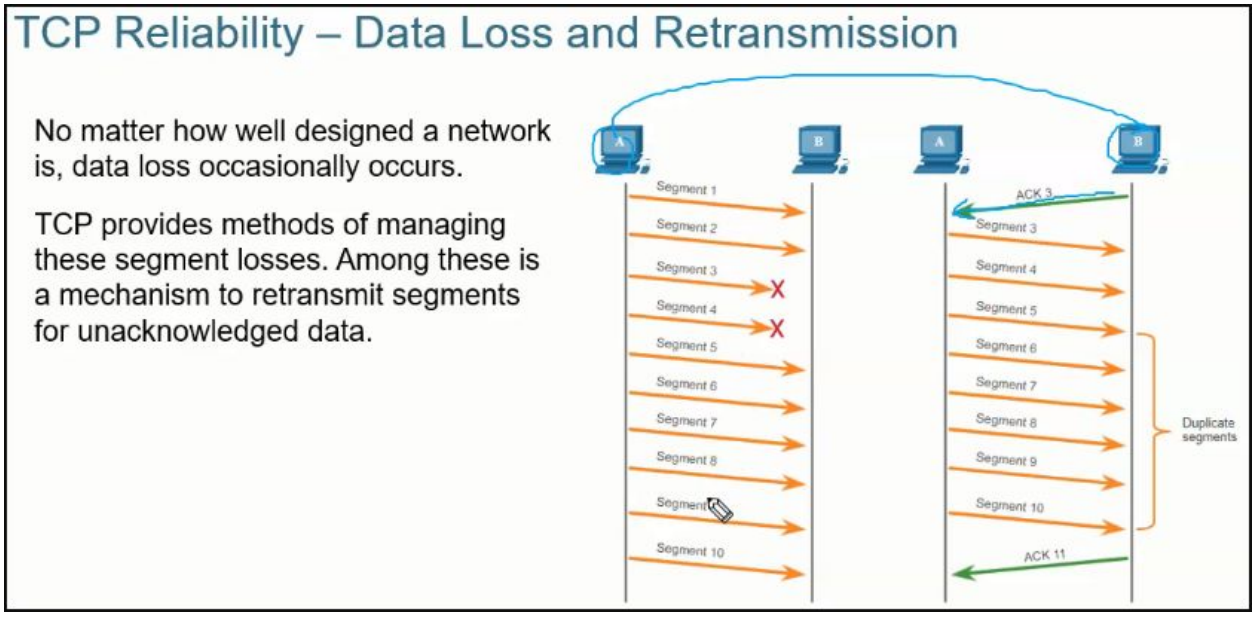
The ISN does not begin at one but is effectively a random number. This is to prevent certain types of malicious attacks. For simplicity, we will use an ISN of 1 for the examples in this chapter.

Segment sequence numbers indicate how to reassemble and reorder received segments,

TCP Reliability- Guaranteed and Ordered Delivery

- TCP can also help maintain the flow of packets so that devices do not become overloaded.
- There may be times when TCP segments do not arrive at their destination or arrive out of order.
- All the data must be received and the data in these segments must be reassembled into the original order.
- Sequence numbers are assigned in the header of each packet to achieve this goal.





The receiving TCP process places the data from a segment into a receiving buffer. Segments are then placed in the proper sequence order and passed to the application layer when reassembled. Any segments that arrive with sequence numbers that are out of order are held for later processing. Then, when the segments with the missing bytes arrive, these segments are processed in order.

TCP Reliability - Data Loss and Retransmission

No matter how well designed a network is, data loss occasionally occurs. TCP provides methods of managing these segment losses. Among these is a mechanism to retransmit segments for unacknowledged data.

The sequence (SEQ) number and acknowledgement (ACK) number are used together to confirm receipt of the bytes of data contained in the transmitted segments. The SEQ number identifies the first byte of data in the segment being transmitted. TCP uses the ACK number sent back to the source to indicate the next byte that the receiver expects to receive. This is called expectational acknowledgement.

