DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

CATEGORY - 1 UNIVERSITY BY UGC

Accredited "A++" by NAAC I Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119

Cisco AICTE Virtual Internship Program 2024

A Cisco AICTE Virtual Internship project report on cyber security submitted in partial fulfillment of the requirements for the AICTE-CISCO virtual Internship in Networking Program 2024

Submitted By : Moturi Satish

AICTE Internship Student Registration ID) : STU64e8c8524da131692977234

Student ID (Enrolment number) : 42110825

Email : moturisatish382@gmail.com

Contact Info : 8555871939

<u>Assessment Criteria</u>: Upon completion of the above projects, students are expected to -

- **Explain the distinction between MAC learning and IP networking**
- **Differences between Layer 2 and Layer 3 networks.**
- **Explain the function of subnetting and the workings of the Address Resolution Protocol (ARP).**
- **Explain the role of a default gateway, the process required for packets to move between subnets, and the initial steps for troubleshooting connectivity issues.**
- **Should be able to differentiate between a router and a switch in more practical manner.**
- **Should be able to articulate the purpose of static routing and the method by which network connectivity is established across different networks.**
- **What does Routing Information Base (RIB) mean**
- **One should be able to define what a routing protocol is, the differences between static and dynamic routing protocols, and provide an example of a dynamic routing protocol.**
- **Explain how the ping function works? Additionally, elaborate on the significance of the ICMP protocol? Develop a ping program using C/C++. Implementing a basic ping program involves using sockets to send ICMP (Internet Control Message Protocol) echo requests and handle ICMP echo replies.**
- **Explain the basic flow of packet from PC to a web server located outside the campus.**

By completing the "NetPath Illuminator" project, students will gain a practical understanding of networking concepts, apply their theoretical knowledge to a real-world application.

(Note : Assignment solutions are given below)

1.Explain the distinction between MAC learning and IP networking

S.NO	MAC Address	IP Address
1.	MAC Address stands for Media Access Control Address.	IP Address stands for Internet Protocol Address.
2.	MAC Address is a six byte hexadecimal address.	IP Address is either a four-byte (IPv4) or a sixteen-byte (IPv6) address.
3.	A device attached with MAC Address can retrieve by ARP protocol.	A device attached with IP Address can retrieve by RARP protocol.
4.	NIC Card's Manufacturer provides the MAC Address.	Internet Service Provider provides IP Address.
5.	MAC Address is used to ensure the physical address of a computer.	IP Address is the logical address of the computer.
6.	MAC Address operates in the data link layer.	IP Address operates in the network layer.
7.	MAC Address helps in simply identifying the device.	IP Address identifies the connection of the device on the network.
8.	MAC Address of computer cannot be changed with time and environment.	IP Address modifies with the time and environment.
9.	MAC Addresses can't be found easily by a third party.	IP Addresses can be found by a third party.
10.	It is a 48-bit address that contains 6 groups of 2 hexadecimal digits, separated by either hyphens (-) or colons(.). Example: 00:FF:FF:AB:BB:AA or 00-FF-FF-AB-BB-AA	IPv4 uses 32-bit addresses in dotted notations, whereas IPv6 uses 128-bit addresses in hexadecimal notations. Example: IPv4 192.168.1.1 IPv6 FFFF:F200:3204:0B00

S.NO	MAC Address	IP Address
11.	No classes are used for MAC addressing.	IPv4 uses A, B, C, D, and E classes for IP addressing.
12.	MAC Address sharing is not allowed.	In IP address multiple client devices can share the IP address.
13.	MAC address help to solve IP address issue.	IP addresses never able to solve MAC address issues.
14.	MAC addresses can be used for broadcasting.	The IP address can be used for broadcasting or multicasting.
15.	MAC address is hardware oriented.	IP address is software oriented.
16.	While communication, Switch needs MAC address to forward data.	While communication, Router need IP address to forward da

2.Differences between Layer 2 and Layer 3 networks.

Layer 2, known as the Data Link Layer, provides node-to-node data transfer with MAC address identification. All nodes on a layer 2 network are visible to one another. Ethernet switches are a common layer 2 example.

Layer 3, known as the Network Layer routes data packets to specific nodes identified by IP addresses. Visibility amongst nodes is point-to-point. As its function suggests, routers are a common layer 3 example.

Both have merits for different applications.

Layer 2 is beneficial for remote support. Enabling both the full functionality of most industrial protocols that sit atop ethernet *and* the ability to search for active IP nodes puts the programmer next to the machine, albeit virtually. There is no functional difference in this example between a layer 2 network and an ethernet cord.

Layer 3 provides a more scalable network infrastructure, like that used in a globally distributed system of machines connected to a central cloud for data analytics. Because data packets are *routed,* layer 3 enables 1-1 NAT and VLAN, among other traditional networking strategies. As distributed networks scale, layer 3 becomes more practical for managing connections.

3.Explain the function of subnetting and the workings of the Address Resolution Protocol (ARP).

An IP address is the *logical* addressing scheme for nodes on a network. IP Addresses exist at the Network layer of the OSI Model and help facilitate the L3 goal of "*end to end*" delivery.

A MAC address is the *physical* addressing scheme for individual NIC cards on each node of a network. MAC addresses exist at the Data Link layer of the OSI Model and help facilitate the L2 goal of "*hop to hop*" delivery.

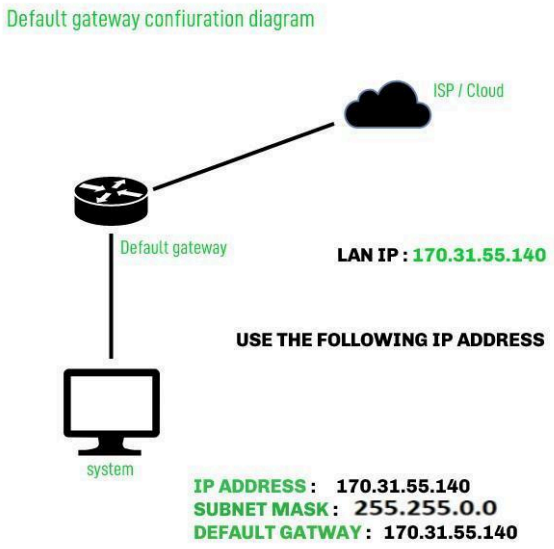
4.Explain the role of a default gateway, the process required for packets to move between subnets, and the initial steps for troubleshooting connectivity issues.

Default Gateway:

- The default gateway is the node that forwards the packet from the source to other networks when there is no routing information about the destination i.e. host (or router) does not know where the destination is present.
- A default gateway is a route to which information is passed when the device does not know where the destination is present.
- It is used when there is no routing information available about the destination.
- It is a node that allows the communication of computers on different networks.
- 'Default' here means the default route which is to be taken when the host does not know where the destination is.
- It is most commonly used for webpage access.
- This is an important part of networking for routing the data and finding the corresponding destination which is in another network.

When Default Gateway is Used:

When the source wants to reach a destination which is outside its network then, the source uses the default gateway to forward the data and locate the destination's network so that data should reach its intended destination. The default gateways are used when the host doesn't know about the destination's network i.e. the network in which the destination is present or when the route information is not available for any destination then it goes to the default gateway so that it can identify in which network the destination is and can forward the data through that route. The default gateway is an important device for the data forwarding and routing of the data on the other network. It helps in the communication of one network computer with the other network computer.



How to Find Your Default Gateway:

1. In Windows:

1. In Windows, we can get our default gateway with Command Prompt.

2. To access the command prompt click on start and search for "**CMD**" and open.
3. In the command prompt window type "**ipconfig**" command and press enter and you can see the Default Gateway in the information generated by the command with the machine's IP address listed.

2. In Linux:

- Open the terminal and use the command "**ip route | grep default** ". It will locate the default gateway.

5.Should be able to differentiate between a router and a switch in more practical manner

Router:

The router is a networking device that works at the network layer i.e., a third layer of the ISO-OSI model, and is the multiport device. It establishes a simple connection between the networks to provide the data flow between the networks. Router transfers data in the form of packets is used in LAN as well as MAN.

It works on network layer 3 and is used in LANs, MANs, and WANs. It stores IP addresses and maintains addresses on its own.

Switch:

It is a point-to-point communication device. Basically, it is a kind of bridge that provides better connections. It is a kind of device that sets up and stops the connections according to the requirements needed at that time. It comes up with many features such as flooding, filtering, and frame transmission.

Router	**Switch**
The main objective of router is to connect various networks simultaneously.	While the main objective of switch is to connect various devices simultaneously.
It works in network layer	While it works in data link layer
Router is used by LAN as well as MAN	While switch is used by only LAN.
Through the router, data is sent in the form of packets.	While through switch data is sent in the form of frame.
There is less collision taking place in the router.	While there is no collision taking place in full duplex switch.
Router is compatible with NAT	While it is not compatible with NAT.

Router	Switch
Router is a relatively much more expensive device than switch.	Switch is an expensive device than hub. but cheaper than router.
maximum speed for wireless is 1-10 Mbps and maximum speed for wired connections is 100 Mbps.	Maximum speed is 10Mbps to 100Mbps.
Router needs at least two networks to connect.	Switch needs at least single network is to connect.
The types of routing are Adaptive and Non-Adaptive .	The types of switching are: Circuit, Packets and Message Switching.

6.Should be able to articulate the purpose of static routing and the method by which network connectivity is established across different networks.

Static routing is a routing protocol that helps to keep your network organized and to optimize routing performance. It enables the router to assign a specific path to each network segment and to keep track of network changes. This helps to improve network stability and continuity. This adds security because a single administrator can only authorize routing to particular networks.

Steps to Configure and Verify Two Router Connections in Cisco Packet Tracer :

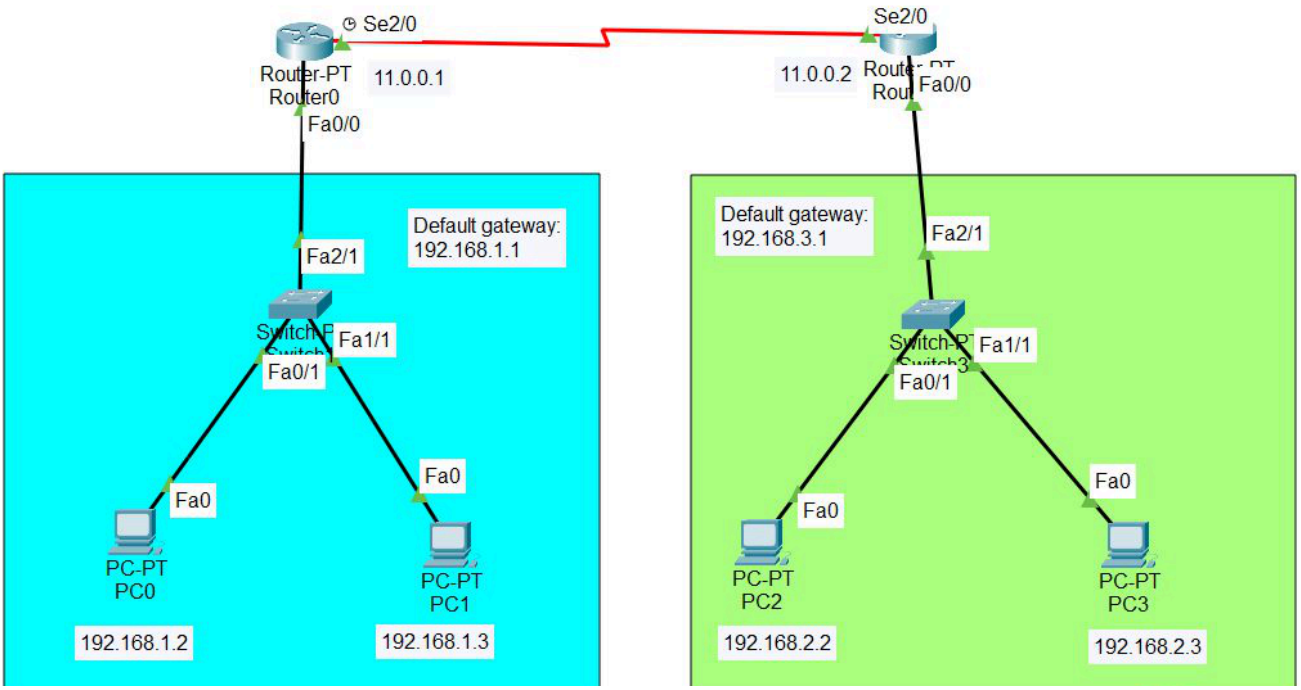
Step 1: First, open the cisco packet tracer desktop and select the devices given below:

