Chapter 6: Network layer

**6.1 Network Layer Protocols:**

The network layer, or OSI Layer 3, provides services to allow end devices to exchange data across the network. To accomplish this end-to-end transport, the network layer uses four basic processes:

1. **Addressing end devices** - In the same way that a phone has a unique telephone number, end devices must be configured with a unique IP address for identification on the network. An end device with a configured IP address is referred to as a host
2. **Encapsulation -** The network layer receives a protocol data unit (PDU) from the transport layer. In a process called encapsulation, the network layer adds IP header information, such as the IP address of the source (sending) and destination (receiving) hosts. After header information is added to the PDU, the PDU is called a packet.
3. **Routing -** The network layer provides services to direct packets to a destination host on another network. To travel to other networks, the packet must be processed by a router. The role of the router is to select paths for and direct packets toward the destination host in a process known as **routing.** A packet may cross many intermediary devices before reaching the destination host. Each route the packet takes to reach the destination host is called a hop.
4. **De-encapsulation -** When the packet arrives at the network layer of the destination host, the host checks the IP header of the packet. If the destination IP address within the header matches its own IP address, the IP header is removed from the packet. This process of removing headers from lower layers is known as de-encapsulation.

* Difference between transport layer and Network Layer: Transport layer (OSI Layer 4) manages the data transport between the processes running on each host. Network layer protocols specify the packet structure and processing used to carry the data from one host to another host.

There are several network layer protocols in existence; however, only the following two are commonly implemented

Internet Protocol version 4 (IPv4) Internet Protocol version 6 (IPv6)

**Characteristics of the IP Protocol:**

The basic characteristics of IP are:

* **Connectionless -** No connection with the destination is established before sending data packets. The network layer is not concerned with, or even aware of, the type of communication contained inside of a packet. IP is connectionless, meaning that no dedicated end-to-end connection is created before data is sent. IP also does not require additional fields in the protocol data unit (PDU) header to maintain an established connection. This process greatly reduces the overhead of IP. Senders are unaware whether destination devices are present and functional when sending packets, nor are they aware if the destination receives the packet, or if they can access and read the packet
* **Best Effort (unreliable) -** Packet delivery is not guaranteed. Unreliable simply means that IP does not have the capability to manage and recover from undelivered or corrupt packets. This is because while IP packets are sent with information about the location of delivery, it contains no information that can be processed to inform the sender whether delivery was successful. There is no synchronization data included in the packet header for tracking the order of packet delivery. There are also no acknowledgments of packet delivery with IP, and there is no error control data to track whether packets were delivered without corruption. Packets may arrive at the destination corrupted, out of sequence, or not at all. If out-of-order or missing packets create problems for the application using the data, then upper layer services, such as TCP, must resolve these issues. This allows IP to function very efficiently. In the TCP/IP suite, the transport layer can use either TCP or UDP based on the need for reliability in communication. Leaving the reliability decision to the transport layer makes IP more adaptable and accommodating for different types of communication.
* **Media Independent -** Operation is independent of the medium carrying the data. It is the responsibility of the OSI data link layer to take an IP packet and prepare it for transmission over the communications medium. This means that the transport of IP packets is not limited to any medium. one major characteristic of the media that the network layer considers: the maximum size of the PDU that each medium can transport. This characteristic is referred to as the **maximum transmission unit (MTU).** Part of the control communication between the data link layer and the network layer is the establishment of a maximum size for the packet. The data link layer passes the MTU value up to the network layer. The network layer then determines how large packets should be. In some cases, an intermediate device, usually a router, must split up a packet when forwarding it from one medium to a medium with a smaller MTU. This process is called **fragmenting the packet** or **fragmentation**
* **Encapsulating IP-** IP encapsulates the transport layer segment by adding an IP header. This header is used to deliver the packet to the destination host. The process of encapsulating data layer by layer enables the services at the different layers to develop and scale without affecting other layers. Routers can implement these different network layer protocols to operate concurrently over a network to and from the same or different hosts.