Prior to later enhancements, TCP could only acknowledge the next byte expected.
Host operating systems today typically employ an optional TCP feature called selective acknowledgment (SACK), negotiated during the three-way handshake. If both hosts support SACK, the receiver can explicitly acknowledge which segments (bytes) were received including any discontinuous segments. The sending host would therefore only need to retransmit the missing data

TCP Flow Control - Window Size and Acknowledgments

TCP also provides mechanisms for flow control. Flow control is the amount of data that the destination can receive and process reliably. Flow control helps maintain the reliability of TCP transmission by adjusting the rate of data flow between source and destination for a given session. To accomplish this, the TCP header includes a 16-bit field called the window size.

The window size determines the number of bytes that can be sent before expecting an acknowledgment. The acknowledgment number is the number of the next expected byte.

The window size is the number of bytes that the destination device of a TCP session can accept and process at one time. In this example, the PC B initial window size for the TCP session is 10,000 bytes. Starting with the first byte, byte number 1, the last byte PC A can send without receiving an acknowledgment is byte 10,000. This is known as the send window of PC A. The window size is included in every TCP segment so the destination can modify the window size at any time depending on buffer availability.

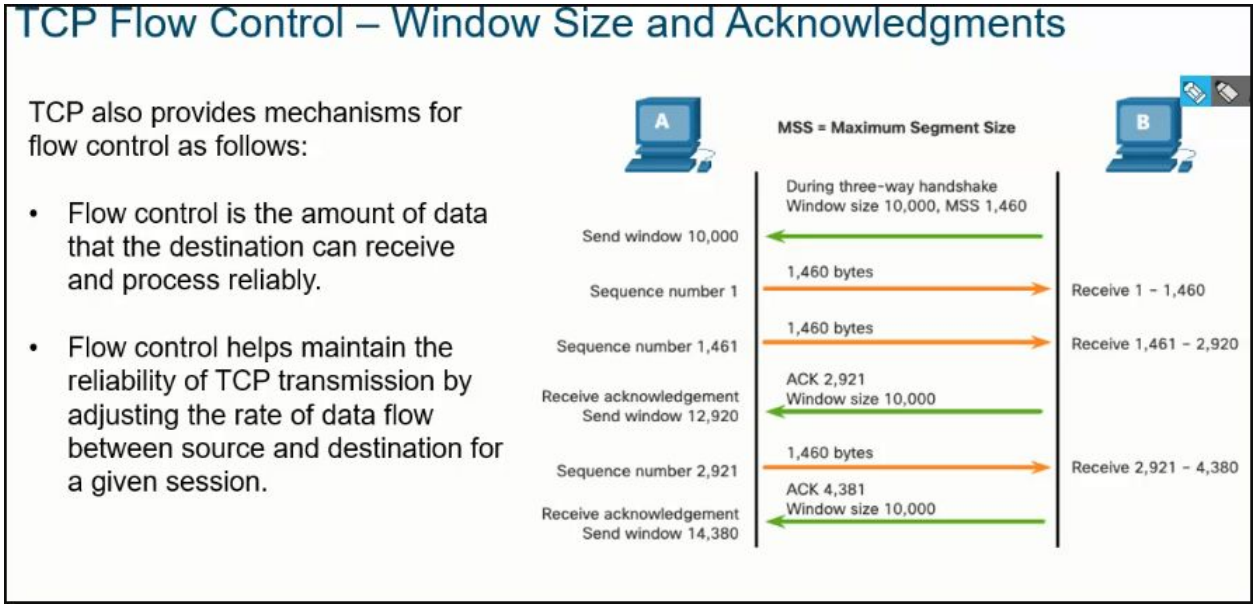
The initial window size is agreed upon when the TCP session is established during the three-way handshake. The source device must limit the number of bytes sent to the destination device based on the window size of the destination. Only after the source device receives an acknowledgment that the bytes have been received, can it continue sending more data for the session. Typically, the destination will not wait for all the bytes for its window size to be received before replying with an acknowledgment. As the bytes are received and processed, the destination will send acknowledgments to inform the source that it can continue to send additional bytes.