S.NO	Device	Model Name	Qty.
1.	PC	PC	4
2.	Switch	PT-Switch	2
3.	Router	PT-Router	2

IP Addressing Table For PCs:

S.NO	Device	IPv4 Address	Subnet Mask	Default Gateway
1.	pc0	192.168.1.2	255.255.255.0	192.168.1.1
2.	pc1	192.168.1.3	255.255.255.0	192.168.1.1
3.	pc2	192.168.2.2	255.255.255.0	192.168.2.1
4.	pc3	192.168.2.3	255.255.255.0	192.168.2.1

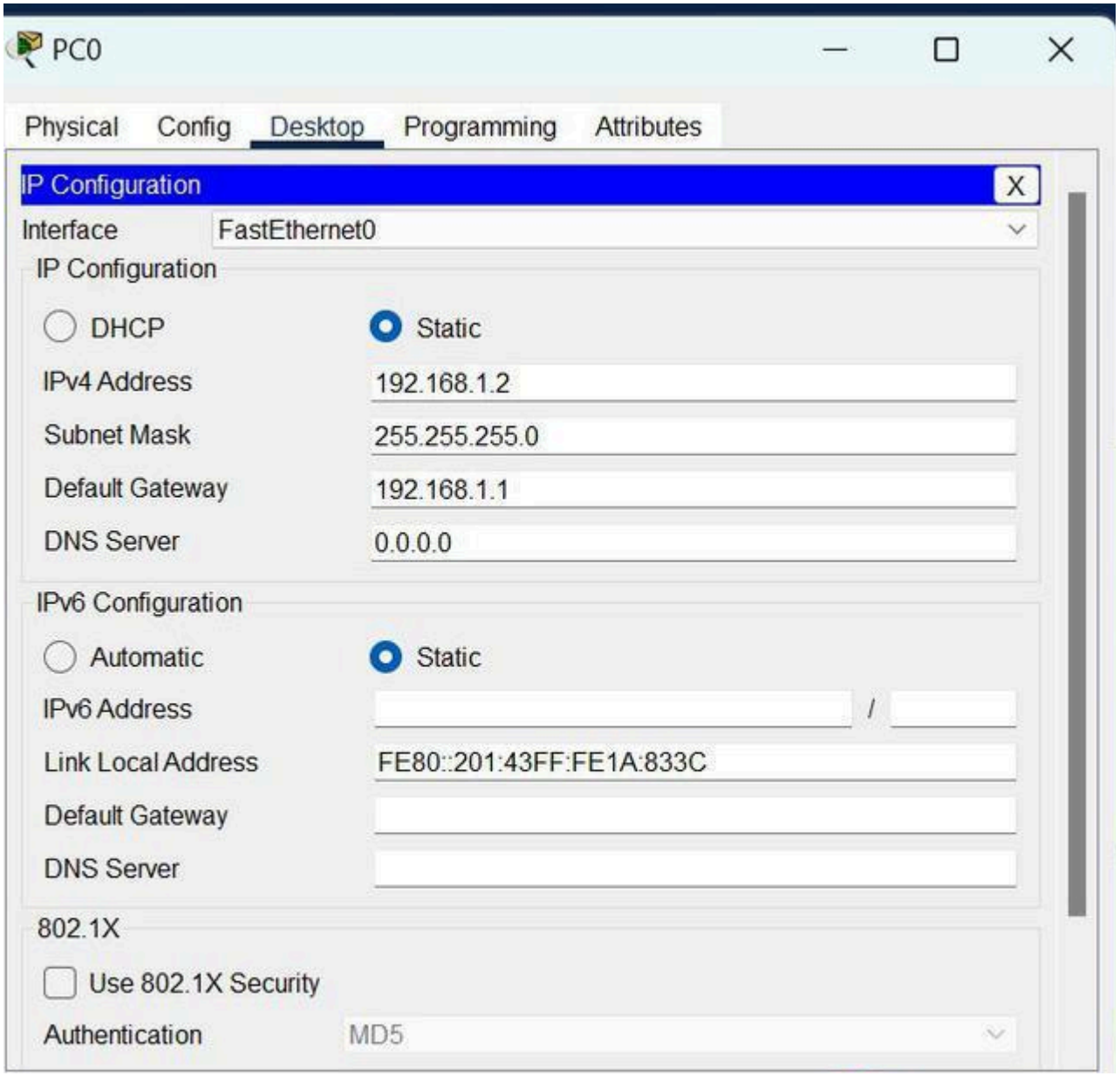
- Then, create a network topology as shown below the image.
- Use an Automatic connecting cable to connect the devices with others.



Correction: Default Gateway in 2nd network is 192.168.2.1

Step 2: Configure the PCs (hosts) with IPv4 address and Subnet Mask according to the IP addressing table given above.

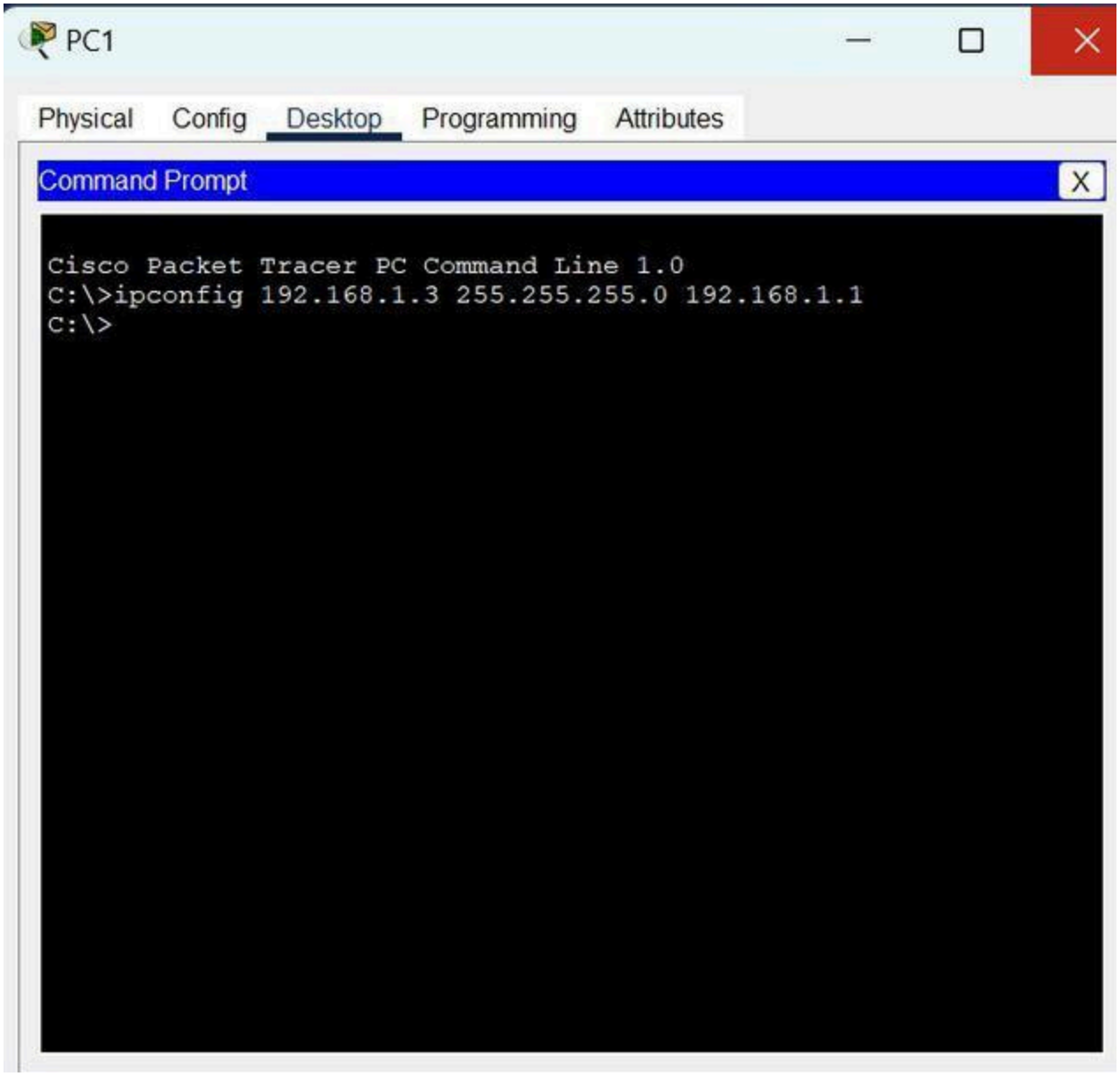
- To assign an IP address in PC0, click on PC0.
- Then, go to desktop and then IP configuration and there you will IPv4 configuration.
- Fill IPv4 address and subnet mask.



Step 3: Assigning IP address using the ipconfig command.

- We can also assign an IP address with the help of a command.
- Go to the command terminal of the PC.
- Then, type ipconfig <IPv4 address><subnet mask><default gateway>(if needed)

Example: ipconfig 192.168.1.3 255.255.255.0 192.168.1.1



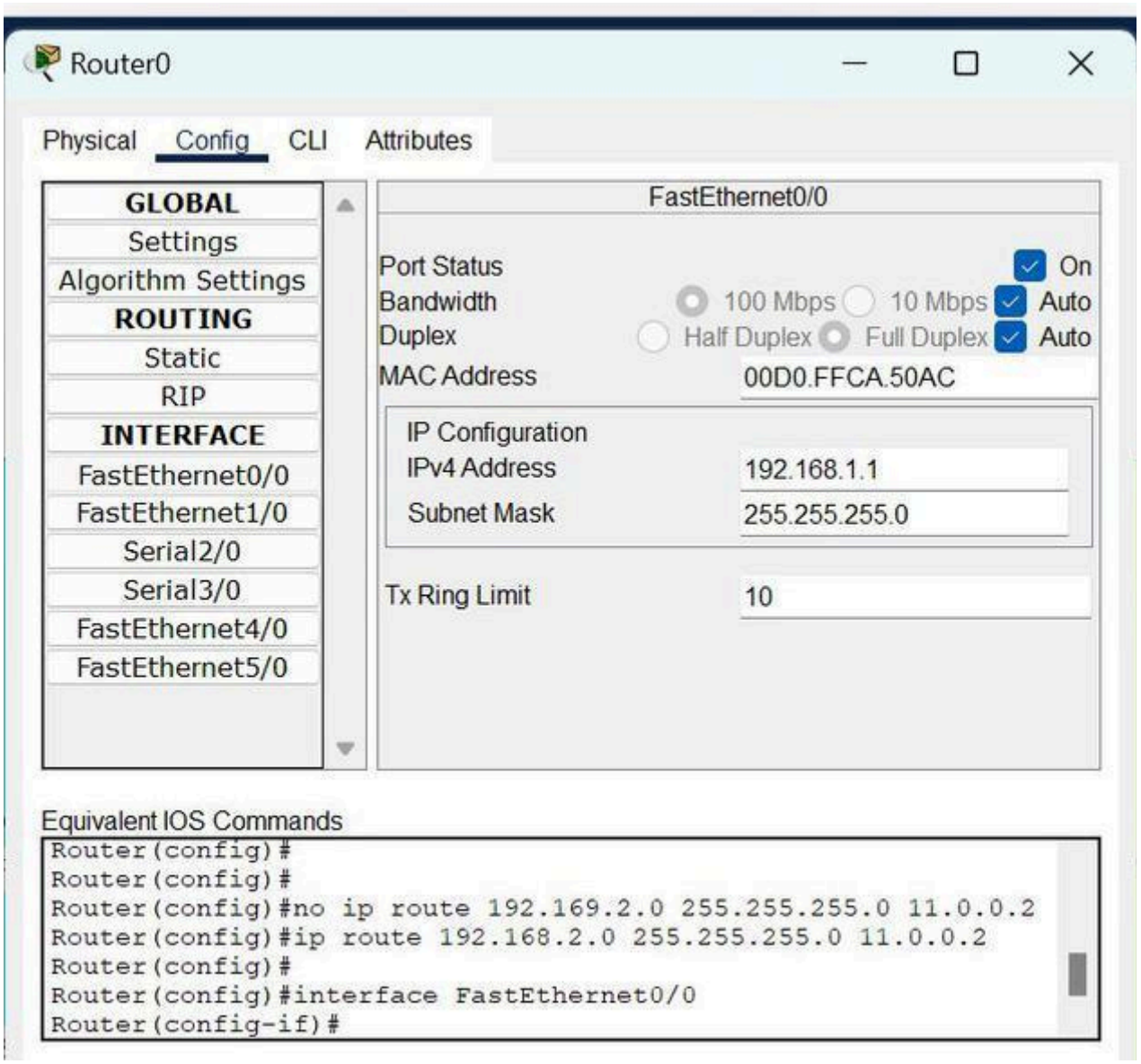
- Repeat the same procedure with other PCs to configure them thoroughly.

