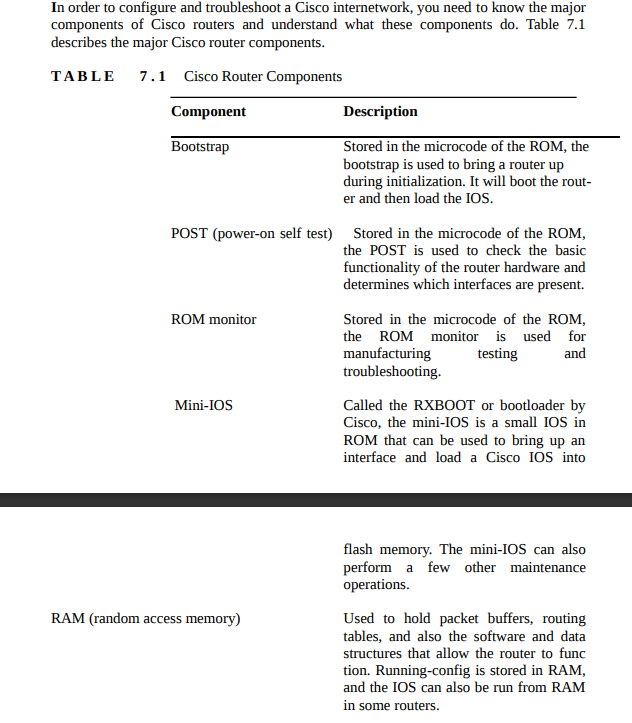
**Unit 5 ----- 7 marks**

1. **What are the component of router? Explain any four of them**



1. **Explain the booting procedure of router?**

The Router Boot Sequence

When a router boots up, it performs a series of steps, called the boot sequence, to test the

hardware and load the necessary software. The boot sequence consists of the following

steps:

The router performs a POST. The POST tests the hardware to verify that all components

of the device are operational and present. For example, the POST checks for the

different interfaces on the router. The POST is stored in and run from ROM.

The bootstrap looks for and loads the Cisco IOS software. The boot-strap is a program in

ROM that is used to execute programs. The bootstrap program is responsible for finding

where each IOS program is located and then loading the file. By default, the IOS

software is loaded from flash memory in all Cisco routers.

The IOS software looks for a valid configuration file stored in NVRAM. This file is

called startup-config and is only there if an administrator copies the running-config file

into NVRAM.

If a startup-config file is in NVRAM, the router will load and run this file. The router is

now operational. If a startup-config file is not in NVRAM, the router will start the setup

mode configuration upon bootup.

**5) What is a Configuration Register? What is its significance? Explain**

The configuration register in a Cisco router is a 16-bit parameter that determines how the router functions during the boot process. It defines various parameters, such as how the router boots, where it loads its operating system from, and other bootup behaviors.

Significance of the Configuration Register:

1. \*Boot Behavior:\* It specifies how the router should boot and which storage device or source it should use to load the operating system.

2. \*Console Speed:\* It determines the console communication speed during the boot process.

3. \*Password Recovery:\* It can be used to bypass passwords or enable password recovery modes.

4. \*Diagnostic Modes:\* The configuration register also facilitates different diagnostic and bootup modes, allowing the router to bypass certain configurations or load alternate images.

5. \*Redundancy and Failover:\* It is used in redundant or failover scenarios, like when a router is part of a High Availability setup.

Changing the configuration register value can alter the router's behavior during the boot process, allowing technicians to troubleshoot boot issues, perform password recovery, or make specific adjustments to the boot sequence. Understanding and modifying the configuration register is crucial for network administrators and technicians in managing and troubleshooting Cisco routers.

**7) Give the procedure to recover a password for a Router.**

If you are locked out of a router because you forgot the password, you can change the configuration register to help you recover. As noted earlier, bit 6 in the configuration register is used to tell the router whether to use the contents of NVRAM to load a router configuration. The default configuration register value for bit 6 is 0x2102, which means that bit 6 is off. With the default setting, the router will look for and load a router configuration stored in NVRAM (startup-config). To recover a password, you need to turn on bit 6, which will tell the router to ignore the NVRAM contents. The

configuration register value to turn on bit 6 is 0x2142. Here are the main steps to password recovery:

1. Boot the router and interrupt the boot sequence by performing a break.
2. Change the configuration register to turn on bit 6 (with the value 0x2142).
3. Reload the router.
4. Enter privileged mode.
5. Copy the startup-config file to running-config.
6. Change the password.
7. Reset the configuration register to the default value.
8. Reload the router.