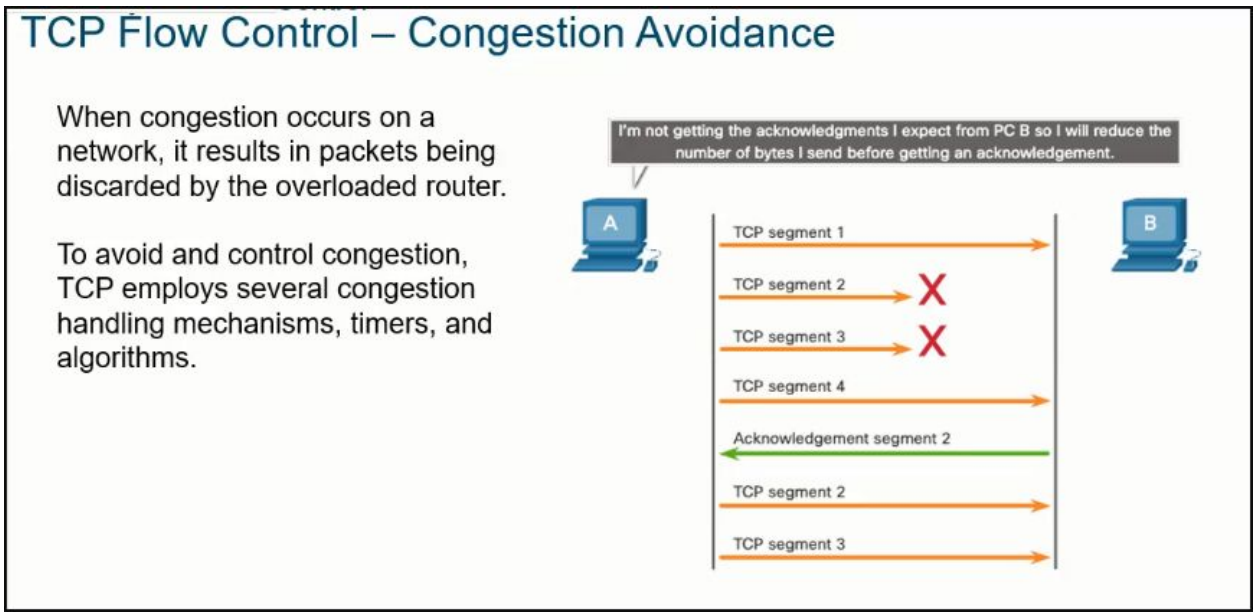
For example, it is typical that PC B would not wait until all 10,000 bytes have been received before sending an acknowledgment. This means PC A can adjust its send window as it receives acknowledgments from PC B. As shown in the figure, when PC A receives an acknowledgment with the acknowledgment number 2,921, which is the next expected byte. The PC A send window will increment 2,920 bytes. This changes the send window from 10,000 bytes to 12,920. PC A can now continue to send up to another 10,000 bytes to PC B as long as it does not send more than its new send window at 12,920.

A destination sending acknowledgments as it processes bytes received, and the continual adjustment of the source send window, is known as sliding windows. In the previous example, the send window of PC A increments or slides over another 2,921 bytes from 10,000 to 12,920.

If the availability of the destination's buffer space decreases, it may reduce its window size to inform the source to reduce the number of bytes it should send without receiving an acknowledgment.

Note: Devices today use the sliding windows protocol. The receiver typically sends an acknowledgment after every two segments it receives. The number of segments received before being acknowledged may vary. The advantage of sliding windows is that it allows the sender to continuously transmit segments, as long as the receiver is acknowledging previous segments. The details of sliding windows are beyond the scope of this course.





TCP Flow Control - Congestion Avoidance

When congestion occurs on a network, it results in packets being discarded by the overloaded router. When packets containing TCP segments do not reach their destination, they are left unacknowledged. By determining the rate at which TCP segments are sent but not acknowledged, the source can assume a certain level of network congestion.

Whenever there is congestion, retransmission of lost TCP segments from the source will occur. If the retransmission is not properly controlled, the additional retransmission of the TCP segments can make the congestion even worse. Not only are new packets with TCP segments

introduced into the network, but the feedback effect of the retransmitted TCP segments that were lost will also add to the congestion. To avoid and control congestion, TCP employs several congestion handling mechanisms, timers, and algorithms.