Step 4: Configure router with IP address and subnet mask.

S.NO	Device	Interface	IPv4 Addressing	Subnet Mask
1.	router0	FastEthernet0/0	192.168.1.1	255.255.255.0
		Serial2/0	11.0.0.1	255.255.255.0
2.	router1	FastEthernet0/0	192.168.2.1	255.255.255.0

S.NO	Device	Interface	IPv4 Addressing	Subnet Mask
		Serial2/0	11.0.0.2	255.255.255.0

- To assign an IP address in router0, click on router0.
- Then, go to config and then Interfaces.
- Then, configure the IP address in FastEthernet and serial ports according to IP addressing Table.
- Fill IPv4 address and subnet mask.



- Repeat the same procedure with other routers to configure them thoroughly.

Step 5: After configuring all of the devices we need to assign the routes to the routers.

To assign static routes to the particular router:

- First, click on router0 then Go to CLI.
- Then type the commands and IP information given below.

CLI command : ip route <network id> <subnet mask><next hop>

Static Routes for Router0 are given below:

Router(config)#ip route 192.168.2.0 255.255.255.0 11.0.0.2

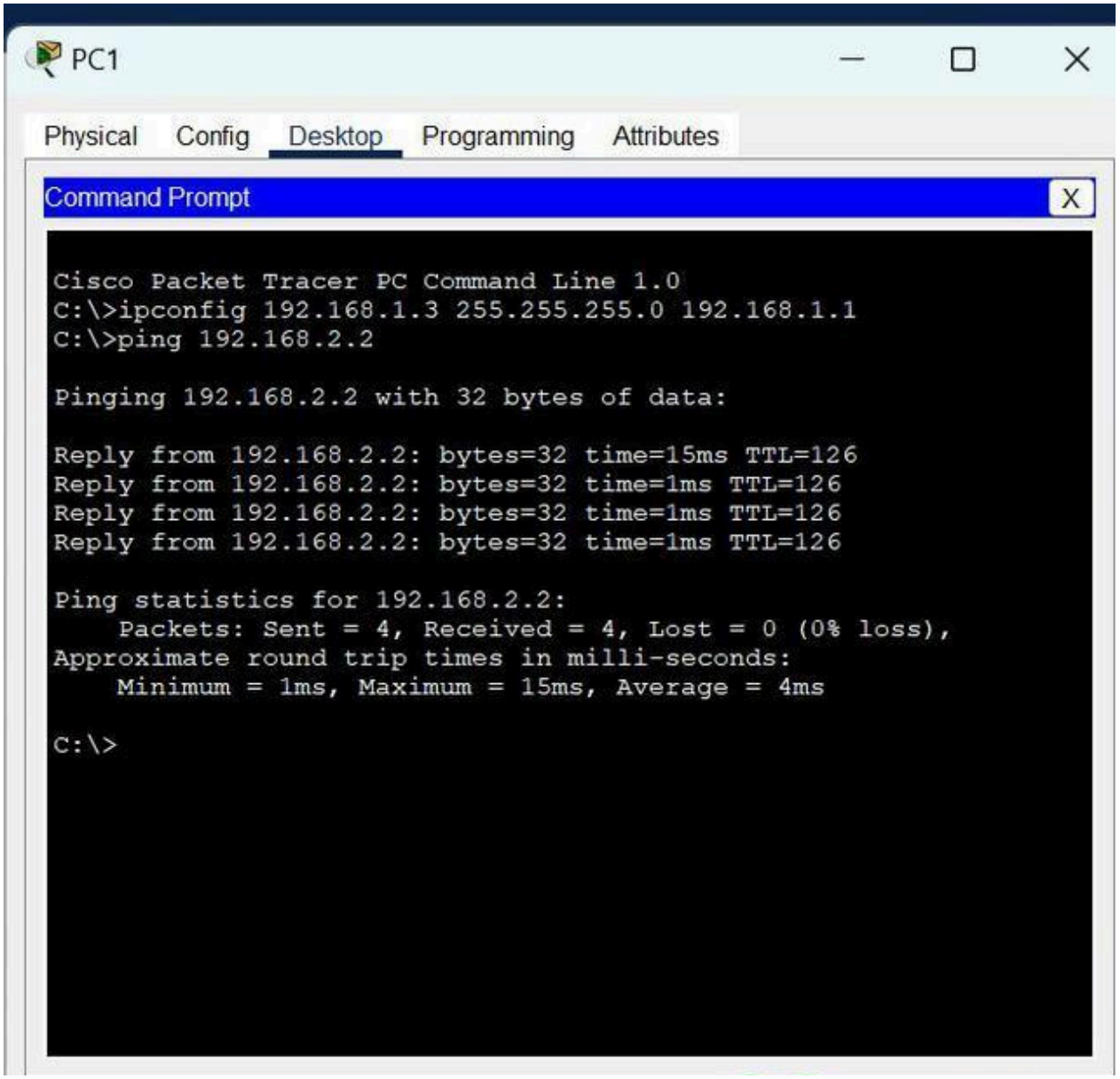
Static Routes for Router1 are given below:

Router(config)#ip route 192.168.1.0 255.255.255.0 11.0.0.1

Step 6: Verifying the network by pinging the IP address of any PC. We will use the ping command to do so.

- First, click on PC1 then Go to the command prompt
- Then type ping <IP address of targeted node>
- As we can see in the below image we are getting replies which means the connection is working very fine

Example : ping 192.168.2.2



7.What does Routing Information Base (RIB) mean

The entire set of BGP routes learned and advertised by a BGP router make up its BGP Routing Information Base (RIB). Conceptually the BGP RIB can be divided into 3 parts:

- RIB-IN
- LOC-RIB
- RIB-OUT

The RIB-IN (or Adj-RIBs-In as defined in RFC 4271) holds the BGP routes that were received from peers and that the router decided to keep (store in its memory).

The LOC-RIB contains modified versions of the BGP routes in the RIB-IN. The path attributes of a RIB-IN route can be modified using BGP import policies. All of the LOC-RIB routes for the same NLRI are compared in a procedure called the BGP decision process that results in the selection of the best path for each NLRI. The best paths in the LOC-RIB are the ones that are actually 'usable' by the local router for forwarding, filtering, auto-discovery, and so on.

The RIB-OUT (or Adj-RIBs-Out as defined in RFC 4271) holds the BGP routes that were advertised to peers. Normally a BGP route is not advertised to a peer (in the RIB-OUT) unless it is 'used' locally but there are exceptions. BGP export policies modify the path attributes of a LOC-RIB route to create the path attributes of the RIB-OUT route. A particular LOC-RIB route can be advertised with different path attribute values to different peers so there can exist a 1:N relationship between LOC-RIB and RIB-OUT routes.

8.One should be able to define what a routing protocol is, the differences between static and dynamic routing protocols, and provide an example of a dynamic routing protocol.

Features	Static Routing	Dynamic Routing
Definition	It happens when a router utilizes a manually specified routing entry rather than information from dynamic routing traffic.	It is a method in which a router may transmit data via a different route or to a specific destination based on the current state of the network's communication circuits.
Configuration Technique	The routing tables are manually updated in static routing.	The tables are automatically updated in dynamic routing.
Routes	Routes are specified by the administrative.	The routes are updated according to the modifications in the network.
Routing Algorithms	It doesn't utilize any complicated routing algorithms.	It utilizes complicated routing algorithms.
Link Affect	When a link fails in static routing, it interrupts the other routing path.	The link failure doesn't affect rerouting in dynamic routing.
Bandwidth	It needs less bandwidth.	It needs more bandwidth.
Security	It offers high security.	It offers less security.
Network Infrastructure	Its network infrastructure is minimal.	Its network infrastructure is large.
Routing Protocols	It doesn't utilize any protocol.	It employs protocols such as eigrp, arp, and others to calculate the routing process.
Additional Resources	It doesn't need any extra resources.	It needs extra resources to hold the information.
Implementation	It is implemented in small networks.	It is implemented in large networks.
Routing table building	Routing locations are hand-typed in static routing.	In dynamic routing, locations are dynamically filled in the table.

9.Explain how the ping function works? Additionally, elaborate on the significance of the ICMP protocol? Develop a ping program using C/C++. Implementing a basic ping program involves using sockets to send ICMP (Internet Control Message Protocol) echo requests and handle ICMP echo replies.

Ping is a necessity for debugging the Internet. Ping is a basic Internet tool that allows a user to verify that a particular IP address exists and can accept requests., with other facilities.