**13) Give the procedure to backup and restore CISCO IOS in a Router.**

Before you upgrade or restore a Cisco IOS, you should copy the exist-ing file to a

TFTP host as a backup in case the new image does not work. You can use any

TFTP host to perform this function. By default, the flash mem-ory in a router is

used to store the Cisco IOS. The following sections describe how to check the

amount of flash memory, copy the Cisco IOS from flash memory to a TFTP host,

and then copy the IOS from a TFTP host to flash memory.

Verifying Flash Memory

Before you attempt to upgrade the Cisco IOS on your router with a new IOS file,

you should verify that your flash memory has enough room to hold the new image.

You can verify the amount of flash memory and the file or files being stored in

flash memory by using the show flash command (sh flash for short):

Router#sh flash

System flash directory:

File LengthName/status

8121000 c2500-js-l.112-18.bin

[8121064 bytes used, 8656152 available, 16777216 total] 16384K bytes of

processor board System flash (Read ONLY) Router#

Notice that the filename in this example is c2500-js-l.112-18.bin.

The name of the file is platform-specific and is derived as follows:

C 2500 is the platform.

j indicates that the file is an enterprise image.

S indicates the file contains extended capabilities.

L indicates that the file can be moved from flash memory if needed and is not

compressed.

11.2-18 is the revision number.

.bin indicates that the Cisco IOS is a binary executable file.

The last line in the router output shows that the flash is 16,384KB (or 16MB). So if

the new file that you want to use is, say, 10MB in size, you know that there is

plenty of room for it. Once you verify that flash mem-ory can hold the IOS you

want to copy, you can continue with your backup operation.

Backing Up the Cisco IOS

To back up the Cisco IOS to a TFTP host, you use the command copy flash tftp.

This is a straightforward command that requires only the source file-name and the

IP address of the TFTP host.

The key to success in this backup routine is to make sure that you have good

connectivity to the TFTP host. You can check this by pinging the device from the

router console prompt, as in the following example:

Router#ping 192.168.0.120

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.0.120, timeout is 2 seconds: !!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/8 ms

The Ping (Packet Internet Groper) utility is used to test network connectivity. It is

used in some examples in this chapter and discussed in more detail in the

“Checking Network Connectivity” section later in this chapter.

After you ping the TFTP host to make sure that IP is working, you can use the copy

flash tftp command to copy the IOS to the TFTP host, as shown below. Notice that

after you enter the command, the name of the file in flash memory is displayed.

This makes it easy for you. You can copy the filename and then paste it when

prompted for the source filename.

Router#copy flash tftp

System flash directory:

File LengthName/status

8121000 c2500-js-l.112-18.bin

[8121064 bytes used, 8656152 available, 16777216 total] Address or name of

remote host [255.255.255.255]?

192.168.0.120

Source file name? c2500-js-l.112-18.bin

Destination file name [c2500-js-l.112-18.bin]? (press enter)

Verifying checksum for 'c2500-js-l.112-18.bin')file #1)...OK

Copy '/c2500-js-l.112-18' from Flash to server as '/c2500-js-l.112-18'? [yes/no]y

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

!!!!!!!!!!!!!!!!!!! [output cut] Upload to server done

Flash copy took 00:02:30 [hh:mm:ss] Router#

In this example, the content of flash memory was copied successfully to the TFTP

host. The address of the remote host is the IP address of the TFTP host. The source

filename is the file in flash memory.

The copy flash tftp command does not prompt you for the location of any file or

ask you where to put the file. TFTP is the “grab it and place it” program in this

situation. The TFTP host must have a default directory specified, or it won’t work.

**14) Give the procedure to backup and restore CISCO configurations in a Router**

To copy the router’s configuration from a router to a TFTP host, you can use either

the copy running-config tftp or copy starting-config tftp command. Either

command will back up the router configuration that is cur-rently running in DRAM

or that is stored in NVRAM.

Verifying the Current Configuration

To verify the configuration in DRAM, use the show running-config com-mand (sh

run for short), as follows:

Router#sh run

Building configuration...

Current configuration:

!

version 12.0

The current configuration information indicates that the router is now running

version 12.0 of the IOS.

Verifying the Stored Configuration

Next, you should check the configuration stored in NVRAM. To see this, use the

show starting-config command (sh start for short), as follows:

Router#sh start

Using 366 out of 32762 bytes

!

version 11.2

The second line shows how much room your backup configuration is using. In this

example, NVRAM is 32KB and only 366 bytes of it are used. Notice that the

version of configuration in NVRAM is 11.2 (because I have not copied runningconfig to startup-config since upgrading the router).

If you are not sure that the files are the same, and the running-config file is what

you want to use, then use the copy running-config startup-config to make sure both

files are the same, as described in the next section.