**IPv4 Packets:**

An IPv4 packet has two parts: IPv4 Header (Identifies the packet characteristics), Payload (Contains the Layer 4 segment information and the actual data)

**IPv4 Header:**

Significant fields in the IPv4 header include:

1. **Version -** Contains a 4-bit binary value identifying the IP packet version (For IPv4, it is always 0100)
2. **Differentiated Services (DS) -** Formerly called the Type of Service (ToS). 8-bit field used to determine the priority of each packet. The first 6 bits identify the Differentiated Services Code Point (DSCP) value that is used by a quality of service (QoS) mechanism. The last 2 bits identify the explicit congestion notification (ECN) value that can be used to prevent dropped packets during times of network congestion
3. **Time-to-Live (TTL) -** Contains an 8-bit binary value that is used to limit the lifetime of a packet. is commonly referred to as hop count. The packet sender sets the initial time-to-live (TTL) value and is decreased by one each time the packet is processed by a router or hop. If the TTL field decrements to zero, the router discards the packet and sends an Internet Control Message Protocol (ICMP) Time Exceeded message to the source IP address. The **traceroute** command uses this field to identify the routers used between the source and destination.
4. **Protocol -** This 8-bit binary value indicates the data payload type that the packet is carrying, which enables the network layer to pass the data to the appropriate upper-layer protocol.
5. **Source IP Address-** Contains 32-bit binary value represents source IP address of the packet.
6. **Destination IP Address-** Contains 32-bit binary value represents destination IP address of the packet.

**IPv4 Header Fields:**

The fields used to identify and validate the packet include:

1. **Internet Header Length (IHL) -** Contains a 4-bit binary value identifying the number of 32-bit words in the header. The IHL value varies due to the Options and Padding fields. The minimum value for this field is 5 (i.e., 5×32 = 160 bits = 20 bytes) and the maximum value is 15 (i.e., 15×32 = 480 bits = 60 bytes).
2. **Total Length –** (Packet Length) this 16-bit field defines the entire packet (fragment) size, including header and data, in bytes. The minimum length packet is 20 bytes (20-byte header + 0 bytes data), and the maximum is 65,535 bytes.
3. **Header Checksum -** The 16-bit field is used for error checking of the IP header. The checksum of the header is recalculated and compared to the value in the checksum field. If the values do not match, the packet is discarded.

A router may have to fragment a packet when forwarding it from one medium to another medium that has a smaller MTU. When this happens, fragmentation occurs and the IPv4 packet uses the following fields to keep track of the fragments

1. **Identification -** This 16-bit field uniquely identifies the fragment of an original IP packet
2. **Flags -** This 3-bit field identifies how the packet is fragmented. It is used with the Fragment Offset and Identification fields to help reconstruct the fragment into the original packet
3. **Fragment Offset -** This 13-bit field identifies the order in which to place the packet fragment in the reconstruction of the original unfragmented packet.

**Limitations of IPv4:**

1. **IP address depletion -** IPv4 has a limited number of unique public IP addresses available. Although there are approximately 4 billion IPv4 addresses, the increasing number of new IP-enabled devices, always-on connections, and the potential growth of less-developed regions have increased the need for more addresses.
2. **Internet routing table expansion -** A routing table is used by routers to make best path determinations. As the number of servers (nodes) connected to the Internet increases, so too does the number of network routes. These IPv4 routes consume a great deal of memory and processor resources on Internet routers.
3. **Lack of end-to-end connectivity -** Network Address Translation (NAT) is a technology commonly implemented within IPv4 networks. NAT provides a way for multiple devices to share a single public IP address. However, because the public IP address is shared, the IP address of an internal network host is hidden. This can be problematic for technologies that require end-to-end connectivity.

**Advantages of IPv6:**

1. **Increased address space -** IPv6 addresses are based on 128-bit hierarchical addressing as opposed to IPv4 with 32 bits. This dramatically increases the number of available IP addresses.
2. **Improved packet handling -** The IPv6 header has been simplified with fewer fields. This improves packet handling by intermediate routers and provides support for extensions and options for increased scalability/longevity.
3. **Eliminates the need for NAT -** With such many public IPv6 addresses, Network Address Translation (NAT) is not needed. Customer sites, from the largest enterprises to single households, can get a public IPv6 network address. This avoids some of the NAT-induced application problems experienced by applications requiring end-to-end connectivity.
4. **Integrated security -** IPv6 natively supports authentication and privacy capabilities. With IPv4, additional features had to be implemented to do this.