Ping sends out ICMP packets by opening a RAW Socket, which is separate from TCP and UDP. Since IP does not have any inbuilt mechanism for sending error and control messages. It depends on Internet control message protocol (ICMP) to provide error control. It is used for reporting errors and management queries.

```
// C program to Implement Ping


#include& lt; stdio.h & gt;
#include& lt; sys / types.h & gt;
#include& lt; sys / socket.h & gt;
#include& lt; netinet / in.h & gt;
#include& lt; arpa / inet.h & gt;
#include& lt; netdb.h & gt;
#include& lt; unistd.h & gt;
#include& lt; string.h & gt;
#include& lt; stdlib.h & gt;
#include& lt; netinet / ip_icmp.h & gt;
#include& lt; time.h & gt;
#include& lt; fcntl.h & gt;
#include& lt; signal.h & gt;
#include& lt; time.h & gt;
#define PING_PKT_S 64
#define PORT_NO 0
#define PING_SLEEP_RATE 1000000 x
#define RECV_TIMEOUT 1
int pingloop = 1;
struct ping_pkt {
        struct icmphdr hdr;
        char msg[PING_PKT_S - sizeof(struct icmphdr)];
};
```

```
unsigned short checksum(void* b, int len)
{
	unsigned short* buf = b;
	unsigned int sum = 0;
	unsigned short result;

	for (sum = 0; len & gt; 1; len -= 2)
		sum += *buf++;
	if (len == 1)
		sum += *(unsigned char*)buf;
	sum = (sum & gt; > 16) + (sum & amp; 0xFFFF);
	sum += (sum & gt; > 16);
	result = ~sum;
	return result;
}

void intHandler(int dummy) { pingloop = 0; }

char* dns_lookup(char* addr_host,
				struct sockaddr_in* addr_con)
{
	printf("\nResolving DNS..\n & quot;);
	struct hostent* host_entity;
	char* ip = (char*)malloc(NI_MAXHOST * sizeof(char));
	int i;

	if ((host_entity = gethostbyname(addr_host)) == NULL) {
		// No ip found for hostname
		return NULL;
	}

	strcpy(ip,
		inet_ntoa(*(struct in_addr*)host_entity - >
```

```
                                        h_addr));

        (*addr_con).sin_family = host_entity - >
        h_addrtype;
        (*addr_con).sin_port = htons(PORT_NO);
        (*addr_con).sin_addr.s_addr = *(long*)host_entity - >
        h_addr;

        return ip;
}



char* reverse_dns_lookup(char* ip_addr)
{
        struct sockaddr_in temp_addr;
        socklen_t len;
        char buf[NI_MAXHOST], *ret_buf;

        temp_addr.sin_family = AF_INET;
        temp_addr.sin_addr.s_addr = inet_addr(ip_addr);
        len = sizeof(struct sockaddr_in);

        if (getnameinfo((struct sockaddr*)&
                                        temp_addr, len, buf, sizeof(buf), NULL,
                                        0, NI_NAMEREQD)) {
                printf(
                        "
                        Could not resolve reverse lookup of hostname\n
                        & quot;);
                return NULL;
        }
        ret_buf
                = (char*)malloc((strlen(buf) + 1) * sizeof(char));
        strcpy(ret_buf, buf);
```

```
	return ret_buf;
}

void send_ping(int ping_sockfd,
			struct sockaddr_in* ping_addr,
			char* ping_dom, char* ping_ip,
			char* rev_host)
{
	int ttl_val = 64, msg_count = 0, i, addr_len, flag = 1,
		msg_received_count = 0;



	char rbuffer[128];
	struct ping_pkt* r_pckt;

	struct ping_pkt pckt;
	struct sockaddr_in r_addr;
	struct timespec time_start, time_end, tfs, tfe;
	long double rtt_msec = 0, total_msec = 0;
	struct timeval tv_out;
	tv_out.tv_sec = RECV_TIMEOUT;
	tv_out.tv_usec = 0;

	clock_gettime(CLOCK_MONOTONIC, & tfs);



	if (setsockopt(ping_sockfd, SOL_IP, IP_TTL, &
				ttl_val, sizeof(ttl_val))
		!= 0) {
		printf(
			"\nSetting socket options to TTL failed !\n
				& quot;);
		return;
```

```
}

else {
        printf("\nSocket set to TTL..\n & quot;);
}


setsockopt(ping_sockfd, SOL_SOCKET, SO_RCVTIMEO,
                (const char*)&
                tv_out, sizeof tv_out);


while (pingloop) {
        flag = 1;
        bzero(& pckt, sizeof(pckt));

        pckt.hdr.type = ICMP_ECHO;
        pckt.hdr.un.echo.id = getpid();

        for (i = 0; i & lt; sizeof(pckt.msg) - 1; i++)
                pckt.msg[i] = i + '0';

        pckt.msg[i] = 0;
        pckt.hdr.un.echo.sequence = msg_count++;
        pckt.hdr.checksum
                = checksum(& pckt, sizeof(pckt));

        usleep(PING_SLEEP_RATE);


        clock_gettime(CLOCK_MONOTONIC, & time_start);
        if (sendto(ping_sockfd, &
                        pckt, sizeof(pckt), 0,
                        (struct sockaddr*)ping_addr,
```

```
                sizeof(*ping_addr))& lt;
        = 0) {
        printf("\nPacket Sending Failed !\n
                                & quot;);
        flag = 0;
}


addr_len = sizeof(r_addr);

if (recvfrom(ping_sockfd,
                        rbuffer, sizeof(rbuffer), 0,
                        (struct sockaddr*)&
                        r_addr, & addr_len)& lt;
        = 0 & amp; & msg_count & gt; 1) {
        printf("\nPacket receive failed !\n
                                & quot;);
}

else {
        clock_gettime(CLOCK_MONOTONIC, & time_end);

        double timeElapsed
                = ((double)(time_end.tv_nsec
                                - time_start.tv_nsec))
                / 1000000.0 rtt_msec
                = (time_end.tv_sec - time_start.tv_sec)
                        * 1000.0
                + timeElapsed;


        if (flag) {
                if (!(r_pckt->hdr.type == 0 & amp; &
                        r_pckt->hdr.code == 0)) {
```

```
                        printf(" Error..Packet received
                                                with ICMP type
                                                % d code % d\n
                                        & quot;
                                , r_pckt->hdr.type, r_pckt->hdr.code);
                }
                else {
                        printf(" % d bytes from
                                                % s(h
                                                :
                                                % s)(% s) msg_seq
                                        = % d ttl = % d rtt =
                                                % Lf ms.\n & quot;
                                , PING_PKT_S, ping_dom, rev_host,
                                ping_ip, msg_count, ttl_val,
                                rtt_msec);

                        msg_received_count++;
                }
            }
        }
}
clock_gettime(CLOCK_MONOTONIC, & tfe);
double timeElapsed
        = ((double)(tfe.tv_nsec - tfs.tv_nsec)) / 1000000.0;

total_msec = (tfe.tv_sec - tfs.tv_sec) * 1000.0
                + timeElapsed

                printf("\n == = % s ping statistics ==
                                        =\n & quot;
                                , ping_ip);
printf("\n % d packets sent, % d packets received,
                % f percent packet loss.Total time
```

```
                :
                % Lf ms.\n\n & quot;
                , msg_count, msg_received_count,
                ((msg_count - msg_received_count) / msg_count)
                        * 100.0,
                total_msec);
}

int main(int argc, char* argv[])
{
        int sockfd;
        char *ip_addr, *reverse_hostname;
        struct sockaddr_in addr_con;
        int addrlen = sizeof(addr_con);
        char net_buf[NI_MAXHOST];

        if (argc != 2) {
                printf("\nFormat % s & lt;
                        address & gt;\n & quot;, argv[0]);
                return 0;
        }

        ip_addr = dns_lookup(argv[1], & addr_con);
        if (ip_addr == NULL) {
                printf(
                        "\nDNS lookup
                                        failed !Could not resolve hostname !\n
                                & quot;);
                return 0;
        }

        reverse_hostname = reverse_dns_lookup(ip_addr);
        printf("\nTrying to connect to '%s' IP
                :
```

```
                % s\n & quot;, argv[1], ip_addr);
        printf("\nReverse Lookup domain
                :
                % s & quot;, reverse_hostname);

        // socket()
        sockfd = socket(AF_INET, SOCK_RAW, IPPROTO_ICMP);
        if (sockfd & lt; 0) {
                printf(
                        "\nSocket file descriptor not received !!\n
                                & quot;);
                return 0;
        }
        else
                printf("\nSocket file descriptor % d received\n
                                        & quot;
                        , sockfd);

        signal(SIGINT, intHandler); // catching interrupt

        // send pings continuously
        send_ping(sockfd, &
                        addr_con, reverse_hostname, ip_addr, argv[1]);

        return 0;
}
```

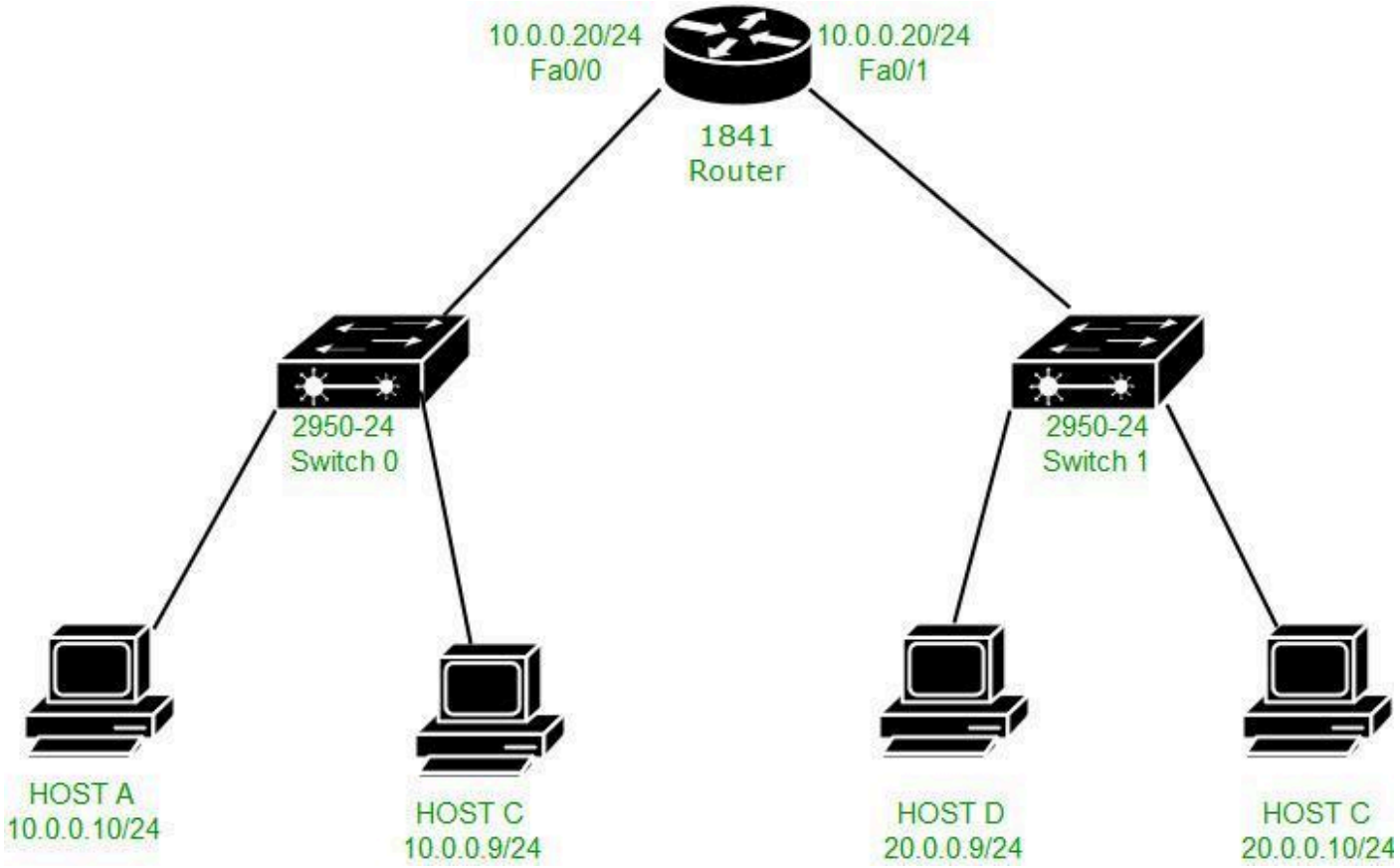
10.Explain the basic flow of packet from PC to a web server located outside the campus.

To deliver the packet to the destination host, the source IP, destination IP, source MAC address and destination MAC address should be known. Some basic rules for the packet flow:

1. If the destination host is present in the same network, then the packet is delivered directly to the destination host.

2. If the destination host is present in a different network then the packet is delivered to the default gateway first which in turn delivers the packet to the destination host.
3. If ARP is not resolved then ARP will be resolved first.
4. MAC address never crosses its broadcast domain.