If the source determines that the TCP segments are either not being acknowledged or not acknowledged in a timely manner, then it can reduce the number of bytes it sends before receiving an acknowledgment. As illustrated in the figure, PC A senses there is congestion and therefore, reduces the number of bytes it sends before receiving an acknowledgment from PC B.

1. The sequence number field is used by the destination host to reassemble segments into the original order.
2. The Window Size field is used to provide flow control.
3. When a sending host senses congestion, it reduces the number of bytes it sends before receiving an acknowledgment from the destination host.

UDP Low Overhead versus Reliability

As explained before, UDP is perfect for communications that need to be fast, like VoIP. This topic explains in detail why UDP is perfect for some types of transmissions. As shown in the figure, UDP does not establish a connection. UDP provides low overhead data transport because it has a small datagram header and no network management traffic.

UDP Datagram Reassembly

Like segments with TCP, when UDP datagrams are sent to a destination, they often take different paths and arrive in the wrong order. UDP does not track sequence numbers the way TCP does. UDP has no way to reorder the datagrams into their transmission order, as shown in the figure.

Therefore, UDP simply reassembles the data in the order that it was received and forwards it to the application. If the data sequence is important to the application, the application must identify the proper sequence and determine how the data should be processed.

UDP Server Processes and Requests

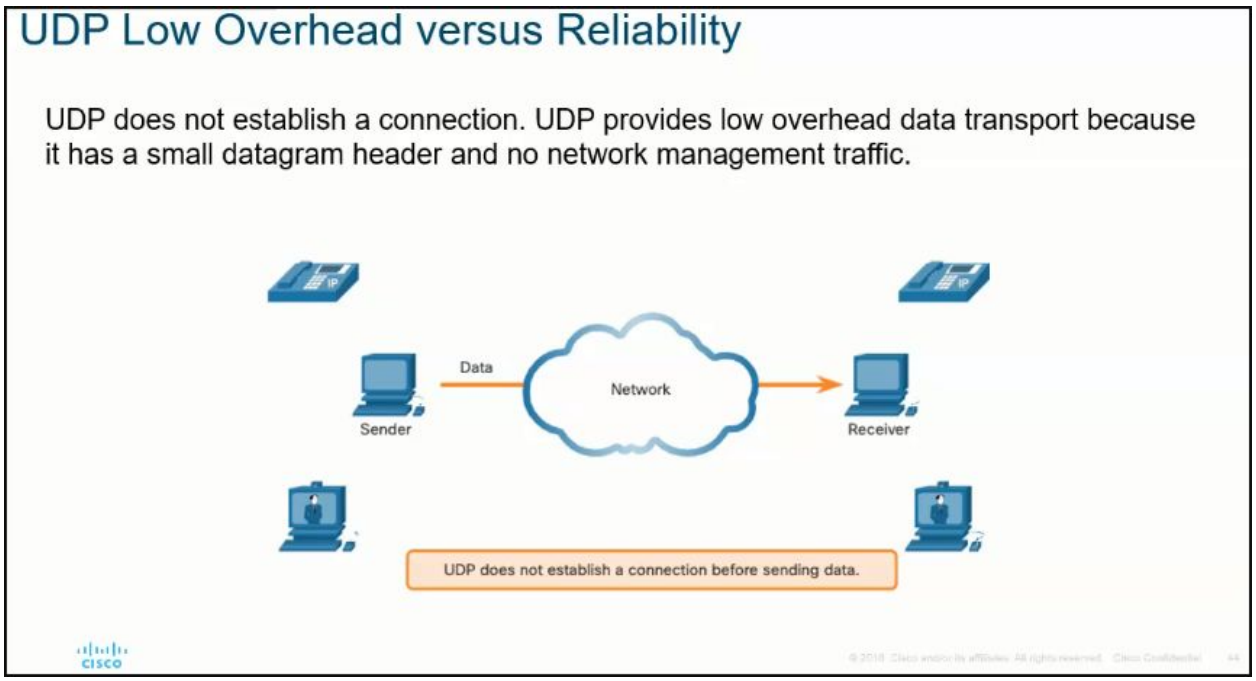
Like TCP-based applications, UDP-based server applications are assigned well-known or registered port numbers, as shown in the figure. When these applications or processes are

running on a server, they accept the data matched with the assigned port number. When UDP receives a datagram destined for one of these ports, it forwards the application data to the appropriate application based on its port number.