**IPv6 Header:**

The IPv6 simplified header offers several advantages over IPv4:

* Better routing efficiency for performance and forwarding-rate scalability
* No requirement for processing checksums
* Simplified and more efficient extension header mechanisms (as opposed to the IPv4 Options field)
* A Flow Label field for per-flow processing with no need to open the transport inner packet to identify the various traffic flows

**Significant Fields in IPv6 Header:**

1. **Version -** This field contains a 4-bit binary value identifying the IP packet version. For IPv6 packets, this field is always set to 0110.
2. **Traffic Class -** This 8-bit field is equivalent to the IPv4 Differentiated Services (DS) field. It also contains a 6-bit Differentiated Services Code Point (DSCP) value used to classify packets and a 2-bit Explicit Congestion Notification (ECN) used for traffic congestion control.
3. **Flow Label -** This 20-bit field provides a special service for real-time applications. It can be used to inform routers and switches to maintain the same path for the packet flow so that packets are not reordered.
4. **Payload Length -** This 16-bit field is equivalent to the Total Length field in the IPv4 header. It defines the entire packet (fragment) size, including header and optional extensions.
5. **Next Header -** This 8-bit field is equivalent to the IPv4 Protocol field. It indicates the data payload type that the packet is carrying, enabling the network layer to pass the data to the appropriate upper-layer protocol. This field is also used if there are optional extension headers added to the IPv6 packet.
6. **Hop Limit**: - This 8-bit field replaces the IPv4 TTL field. This value is decremented by one by each router that forwards the packet. When the counter reaches 0 the packet is discarded and an ICMPv6 message is forwarded to the sending host, indicating that the packet did not reach its destination.
7. **Source Address -** This 128-bit field identifies the IPv6 address of the sending host.
8. **Destination Address -** This 128-bit field identifies the IPv6 address of the receiving host.

An IPv6 packet may also contain extension headers (EH), which provide optional network layer information. Extension headers are optional and are placed between the IPv6 header and the payload. EHs are used for fragmentation, security, to support mobility, and more.

* **Routing**

**How a Host Remotes:**

**Host Forwarding Decision:** Another role of the network layer is to direct packets between hosts

A host can send a packet to:

* **Itself** - A host can ping itself by sending a packet to a special IPv4 address of 127.0.0.1 which is referred to as the loopback interface. This loopback address is automatically assigned to a host when TCP/IP is running. For Testing Purposes. Any IP within the network 127.0.0.0/8 refers to the local host.
* **Local host** - This is a host on the same network as the sending host. The hosts share the same network address.
* **Remote host** - This is a host on a remote network. The hosts do not share the same network address.

Whether a packet is destined for a local host, or a remote host is determined by the IP address and subnet mask combination of the source (or sending) device compared to the IP address and subnet mask of the destination device.

Devices that are beyond the local network segment are known as remote hosts. When a source device sends a packet to a remote destination device, then the help of routers and routing is needed. Routing is the process of identifying the best path to a destination. The router connected to the local network segment is referred to as the **default gateway**.

* **Default Gateway**

The default gateway is the device that routes traffic from the local network to devices on remote networks. If the host is sending a packet to a device on a different IP network, then the host must forward the packet through the intermediate device to the default gateway. This is because a host device does not maintain routing information, beyond the local network, to reach remote destinations. The default gateway does. The default gateway, which is most often a router, maintains a routing table. A routing table is a data file in RAM that is used to store route information about directly connected network, as well as entries of remote networks the device has learned about. A router uses the information in the routing table to determine the best path to reach those destinations.

The local table of the host typically contains:

* **Direct connection** - This is a route to the loopback interface (127.0.0.1).
* **Local network route** - The network which the host is connected to is automatically populated in the host routing table.
* **Local default route** - The default route represents the route that packets must take to reach all remote network addresses. The default route is created when a default gateway address is present on the host. The default gateway address is the IP address of the network interface of the router that is connected to the local network. The default gateway address can be configured on the host manually or learned dynamically.

It is important to note that the default route, and therefore, the default gateway, is only used when a host must forward packets to a remote network. It is not required, nor even needs to be configured, if only sending packets to devices on the local network.

* **IPv4 Host Routing Table - Host**

Entering the **netstat -r** command or the equivalent **route print** command, displays three sections related to the current TCP/IP network connections:

* **Interface List -** Lists the Media Access Control (MAC) address and assigned interface number of every network-capable interface on the host including Ethernet, Wi-Fi, and Bluetooth adapters.
* **IPv4 Route Table -** Lists all known IPv4 routes, including direct connections, local network, and local default routes.
* **IPv6 Route Table -** Lists all known IPv6 routes, including direct connections, local network, and local default routes.