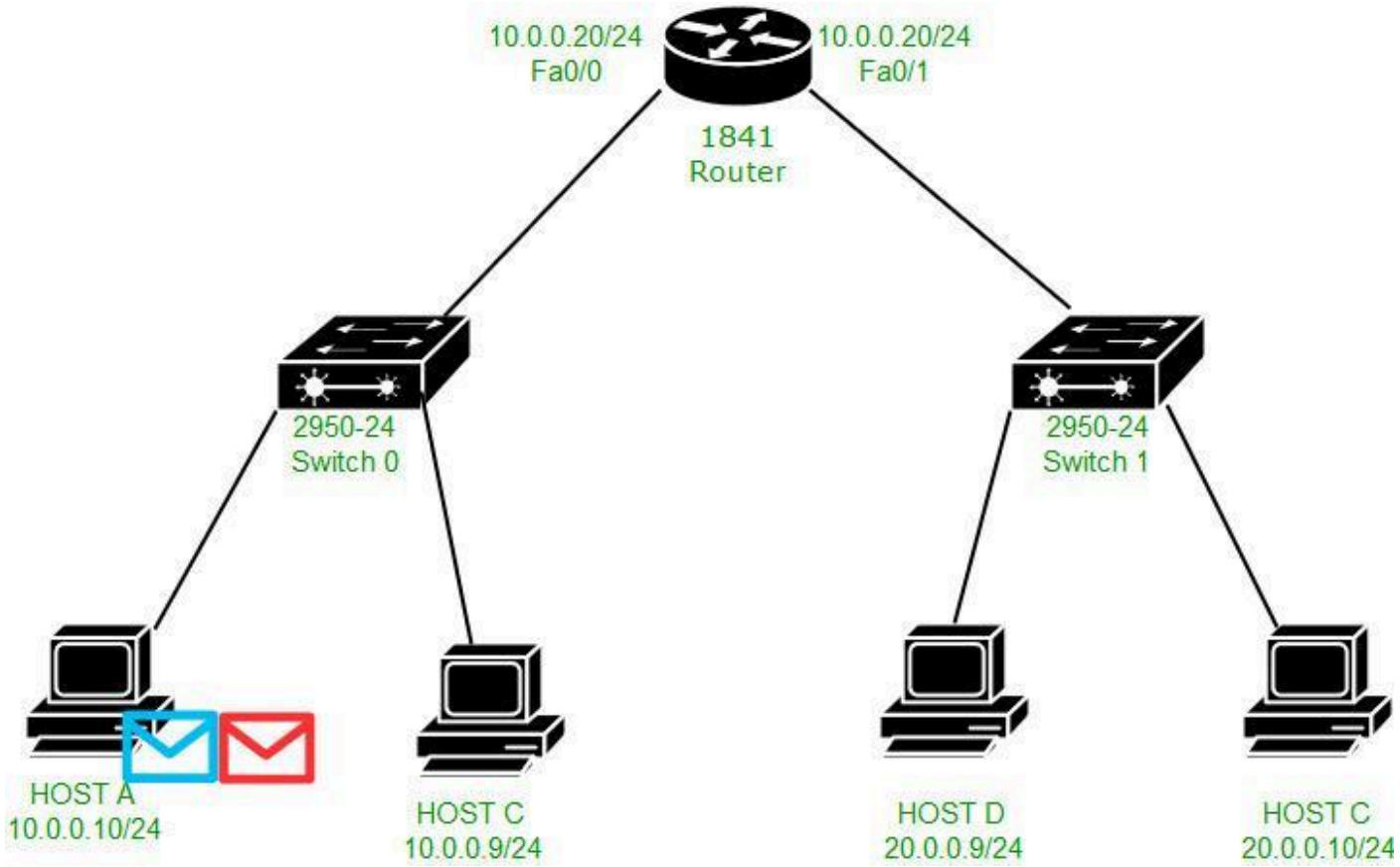
Explanation –



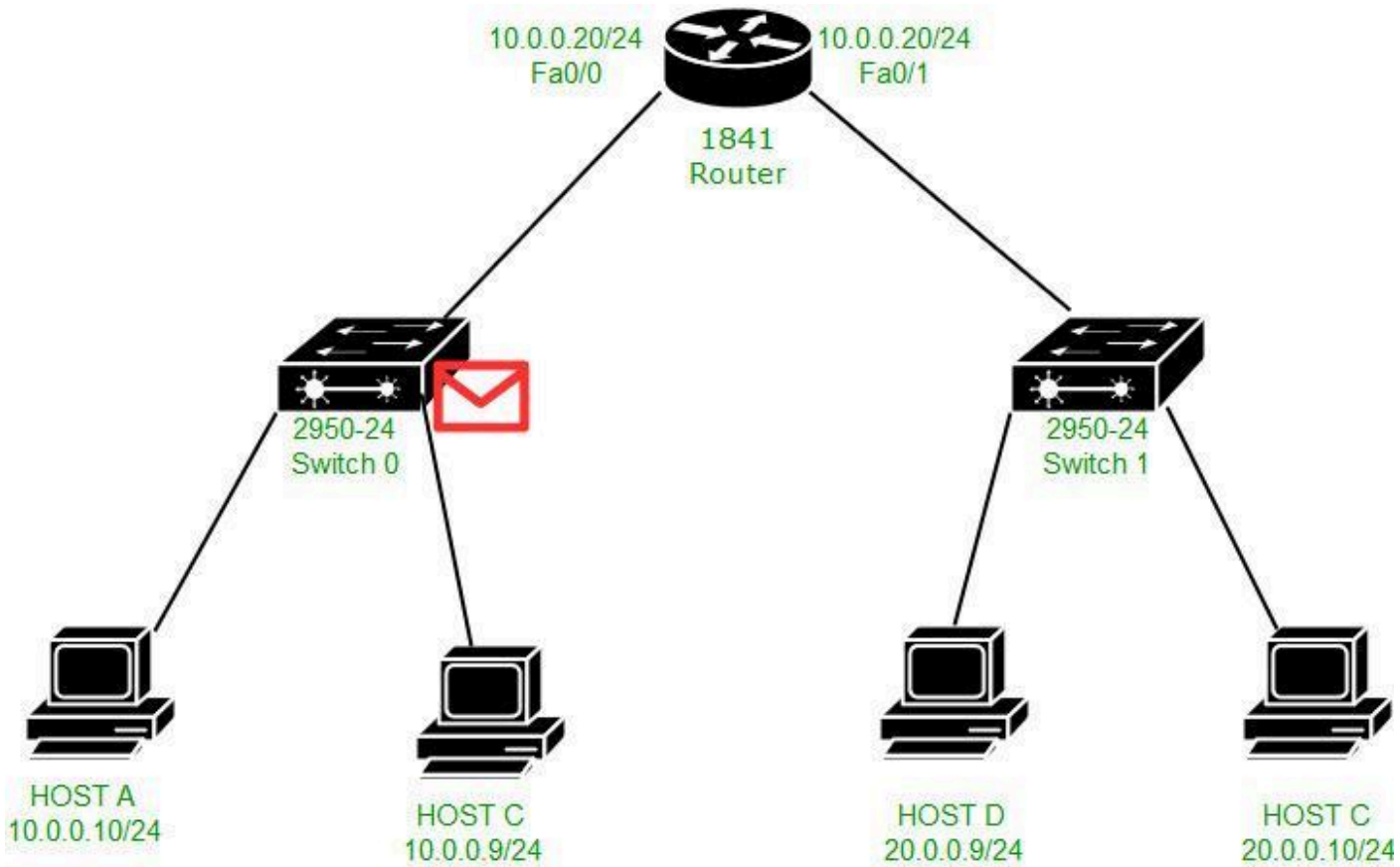
Here is a topology, in which there is host A (IP address – 10.0.0.10 and MAC address – 000D.BD22.7C22), host C (IP address – 10.0.0.9), host B (IP address – 20.0.0.10), host C (IP address-20.0.0.9 and MAC address – 00E0.A3E2.03DC) and the router (IP address – 10.0.0.20 and MAC address – 000B.BE8E.5201 on fa0/0,IP address – 20.0.0.20 and MAC address – 000B.BE8E.5202 on fa0/1).

Now we will try to ping from host A (IP address – 10.0.0.10) to host B (IP address – 20.0.0.10). First, **AND operation** is performed by source host between source IP address, source subnet mask, and destination IP address, source subnet mask to know if the destination is present in same or different network.

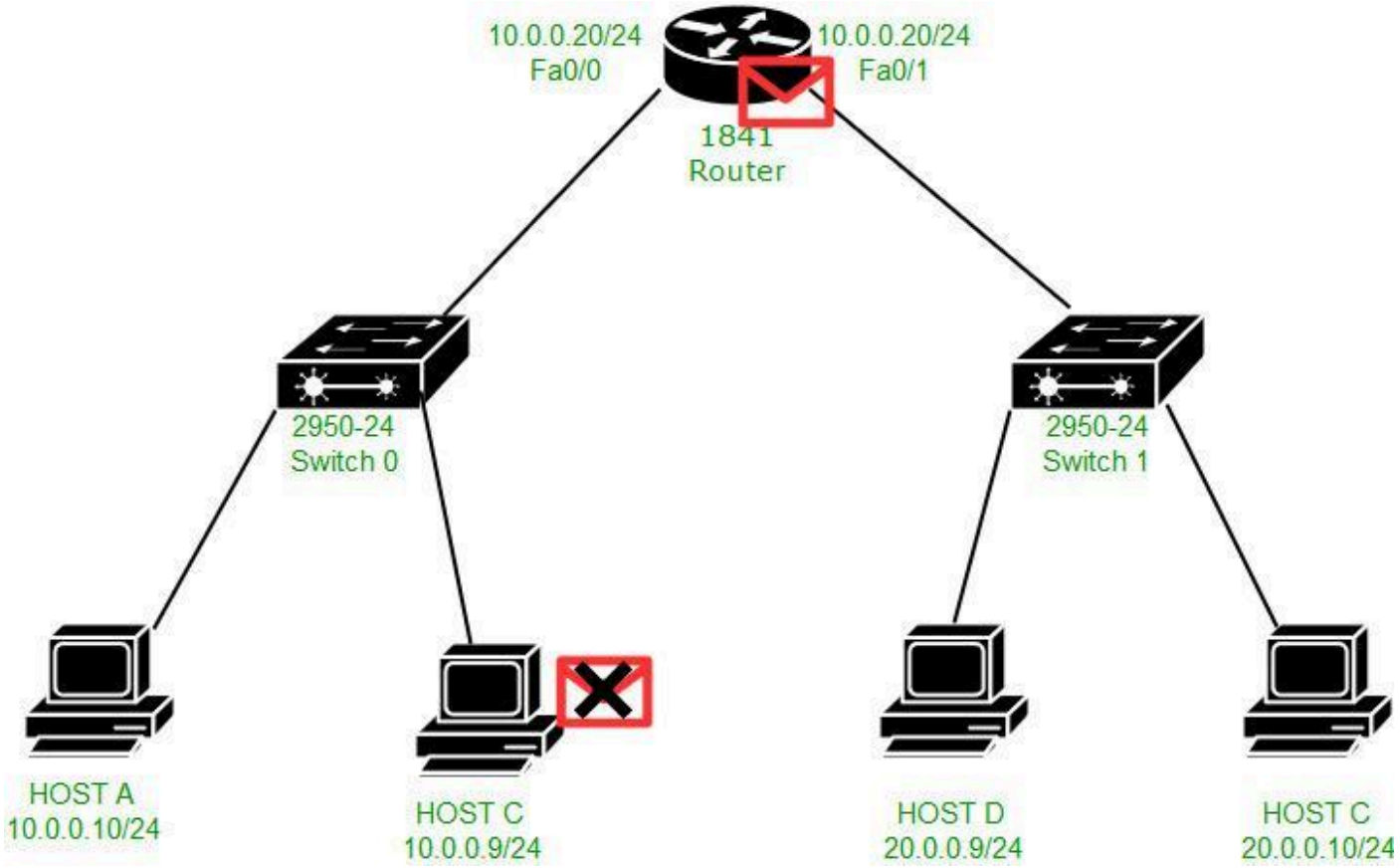
If the result is the same then the destination is in the same network otherwise in a different network. Here, the destination is present in different networks, therefore, the result will be different and the packet will be delivered to a default gateway.