UDP Client Processes

As with TCP, client-server communication is initiated by a client application that requests data from a server process. The UDP client process dynamically selects a port number from the range of port numbers and uses this as the source port for the conversation. The destination port is usually the well-known or registered port number assigned to the server process.

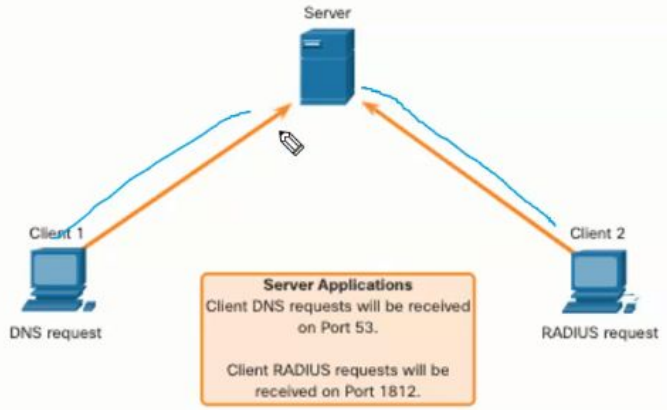
After a client has selected the source and destination ports, the same pair of ports are used in the header of all datagrams in the transaction. For the data returning to the client from the server, the source and destination port numbers in the datagram header are reversed.



UDP Server Processes and Requests

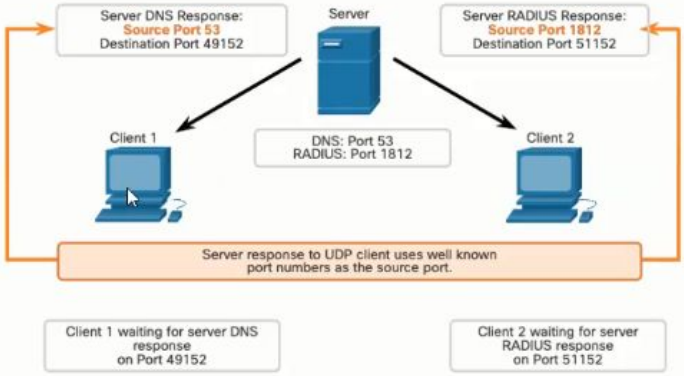
UDP-based server applications are assigned well-known or registered port numbers.

UDP receives a datagram destined for one of these ports, it forwards the application data to the appropriate application based on its port number.



UDP Client Processes

- The UDP client process dynamically selects a port number from the range of port numbers and uses this as the source port for the conversation.
- The destination port is usually the well-known or registered port number assigned to the server process.
- After a client has selected the source and destination ports, the same pair of ports are used in the header of all datagrams in the transaction.



1. UDP is desirable for protocols that make simple request and reply transactions because of its low overhead.
2. UDP reassembles the data in the order that it was received.
3. The correct valid source and destination ports for a host requesting DNS service is Source: 49152, Destination: 53.

What did I learn in this module?

Transportation of Data

The transport layer is the link between the application layer and the lower layers that are responsible for network transmission. The transport layer is responsible for logical communications between applications running on different hosts. The transport layer includes TCP and UDP. Transport layer protocols specify how to transfer messages between hosts and is responsible for managing reliability requirements of a conversation. The transport layer is responsible for tracking conversations (sessions), segmenting data and reassembling segments, adding header information, identifying applications, and conversation multiplexing. TCP is stateful, reliable, acknowledges data, resends lost data, and delivers data in sequenced order. Use TCP for email and the web. UDP is stateless, fast, has low overhead, does not requires acknowledgments, do not resend lost data, and delivers data in the order it arrives. Use UDP for VoIP and DNS.