Notice the output is divided into five columns which identify:

* **Network Destination -** Lists the reachable networks.
* **Netmask -** Lists a subnet mask that informs the host how to determine the network and the host portions of the IP address.
* **Gateway -** Lists the address used by the local computer to get to a remote network destination. If a destination is directly reachable, it will show as “on-link” in this column.
* **Interface -** Lists the address of the physical interface used to send the packet to the gateway that is used to reach the network destination.
* **Metric -** Lists the cost of each route and is used to determine the best route to a destination.
* **IPv4 Host Routing Entries – Host**

**0.0.0.0**

The local default route; that is, all packets with destinations that do not match other specified addresses in the routing table are forwarded to the gateway. Therefore, all non-matching destination routes are sent to the gateway with IP address 192.168.10.1 (R1) exiting from the interface with IP address 192.168.10.10. Note that the destination address specified in the packet does not change; rather, the host simply knows to forward the packet to the gateway for further processing.

**127.0.0.0 – 127.255.255.255**

These loopback addresses all relate to the direct connection and provide services to the local host.

**192.168.10.0 - 192.168.10.255**

These addresses all relate to the host and local network. All packets with destination address that fall into this category will exit out of the 192.168.10.10 interface.

* **192.168.10.0 -** The local network route address; represents all computers on the 192.168.10.x network.
* **192.168.10.10 -** The address of the local host.
* **192.168.10.255 -** The network broadcast address; sends messages to all hosts on the local network route.

**224.0.0.0**

These are special multicast class D addresses reserved for use through either the loopback interface (127.0.0.1) or the host IP address (192.168.10.10).

**255.255.255.255**

The last two addresses represent the limited broadcast IP address values for use through either the loopback interface (127.0.0.1) or the host IP address (192.168.10.10). These addresses can be used to find a DHCP server before the local IP is determined.

* **IPv6 Route Table - Host**

The IPv6 Route Table section displays four columns which identify:

* **If -** Lists the interface numbers from the Interface List section of the **netstat –r** command. The interface numbers correspond to the network capable interface on the host, including Ethernet, Wi-Fi, and Bluetooth adapters.
* **Metric -** Lists the cost of each route to a destination. Lower numbers indicate preferred routes.
* **Network Destination -** Lists the reachable networks.
* **Gateway -** Lists the address used by the local host to forward packets to a remote network destination. On-link indicates that the host is currently connected to it.

the IPv6 Route section generated by the **netstat –r** command to reveal the following network destinations:

* **::/0 -** This is the IPv6 equivalent of the local default route.
* **::1/128 -** This is equivalent to the IPv4 loopback address and provides services to the local host.
* **2001::/32 -** This is the global unicast network prefix.
* **2001:0:9d38:953c:2c30:3071:e718:a926/128 -** This is the global unicast IPv6 address of the local computer.
* **fe80::/64 -** This is the local link network route address and represents all computers on the local link IPv6 network.
* **fe80::2c30:3071:e718:a926/128 -** This is the link local IPv6 address of the local computer.
* **ff00::/8 -** These are special reserved multicast class D addresses equivalent to the IPv4 224.x.x.x addresses.

**Router Packet Forwarding Decision:**

When a host sends a packet to another host, it will use its routing table to determine where to send the packet. If the destination host is on a remote network, the packet is forwarded to the address of a gateway device.

What happens when a packet arrives on a router interface? The router looks at its routing table to determine where to forward packets. The routing table of a router stores information about:

* **Directly-connected routes -** These routes come from the active router interfaces. Routers add a directly connected route when an interface is configured with an IP address and is activated. Each of the router's interfaces is connected to a different network segment. Routers maintain information about the network segments that they are connected to within the routing table.
* **Remote routes -** These routes come from remote networks connected to other routers. Routes to these networks can either be manually configured on the local router by the network administrator or dynamically configured by enabling the local router to exchange routing information with other routers using dynamic routing protocols.

**Router Routing Tables:**

The routing table of a router is similar to the routing table of a host. They both identify the:

* Destination network
* Metric associated with the destination network
* Gateway to get to the destination network

the **show ip route** command can be used to display the routing table of a router. A router also provides additional route information, including how the route was learned, when it was last updated, and which specific interface to use to get to a predefined destination.

When a packet arrives at the router interface, the router examines the packet header to determine the destination network. If the destination network matches a route in the routing table, the router forwards the packet using the information specified in the routing table. If there are two or more possible routes to the same destination, the metric is used to decide which route appears on the routing table.