Copying the Current Configuration to NVRAM

By copying running-config to NVRAM as a backup, as shown in the fol-lowing

output, you are assured that your running-config will always be reloaded if the

router gets rebooted. In the new IOS version 12.0, you are prompted for the

filename you want to use. Also, in this example, since the version of IOS was 11.2

the last time a copy run start was performed, the router will let you know that it is

going to replace that file with the new 12.0 version.

Router#copy run start

Destination filename [startup-config]? (press enter)

Warning: Attempting to overwrite an NVRAM configuration previously written by

a different version of the system image.

Overwrite the previous NVRAM configuration?[confirm](press enter)

Building configuration...

[OK]

Now when you run show starting-config, the version shows 12.0:

Router#sh start

Using 487 out of 32762 bytes

!

version 12.0

Copying the Configuration to a TFTP Host

Once the file is copied to NVRAM, you can make a second backup to a TFTP host

by using the copy running-config tftp command (copy run tftp for short), as

follows:

Router#copy run tftp

Address or name of remote host []? 192.168.0.120

Destination filename [router-confg]? todd1-confg

!!

487 bytes copied in 12.236 secs (40 bytes/sec) Router#

Notice that this took only two exclamation points (!!), which are two UDP

acknowledgments. In this example, I named the file todd1-confg because I had not

set a hostname for the router. If you have a hostname con-figured, the command

will automatically use the hostname plus the exten-sion –confg as the name of the

file.

Restoring the Cisco Router Configuration

If you have changed your router’s running-config and want to restore the

configuration to the version in startup-config, the easiest way to do this is to use

the copy startup-config running-config command (copy start run for short). You

can also use the older Cisco command, config mem, to restore a configuration. Of

course, this will work only if you first copied running-config into NVRAM before

making any changes.

If you copied the router’s configuration to a TFTP host as a sec-ond backup, you

can restore the configuration using the copy tftp running-config command (copy

tftp run for short) or the copy tftp startup-config command (copy tftp start for

short), as shown below. Remember that the old command that provides this

function is config net.

Router#copy tftp run

Address or name of remote host []? 192.168.0.120 Source filename []? todd1-

confg

Destination filename [running-config]? (press enter) Accessing

tftp://192.168.0.120/todd1-confg...

Loading todd1-confg from 192.168.0.120 (via Ethernet0):

!!

[OK - 487/4096 bytes]

487 bytes copied in 5.400 secs (97 bytes/sec) Router#

00:38:31: %SYS-5-CONFIG: Configured from tftp:// 192.168.0.120/todd1-confg

Router#

The configuration file is an ASCII text file. This means that before you copy the

configuration stored on a TFTP host back to a router, you can make changes to the

file with any text editor.

**15) How to configure the Routers interface with a IP address? Explain**

To configure a router's interface with an IP address, follow these steps in Cisco's command-line interface (CLI):

1. \*Access Configuration Mode:\*

markdown

enable

configure terminal

2. \*Select the Interface:\*

Identify the interface you want to configure (for example, GigabitEthernet0/0).

markdown

interface interface\_name

Replace interface\_name with the specific interface, such as GigabitEthernet0/0, FastEthernet0/0, etc.

3. \*Enable the Interface and Assign an IP Address:\*

markdown

ip address <IP\_address> <subnet\_mask>

Replace <IP\_address> with the desired IP address and <subnet\_mask> with the subnet mask (e.g., 255.255.255.0).

4. \*Activate the Interface:\*

markdown

no shutdown

5. \*Exit and Save Configurations:\*

markdown

exit

write memory

Here's an example of configuring an interface:

markdown

enable

configure terminal

interface GigabitEthernet0/0

ip address 192.168.1.1 255.255.255.0

no shutdown

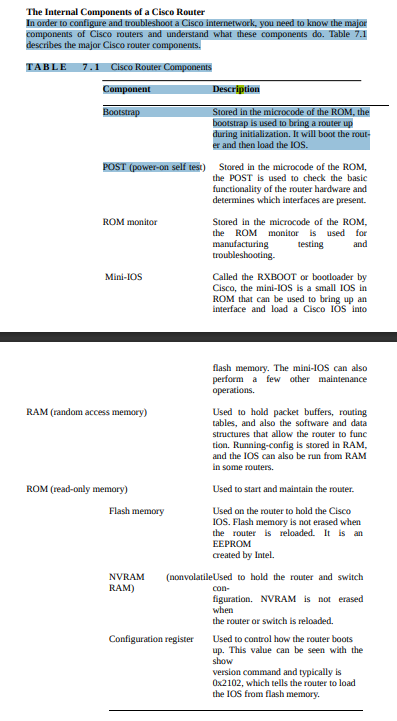
exit

write memory

These commands enable the specified interface, assign it an IP address (192.168.1.1 in this case), activate the interface, and save the configurations. Adjust the IP address and interface details as needed for your network setup.