TCP Overview

TCP establishes sessions, ensures reliability, provides same-order delivery, and supports flow control. A TCP segment adds 20 bytes of overhead as header information when encapsulating the application layer data. TCP header fields are the Source and Destination Ports, Sequence Number, Acknowledgment Number, Header Length, Reserved, Control Bits, Window Size, Checksum, and Urgent. Applications that use TCP are HTTP, FTP, SMTP, and Telnet.

UPD Overview

UDP reconstructs data in the order it is received, lost segments are not resent, no session establishment, and UPD does not inform the sender of resource availability. UDP header fields are Source and Destination Ports, Length, and Checksum. Applications that use UDP are DHCP, DNS, SNMP, TFTP, VoIP, and video conferencing.

Port Numbers

The TCP and UDP transport layer protocols use port numbers to manage multiple simultaneous conversations. This is why the TCP and UDP header fields identify a source and destination application port number. The source and destination ports are placed within the segment. The segments are then encapsulated within an IP packet. The IP packet contains the IP address of the source and destination. The combination of the source IP address and source port number, or the destination IP address and destination port number is known as a socket. The socket is used to identify the server and service being requested by the client. There is a range of port numbers from 0 through 65535. This range is divided into groups: Well-known Ports, Registered Ports, Private and/or Dynamic Ports. There are a few Well-Known Port numbers that are reserved for common applications such at FTP, SSH, DNS, HTTP and others. Sometimes it is necessary to know which active TCP connections are open and running on a networked host. Netstat is an important network utility that can be used to verify those connections.

TCP Communications Process

Each application process running on a server is configured to use a port number. The port number is either automatically assigned or configured manually by a system administrator. TCP

server processes are as follows: clients sending TCP requests, requesting destination ports, requesting source ports, responding to destination port and source port requests. To terminate a single conversation supported by TCP, four exchanges are needed to end both sessions. Either the client or the server can initiate the termination. The three-way handshake establishes that the destination device is present on the network, verifies that the destination device has an active service and is accepting requests on the destination port number that the initiating client intends to use, and informs the destination device that the source client intends to establish a communication session on that port number. The six control bits flags are: URG, ACK, PSH, RST, SYN, and FIN.

Reliability and Flow Control

For the original message to be understood by the recipient, all the data must be received and the data in these segments must be reassembled into the original order. Sequence numbers are assigned in the header of each packet. No matter how well designed a network is, data loss occasionally occurs. TCP provides ways to manage segment losses. There is a mechanism to retransmit segments for unacknowledged data. Host operating systems today typically employ an optional TCP feature called selective acknowledgment (SACK), negotiated during the three-way handshake. If both hosts support SACK, the receiver can explicitly acknowledge which segments (bytes) were received including any discontinuous segments. The sending host would therefore only need to retransmit the missing data. Flow control helps maintain the reliability of TCP transmission by adjusting the rate of data flow between source and destination. To accomplish this, the TCP header includes a 16-bit field called the window size. The process of the destination sending acknowledgments as it processes bytes received and the continual adjustment of the source's send window is known as sliding windows. A source might be transmitting 1,460 bytes of data within each TCP segment. This is the typical MSS that a destination device can receive. To avoid and control congestion, TCP employs several congestion handling mechanisms. It is the source that is reducing the number of unacknowledged bytes it sends and not the window size determined by the destination.

UPD Communication

UDP is a simple protocol that provides the basic transport layer functions. When UDP datagrams are sent to a destination, they often take different paths and arrive in the wrong order. UDP does not track sequence numbers the way TCP does. UDP has no way to reorder the datagrams into their transmission order. UDP simply reassembles the data in the order that it was received and forwards it to the application. If the data sequence is important to the application, the application must identify the proper sequence and determine how the data should be processed. UDP-based server applications are assigned well-known or registered port numbers. When UDP receives a datagram destined for one of these ports, it forwards the application data to the appropriate application based on its port number. The UDP client process dynamically selects a port number from the range of port numbers and uses this as the source port for the conversation. The destination port is usually the well-known or registered port number assigned to the server process. After a client has selected the source and destination ports, the same pair of ports are used in the header of all datagrams used in the

transaction. For the data returning to the client from the server, the source and destination port numbers in the datagram header are reversed.