**Directly Connected Routing Tables:**

Two routing table entries are automatically created when an active router interface is configured with an IP address and subnet mask.

1. Route Source:

* **C -** Identifies a directly connected network. Directly connected networks are automatically created when an interface is configured with an IP address and activated.
* **L -** Identifies that this is a link local route. Link local routes are automatically created when an interface is configured with an IP address and activated.

1. **Destination network:** It identifies the address of the remote network.
2. **Outgoing interface:** It identifies the exit interface to use when forwarding packets to the destination network.

A router typically has multiple interfaces configured. The routing table stores information about both directly-connected and remote routes. As with directly connected networks, the route source identifies how the route was learned.

common codes for remote networks include:

* **S -** Identifies that the route was manually created by an administrator to reach a specific network. This is known as a static route.
* **D -** Identifies that the route was learned dynamically from another router using the Enhanced Interior Gateway Routing Protocol (EIGRP).
* **O -** Identifies that the route was learned dynamically from another router using the Open Shortest Path First (OSPF) routing protocol.

**Remote Network Routing Tables:**

The entry identifies the following information:

* **Route source -** Identifies how the route was learned.
* **Destination network -** Identifies the address of the remote network.
* **Administrative distance -** Identifies the trustworthiness of the route source.
* **Metric -** Identifies the value assigned to reach the remote network. Lower values indicate preferred routes.
* **Next-hop -** Identifies the IP address of the next router to forward the packet.
* **Route timestamp -** Identifies when the route was last heard from.
* **Outgoing interface -** Identifies the exit interface to use to forward a packet toward the final destination.

**Next-Hop Address:** A next hop is the address of the device that will process the packet next. For a host on a network, the address of the default gateway (router interface) is the next hop for all packets that must be sent to another network. In the routing table of a router, each route to a remote network lists a next hop.

When a packet destined for a remote network arrives at the router, the router matches the destination network to a route in the routing table. If a match is found, the router forwards the packet to the IP address of the next hop router using the interface identified by the route entry.

Networks directly connected to a router have no next-hop address, because a router can forward packets directly to hosts on these networks using the designated interface. just as a host can use a default gateway to forward a packet to an unknown destination, a router can also be configured to use a default static route to create a Gateway of Last Resort.

**Anatomy of a Router:**

**Router Memory:**

* 1. **RAM**

RAM is used to store various applications and processes including:

* **Cisco IOS -** The IOS is copied into RAM during bootup.
* **Running configuration file -** This is the configuration file that stores the configuration commands that the router IOS is currently using. It is also known as the running-config.
* **IP routing table -** This file stores information about directly-connected and remote networks. It is used to determine the best path to use to forward packets.
* **ARP cache -** This cache contains the IPv4 address to MAC address mappings, similar to the Address Resolution Protocol (ARP) cache on a PC. The ARP cache is used on routers that have LAN interfaces, such as Ethernet interfaces.
* **Packet buffer -** Packets are temporarily stored in a buffer when received on an interface or before they exit an interface.

dynamic random-access memory (DRAM). DRAM is a very common kind of RAM that stores the instructions and data needed to be executed by the CPU. Unlike ROM, RAM is volatile memory and requires continual power to maintain its information. It loses all of its content when the router is powered down or restarted.

* 1. **ROM**

Cisco routers use ROM to store:

* **Bootup instructions -** Provides the startup instructions.
* **Basic diagnostic software -** Performs the power-on self-test (POST) of all components.
* **Limited IOS -** Provides a limited backup version of the OS, in case the router cannot load the full featured IOS.

ROM is firmware embedded on an integrated circuit inside the router and does not lose its contents when the router loses power or is restarted.

* 1. **NVRAM**

NVRAM is used by the Cisco IOS as permanent storage for the startup configuration file (startup-config). Like ROM, NVRAM does not lose its contents when power is turned off.

* 1. **Flash Memory**

Flash memory is non-volatile computer memory used as permanent storage for the IOS and other system related files. The IOS is copied from flash into RAM during the bootup process

**Router Backplane:**

A Cisco 1941 router includes the following connections:

* **Console ports -** Two console ports for the initial configuration and command-line interface (CLI) management access using a regular RJ-45 port and a new USB Type-B (mini-B USB) connector.
* **AUX port -** An RJ-45 port for remote management access; this is similar to the Console port.
* **Two LAN interfaces -** Two Gigabit Ethernet interfaces for LAN access.
* **Enhanced high-speed WAN interface card (EHWIC) slots -** Two slots that provide modularity and flexibility by enabling the router to support different types of interface modules, including Serial, digital subscriber line (DSL), switch port, and wireless.