**10 marks unit 5**

**What are the different component of a Router? Explain each of them.**



**2. In detail, explain the Booting process of a Router**

The booting process of a Cisco router involves several stages, each crucial for initializing the hardware, loading the operating system, and becoming operational. Here's a detailed overview of the router boot process:

1. \*Power-On Self-Test (POST):\*

- When a router is powered on or restarted, it initiates the Power-On Self-Test (POST) sequence.

- During POST, the router checks its hardware components, such as CPU, memory, interfaces, and other vital elements, to ensure they're functioning correctly.

- If any hardware component fails during the POST, the router might not proceed with the boot process and may display error messages indicating the faulty component.

2. \*Bootstrap Program (ROMMON):\*

- After a successful POST, the router's bootstrap program (ROMMON - ROM Monitor) is loaded from the Read-Only Memory (ROM).

- ROMMON helps initialize the router and is responsible for locating the IOS (Internetwork Operating System) software.

3. \*Locating the IOS Image:\*

- The router checks the configuration register to determine where to find the IOS image.

- By default, the configuration register value is 0x2102, which directs the router to load the IOS from Flash memory and the startup configuration from NVRAM.

4. \*Loading the IOS Image:\*

- The router loads the IOS software from the storage device indicated by the configuration register (usually Flash memory).

- If the IOS image is corrupt or missing, the router might fail to boot, displaying an error or prompting the user to recover or reload a valid image.

5. \*Checking Startup Configuration:\*

- Once the IOS is loaded, the router checks the startup-config file in NVRAM.

- If a valid configuration file is found, the router applies the configurations, including IP addresses, interface settings, routing protocols, etc.

6. \*Initialization and Becoming Operational:\*

- After loading the IOS and applying the startup configurations, the router initializes its interfaces and becomes operational.

- It starts performing its routing functions, forwarding packets, and participating in the network as per its configured settings.

7. \*Logging and System Messages:\*

- Throughout the boot process, the router generates log messages that provide details about the boot sequence, hardware, software, and any errors encountered during the process.

- These messages are stored in the router's RAM and can be accessed to troubleshoot booting or operational issues.

**3.** **With an example illustrate Routing between two host using single Router (chatgpt answer)**

**Let's assume a simple network with the following details:**

- \*Host A:\* IP address 192.168.1.2/24

- \*Host B:\* IP address 192.168.2.2/24

- \*Router:\* Two interfaces connected to Host A and Host B with IP addresses 192.168.1.1/24 and 192.168.2.1/24 respectively.

### Configuring the Router:

1. \*Access the Router's CLI:\*

Access the router's command-line interface.

2. \*Configure Interfaces:\*

Configure the router interfaces connected to Host A and Host B.

markdown

enable

configure terminal

interface GigabitEthernet0/0

ip address 192.168.1.1 255.255.255.0

no shutdown

exit

interface GigabitEthernet0/1

ip address 192.168.2.1 255.255.255.0

no shutdown

exit

### Configuring Hosts:

1. \*Host A:\*

Configure the IP address and default gateway on Host A.

IP: 192.168.1.2

Subnet Mask: 255.255.255.0

Gateway: 192.168.1.1 (Router interface connected to Host A)

2. \*Host B:\*

Configure the IP address and default gateway on Host B.

IP: 192.168.2.2

Subnet Mask: 255.255.255.0

Gateway: 192.168.2.1 (Router interface connected to Host B)

### Routing Configuration on the Router:

1. \*Enable IP Routing:\*

By default, most routers have IP routing enabled. Ensure it's enabled using the following command (though it might not be necessary):

markdown

ip routing

2. \*Verification:\*

Test connectivity between Host A and Host B by pinging from one host to the other. For instance, from Host A:

markdown

ping 192.168.2.2

### Explanation:

- The router acts as the gateway for both Host A and Host B.

- Host A and Host B are on different subnets. The router's interfaces with IP addresses in those subnets allow traffic to be routed between them.

- Configuring the correct IP addresses on the hosts and setting the router's interfaces properly allows them to communicate through the router.

This setup enables the router to forward packets between the two hosts, facilitating communication across different networks.