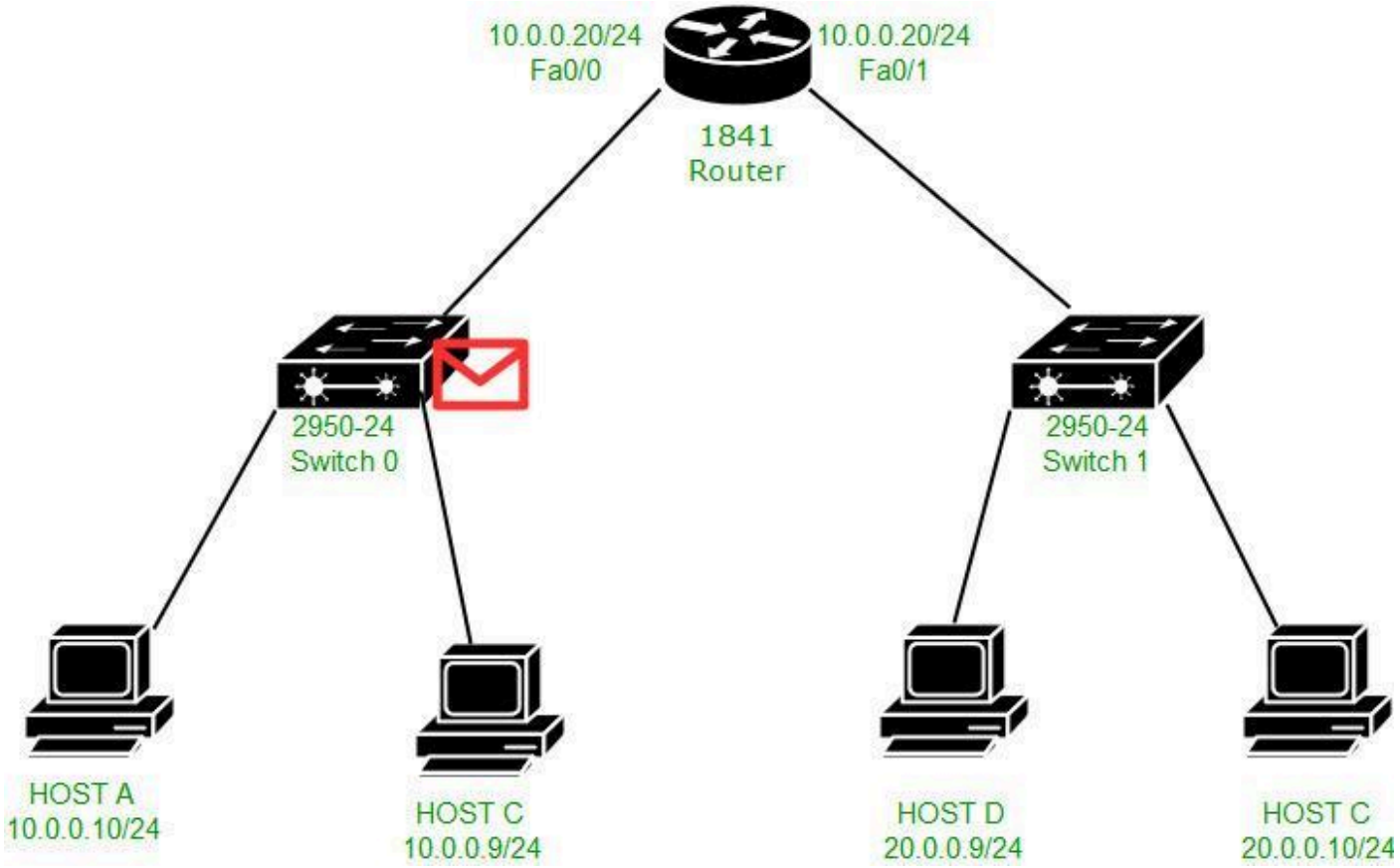
We see that 2 messages are generated ICMP(purple) and ARP(green). ARP has been generated because ARP has not been resolved.

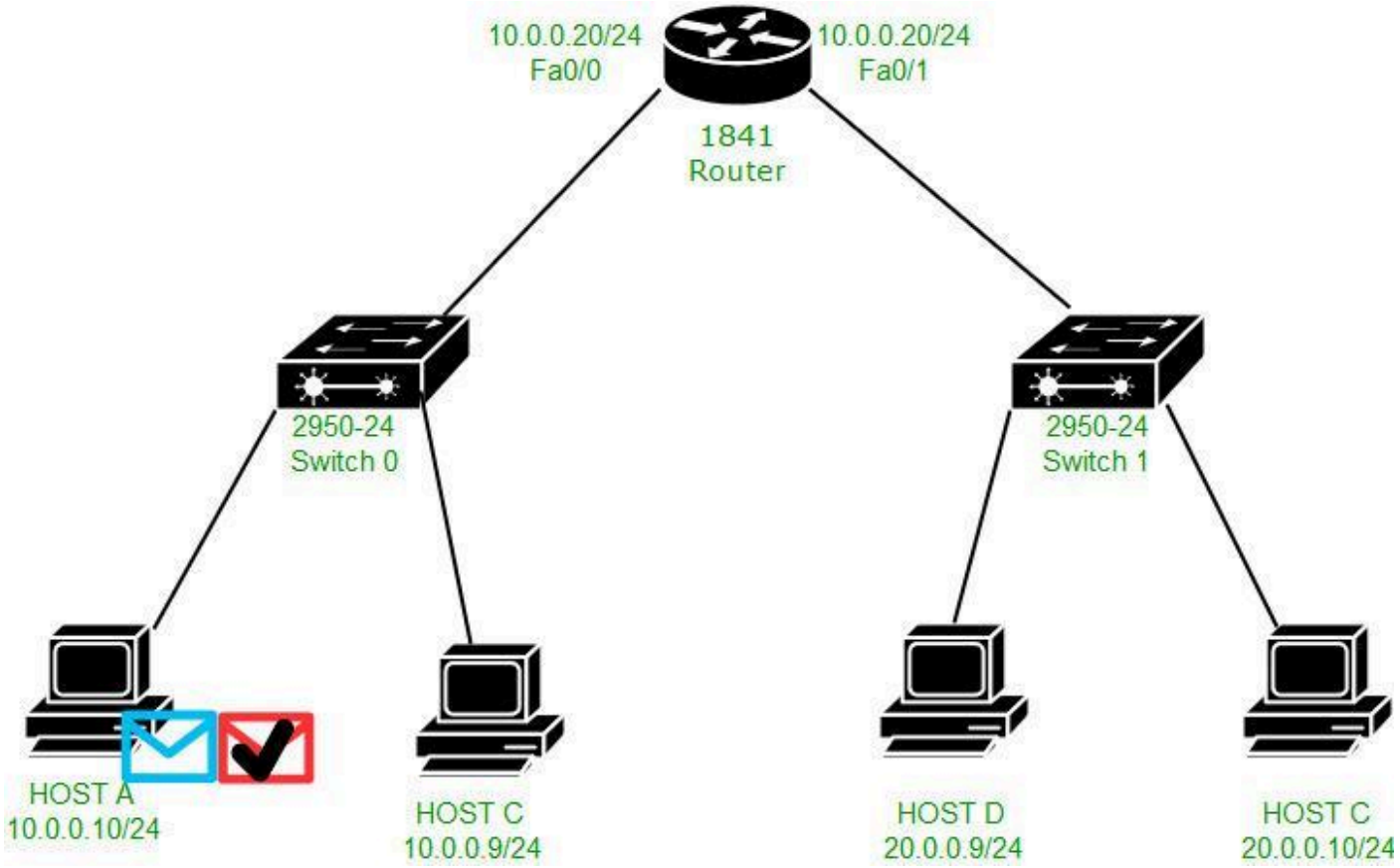


Now as the ARP should be resolved first, therefore the ARP request will be broadcast which is received by switch:

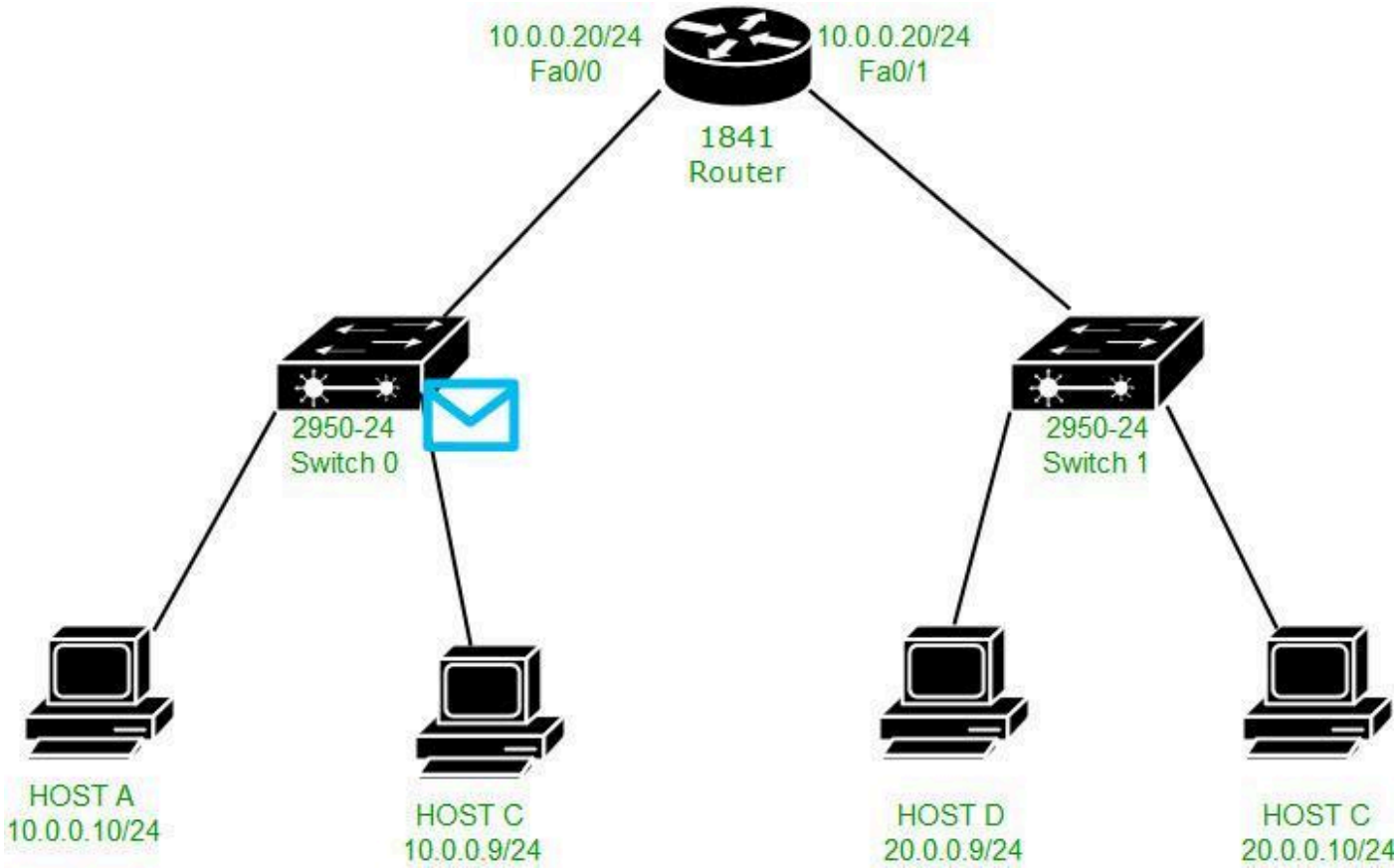


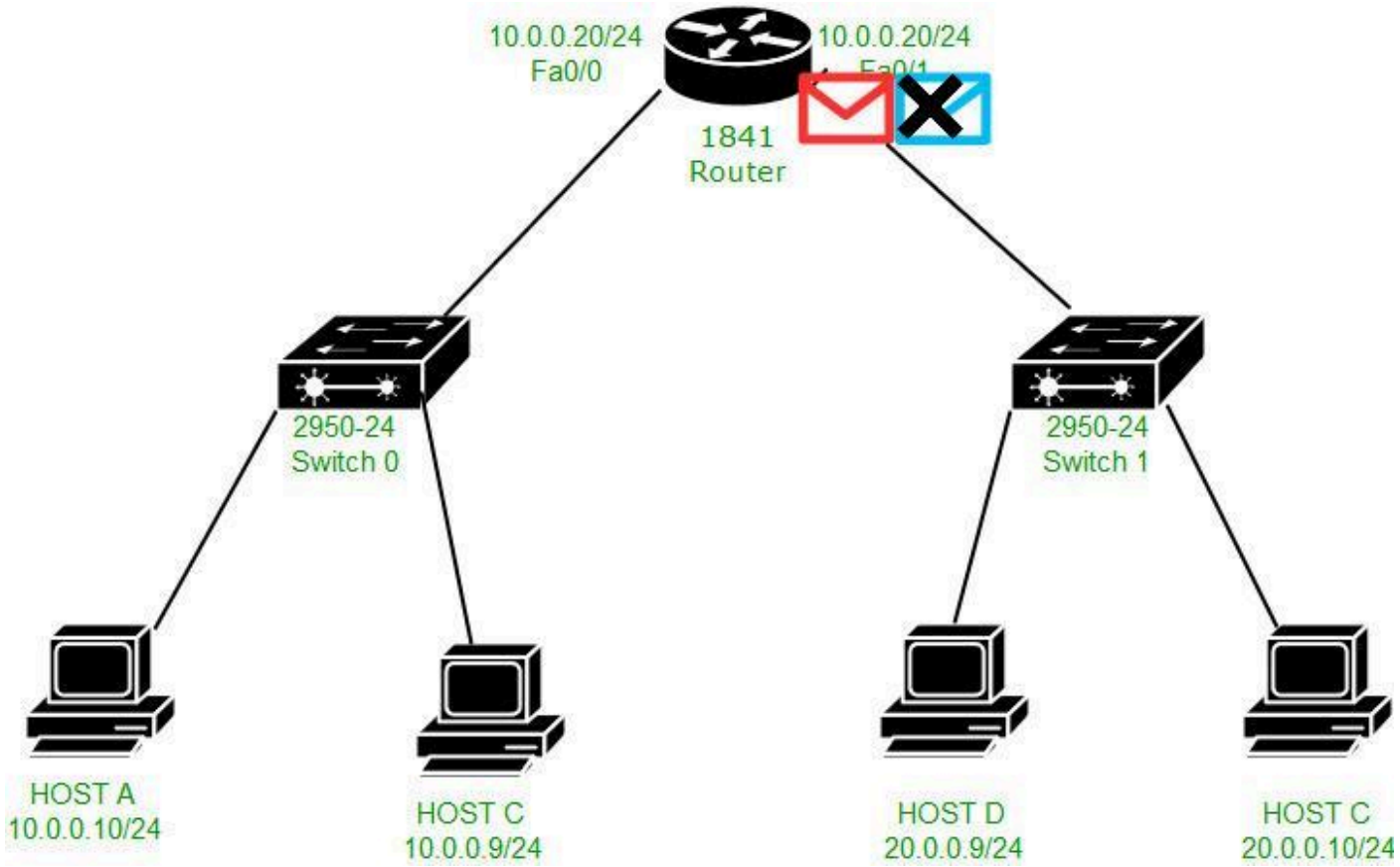
The switch in turn broadcast the ARP request to the host and the router. The PC discards the request and the router accepts it.





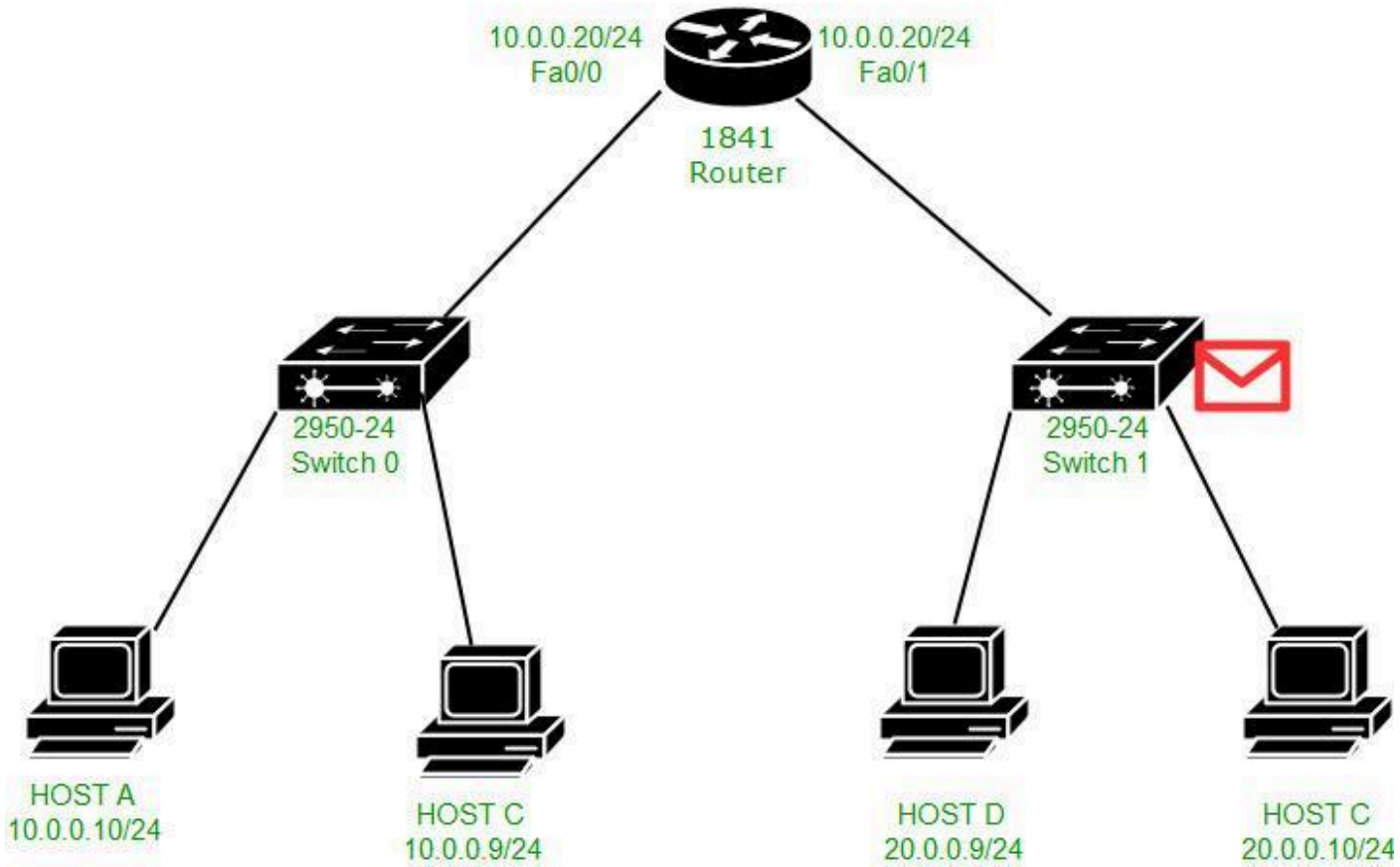
Now the ARP reply is unicast to host A by the router as shown in the above figure.