**Connecting Router:** These devices have several types of ports and interfaces. These ports and interfaces are used to connect cables to the device.

The connections on a Cisco router can be grouped into two categories:

* **Management ports -** These are the console and auxiliary ports used to configure, manage, and troubleshoot the router. Unlike LAN and WAN interfaces, management ports are not used for packet forwarding.
* **Inband Router interfaces -** These are the LAN and WAN interfaces configured with IP addressing to carry user traffic. Ethernet interfaces are the most common LAN connections, while common WAN connections include serial and DSL interfaces.

**LAN and WAN interfaces:**

The most common methods are:

* **Console** - Uses a low speed serial or USB connection to provide direct connect, out-of-band management access to a Cisco device.
* **Telnet or SSH** - Two methods for remotely accessing a CLI session across an active network interface.
* **AUX port** - Used for remote management of the router using a dial-up telephone line and modem.

Router interfaces can be grouped into two categories:

* **Ethernet LAN interfaces** - Used for connecting cables that terminate with LAN devices, such as computers and switches. This interface can also be used to connect routers to each other. Several conventions for naming Ethernet interfaces are popular: the older Ethernet, FastEthernet, and Gigabit Ethernet. The name used depends on the device type and model.
* **Serial WAN interfaces** - Used for connecting routers to external networks, usually over a larger geographical distance. Similar to LAN interfaces, each serial WAN interface has its own IP address and subnet mask, which identifies it as a member of a specific network.

**Router Boot-up**

Cisco IOS for routers provides the following:

* Addressing
* Interfaces
* Routing
* Security
* QoS
* Resources Management

**Bootset Files:**

A router loads the following two files into RAM when it is booted:

* **IOS image file -** The IOS facilitates the basic operation of the device’s hardware components. The IOS image file is stored in flash memory.
* **Startup configuration file -** The startup configuration file contains commands that are used to initially configure a router and create the running configuration file stored in in RAM. The startup configuration file is stored in NVRAM. All configuration changes are stored in the running configuration file and are implemented immediately by the IOS.

**Router Bootup Process:**

There are three major phases to the bootup process that is:

**1.** **Perform the POST and load the bootstrap program**- The Power-On Self Test (POST) is a common process that occurs on almost every computer during bootup. The POST process is used to test the router hardware. When the router is powered on, software on the ROM chip conducts the POST. During this self-test, the router executes diagnostics from ROM on several hardware components, including the CPU, RAM, and NVRAM. After the POST has been completed, the router executes the bootstrap program.

After the POST, the bootstrap program is copied from ROM into RAM. Once in RAM, the CPU executes the instructions in the bootstrap program. The main task of the bootstrap program is to locate the Cisco IOS and load it into RAM.

**2.** **Locate and load the Cisco IOS software-** The IOS is typically stored in flash memory and is copied into RAM for execution by the CPU. During self-decompression of the IOS image file, a string of pounds signs (#) will be displayed.

If the IOS image is not located in flash, then the router may look for it is using a TFTP server. If a full IOS image cannot be located, a scaled-down version of the IOS is copied from ROM into RAM. This version of IOS is used to help diagnose any problems and can be used to load a complete version of the IOS into RAM.

**3. Locate and load the startup configuration file or enter setup mode-** The bootstrap program then searches for the startup configuration file (also known as startup-config), in NVRAM. This file has the previously saved configuration commands and parameters. If it exists, then it is copied into RAM as the running configuration file, running-config. The running-config file contains interface addresses, starts routing processes, configures router passwords, and defines other characteristics of the router.

If the startup-config file does not exist in NVRAM, the router may search for a Trivial File Transfer Protocol (TFTP) server. If the router detects that it has an active link to another configured router, it sends a broadcast searching for a configuration file across the active link.

If a TFTP server is not found, then the router displays the setup mode prompt. Setup mode is a series of questions prompting the user for basic configuration information. Setup mode is not intended to be used to enter complex router configurations, and it is not commonly used by network administrators.

**Show Version Output:**

You can use the **show version** command to verify and troubleshoot some of the basic hardware and software components of the router.