**4. With an example illustrate Routing between two host using single multiple Router**

Let's consider a scenario where two hosts, Host A and Host B, are in different networks and need to communicate through multiple routers. Host A is on network 192.168.1.0/24, and Host B is on network 192.168.2.0/24. Three routers, R1, R2, and R3, will facilitate the communication between these hosts.

The topology is as follows:

Host A --- R1 --- R2 --- R3 --- Host B

### Step-by-Step Routing:

1. \*Configuring IP addresses on interfaces:\*

- Host A: IP 192.168.1.10 (subnet 255.255.255.0)

- Host B: IP 192.168.2.10 (subnet 255.255.255.0)

- R1: Interface 1 - IP 192.168.1.1/24; Interface 2 - IP 10.0.0.1/30

- R2: Interface 1 - IP 10.0.0.2/30; Interface 2 - IP 10.0.0.5/30

- R3: Interface 1 - IP 10.0.0.6/30; Interface 2 - IP 192.168.2.1/24

2. \*Setting up routing tables:\*

R1:

ip route 192.168.2.0 255.255.255.0 10.0.0.2

R2:

ip route 192.168.1.0 255.255.255.0 10.0.0.1

ip route 192.168.2.0 255.255.255.0 10.0.0.6

R3:

ip route 192.168.1.0 255.255.255.0 10.0.0.5

3. \*Host Configuration:\*

- Host A's default gateway: 192.168.1.1 (R1)

- Host B's default gateway: 192.168.2.1 (R3)

### Communication Flow:

1. \*Host A to Host B:\*

- Host A sends data to Host B.

- Host A checks its routing table, finds the destination (Host B) is on a different network, and forwards the data to its default gateway (R1).

2. \*R1 to R2:\*

- R1 receives the data from Host A and checks its routing table.

- R1 determines the next hop to reach Host B (192.168.2.0/24) is through R2 (10.0.0.2).

- R1 forwards the data packet to R2.

3. \*R2 to R3:\*

- R2 receives the data from R1 and looks at its routing table.

- R2 finds two routes to the destination network (192.168.2.0/24) - through R1 and directly to R3.

- As the more specific route to R3 exists, R2 forwards the packet directly to R3 (10.0.0.6).

4. \*R3 to Host B:\*

- R3 receives the packet from R2 and examines its routing table.

- R3 identifies that the destination IP belongs to its connected network.

- R3 delivers the packet directly to Host B.

This illustrates the communication flow between two hosts across multiple routers using appropriate routing configurations and table entries. Adjustments in routing tables and IP addresses may be required based on the specific network setup.

**3. With an example illustrate Routing between two host using single Router (sir lab notes)**

Configuring R1:

R1(config)#interface fa0/0

R1(config-if)#ip address 10.0.0.10 255.0.0.0

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#interface fa0/1

R1(config-if)#ip address 20.0.0.10 255.0.0.0

R1(config-if)#no shutdown

R1(config-if)#exit

Connect Node A with 10.0.0.1 & mask 255.0.0.0 to FA0/0

Connect Node B with 20.0.0.1 & mask 255.0.0.0 to FA0/1

/\* verify the roputing table

Router# Sh ip route

Check the connectivity between Node A & Node B for a success

**4. With an example illustrate Routing between two host using multiple Router (sir lab notes)**

Configuring R1:

R1(config)#interface fa0/0

R1(config-if)#ip address 10.0.0.10 255.0.0.0

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#interface s0/0

R1(config-if)#ip address 30.0.0.10 255.0.0.0

R1(config-if)#clock rate 64000

R1(config-if)#no shutdown

R1(config-if)#exit

Connect Node A with 10.0.0.1 & mask 255.0.0.0 to FA0/0

/\* verify the roputing table

R1# sh ip route

Note: Interface Serial0/0 of Router R1 is a DCE end, so clock rate must be given to this.

Configuring R2:

R2(config)#interface fa0/0

R2(config-if)#ip address 20.0.0.10 255.0.0.0

R2(config-if)#no shutdown

R2(config-if)#exit

R2(config)#interface s0/0

R2(config-if)#ip address 30.0.0.20 255.0.0.0

R2(config-if)#no shutdown

R2(config-if)#exit

Connect Node B with 20.0.0.1 & mask 255.0.0.0 to FA0/0

Check the connectivity between Node A & Node B

/\*Adding static route on R1 for network 20.0.0.0

R1(config)#ip route 20.0.0.0 255.0.0.0 30.0.0.20

/\* verify the roputing table

R1# sh ip route

/\*Adding static route on R2 for network 10.0.0.0

R2(config)#ip route 10.0.0.0 255.0.0.0 30.0.0.10

/\* verify the roputing table

R2# sh ip route

Check the connectivity between Node A & Node B for a success