The output from the **show version** command includes:

* **IOS version -** Version of the Cisco IOS software in RAM and that is being used by the router.
* **ROM Bootstrap Program -** Displays the version of the system bootstrap software, stored in ROM that was initially used to boot up the router.
* **Location of IOS -** Displays where the bootstrap program is located and loaded the Cisco IOS, and the complete filename of the IOS image.
* **CPU and Amount of RAM -** The first part of this line displays the type of CPU on this router. The last part of this line displays the amount of DRAM. Some series of routers, like the Cisco 1941 ISR, use a fraction of DRAM as packet memory. Packet memory is used for buffering packets. To determine the total amount of DRAM on the router, add both numbers.
* **Interfaces -** Displays the physical interfaces on the router. In this example, the Cisco 1941 ISR has two Gigabit Ethernet interfaces and two low-speed serial interfaces.
* **Amount of NVRAM and Flash -** This is the amount of NVRAM and the amount of flash memory on the router. NVRAM is used to store the startup-config file and flash is used to permanently store the Cisco IOS.

**Configuring Interfaces:**

To enable a router interface, configure the following:

* **IPv4 address and subnet mask -** Configures the IP address and subnet mask using the

**ip address** *ip-address subnet-mask* interface configuration command.

* **Activate the interface -** By default, LAN and WAN interfaces are not activated. The interface must be activated using the **no shutdown** command. This is similar to powering on the interface. The interface must also be connected to another device (a hub, a switch, or another router) for the physical layer to be active.

Other interface verification commands include:

* **show ip route** **-** Displays the contents of the IPv4 routing table stored in RAM.
* **show interfaces** **-** Displays statistics for all interfaces on the device.
* **show ip interface** **-** Displays the IPv4 statistics for all interfaces on a router.

**Configuring Default Gateway on a Host**

Most routers have, at a minimum, two interfaces. Each interface is configured with a separate IP address in a separate network.

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 is only used when the host wants to send a packet to a device on another network. The default gateway address is generally the router interface address attached to the local network of the host. While it does not matter what address is actually configured on the router interface, the IP address of the host device and the router interface address must be in the same network.

The figures display a topology of a router with two separate interfaces. Each interface is connected to a separate network. G0/0 is connected to network 192.168.10.0, while G0/1 is connected to network 192.168.11.0. Each host device is configured with the appropriate default gateway address.

In Figure 1, PC1 sends a packet to PC2. In this example, the default gateway is not used; rather, PC1 addresses the packet with the IP address of PC2 and forwards the packet directly to PC2 through the switch.

In Figure 2, PC1 sends a packet to PC3. In this example, PC1 addresses the packet with the IP address of PC3, but then forwards the packet to the router. The router accepts the packet, accesses its route table to determine the appropriate exit interface based on the destination address, and then forwards the packet out of the appropriate interface to reach PC3.

**Configuring Default Gateway on Switch**

A default gateway is used by all devices that require the use of a router to determine the best path to a remote destination. End devices require default gateway addresses, but so do intermediate devices, such as the Cisco IOS switch.

The IP address information on a switch is only necessary to manage the switch remotely. In other words, to be able to telnet to the switch, the switch must have an IP address to Telnet to. If the switch is only accessed from devices within the local network, only an IP address is required.

Configuring the IP address on a switch is done on the switch virtual interface (SVI):

S1(config)# **interface vlan1**

S1(config-if)# **ip address 192.168.10.50 255.255.255.0**

S1(config-if)# **no shut**

However, if the switch must be accessible by devices in a different network, the switch must be configured with a default gateway address, because packets that originate from the switch are handled just like packets that originate from a host device. Therefore, packets that originate from the switch and are destined for a device on the same network are forwarded directly to the appropriate device. Packets that originate from the switch and are destined for a device on a remote network must be forwarded to the default gateway for path determination.

To configure a default gateway on a switch, use the following global configuration command:

S1(config)# **ip default-gateway 192.168.10.1**

Figure 1 shows an administrator connecting to a switch on a remote network. For the switch to forward response packets to the administrator, the default gateway must be configured.

A common misconception is that the switch uses its configured default gateway address to determine where to forward packets originating from hosts connected to the switch and destined for hosts on a remote network. Actually, the IP address and default gateway information is only used for packets that originate from the switch. Packets originating from hosts connected to the switch must already have default gateway information configured to communicate on remote networks. In Figure 2, practice configuring a default gateway on a switch.

Graphical user interface, application

Description automatically generated