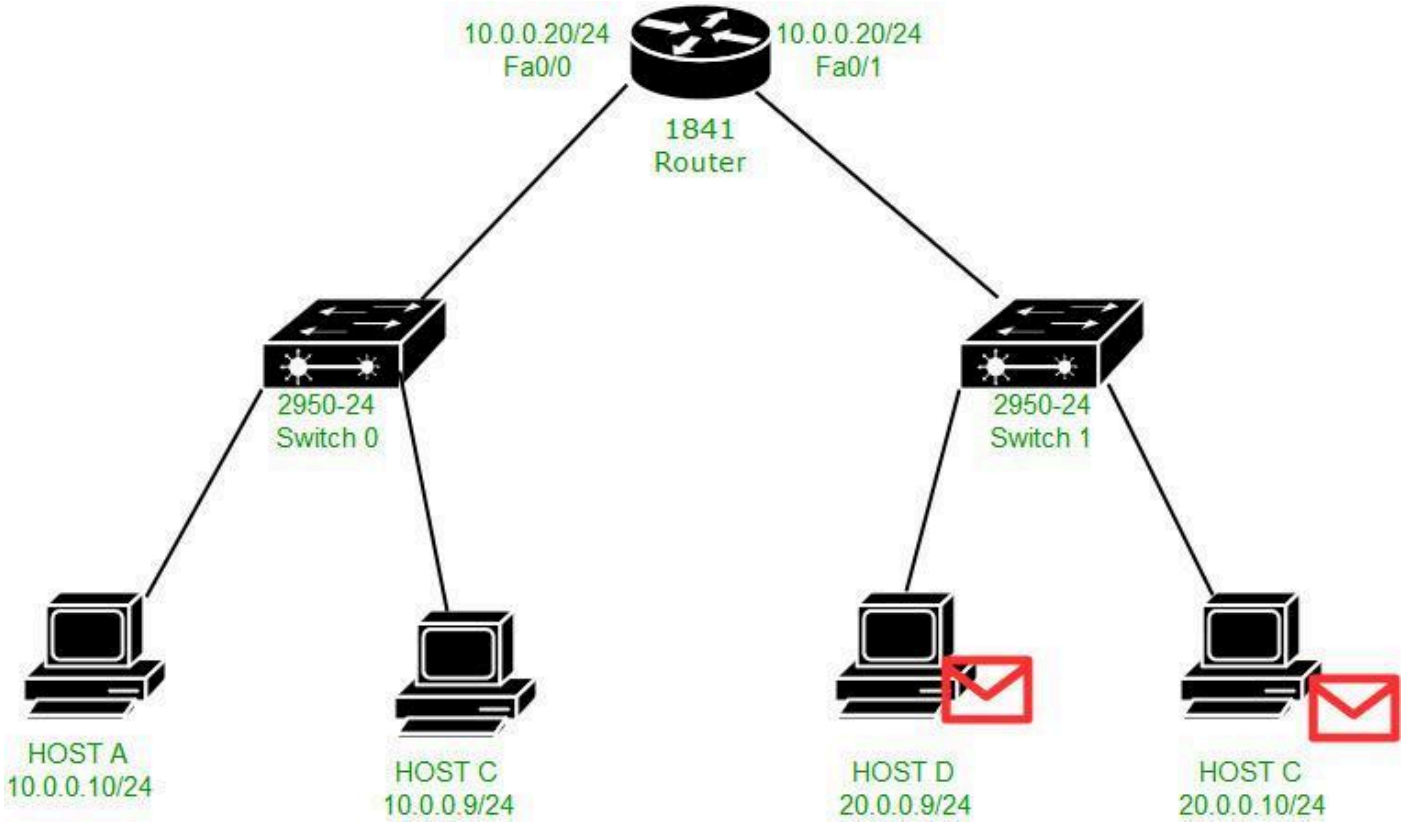




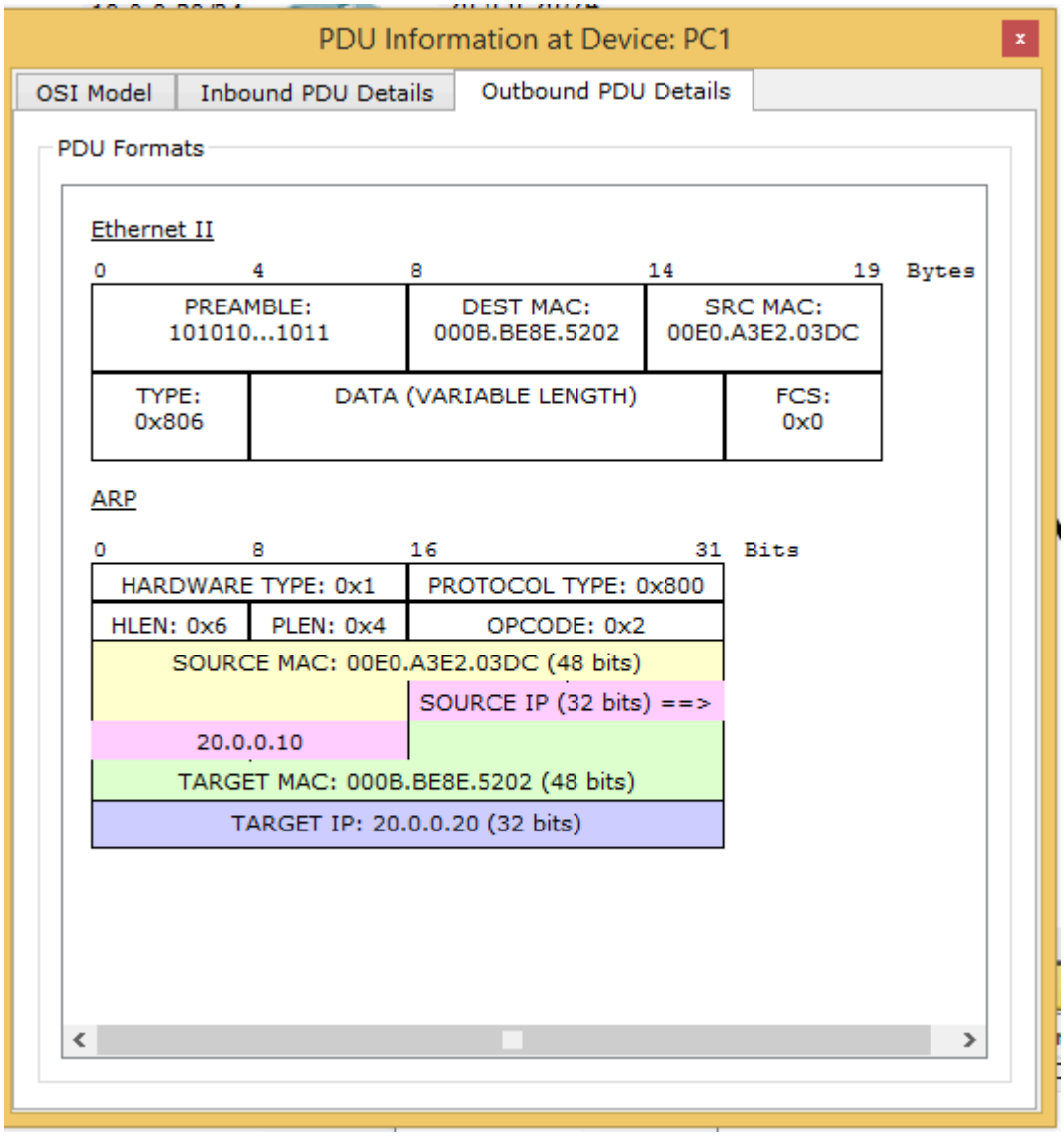
Now the ICMP packet will be unicast to the default gateway (IP address – 10.0.0.20 and MAC address – 000B.BE8E.5201) as shown in the above figures.

Note – The ICMP packet will be unicast to the default gateway as the ARP has been resolved now.



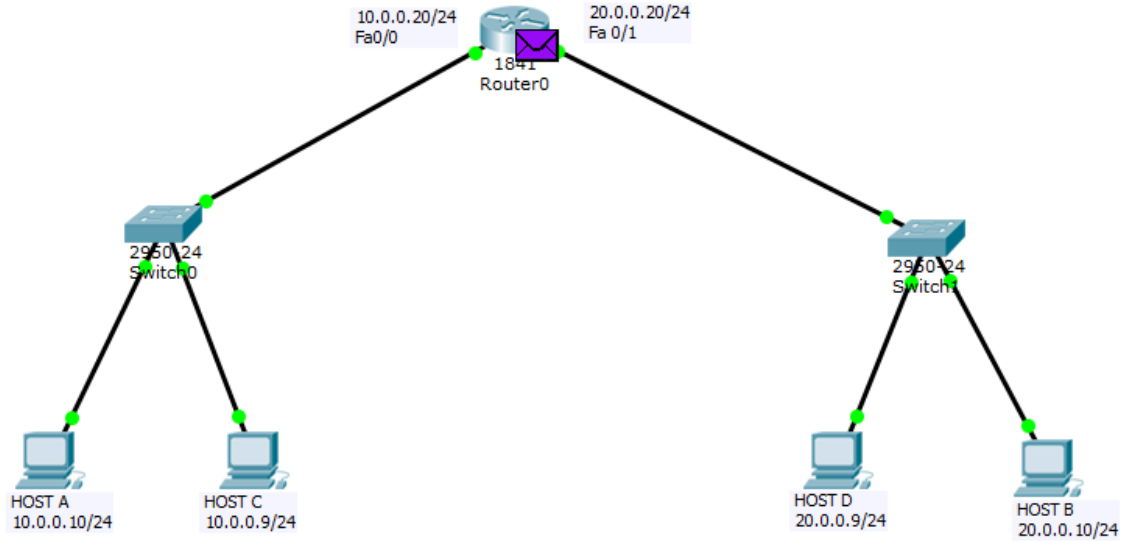


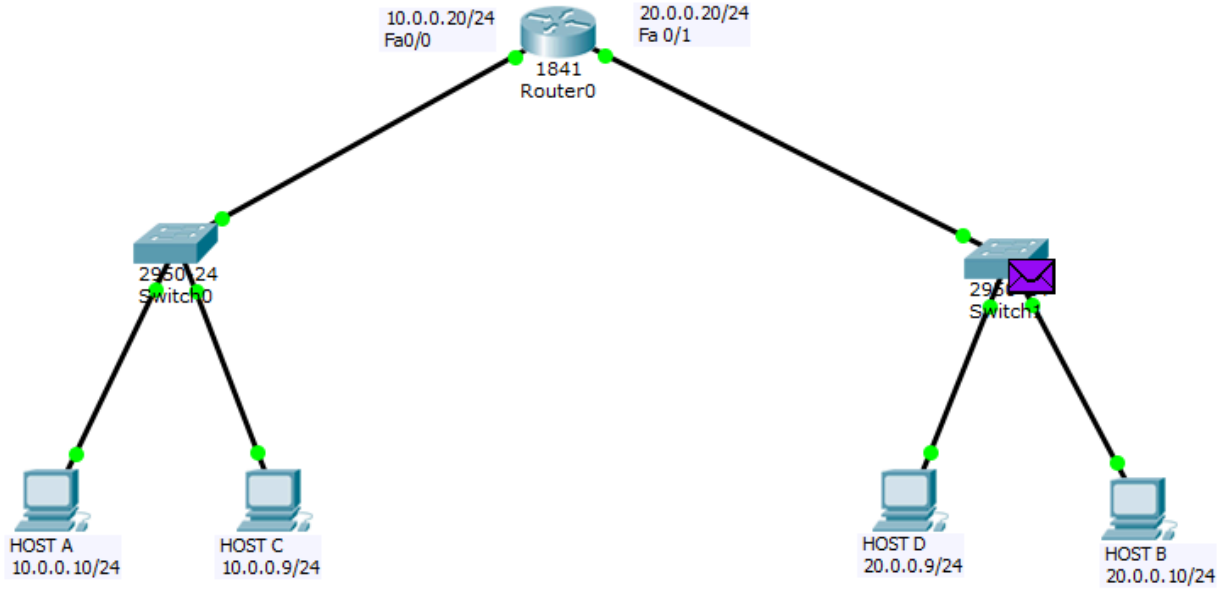
Now the ARP has to be resolved again because the router has to deliver the packet to host B and the ARP table has no entry for host B. Therefore, the ARP request is broadcast in the network 20.0.0.0/24. The packet is received by the Switch which in turn broadcast the request to host B and D. Host D will reject the request and host B will accept it and generate an ARP reply for the MAC address 000B.BE8E.5202 (router fa0/1 MAC address) because the ARP reply has to be given to that MAC address from which the ARP request has been received.

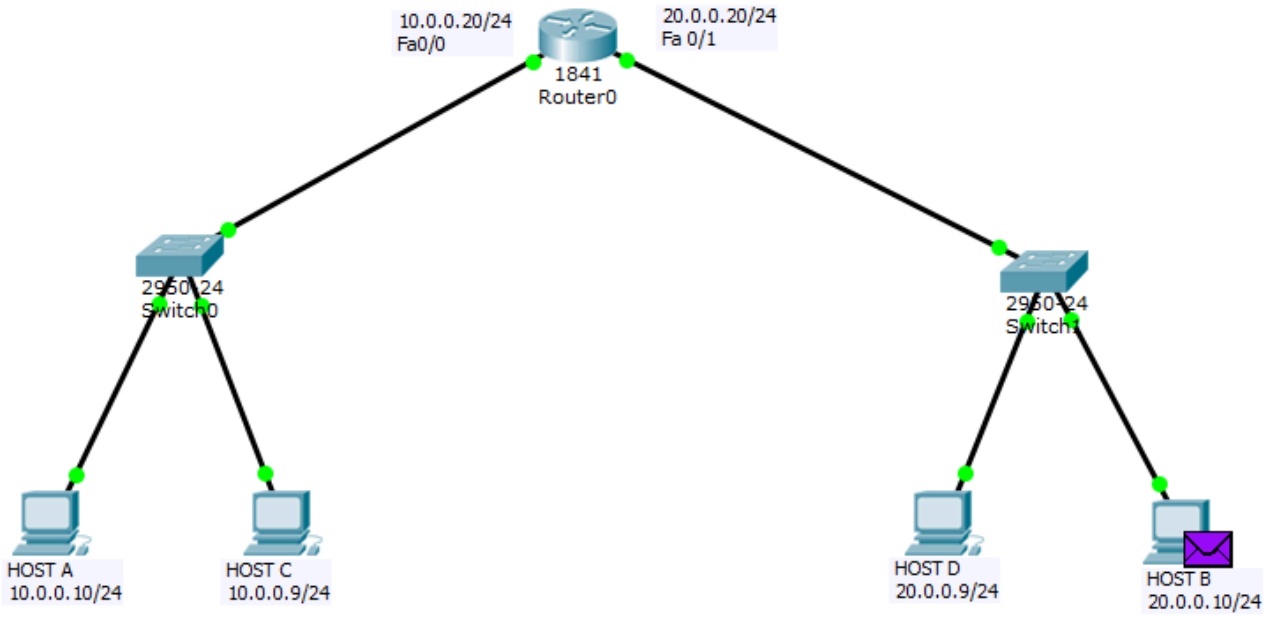


As you can see in the figure, the ARP reply packet is unicast to the router's interface fa0/1 MAC address(000B.BE8E.5202) and the source MAC is 00E0.A3E2.03DC.

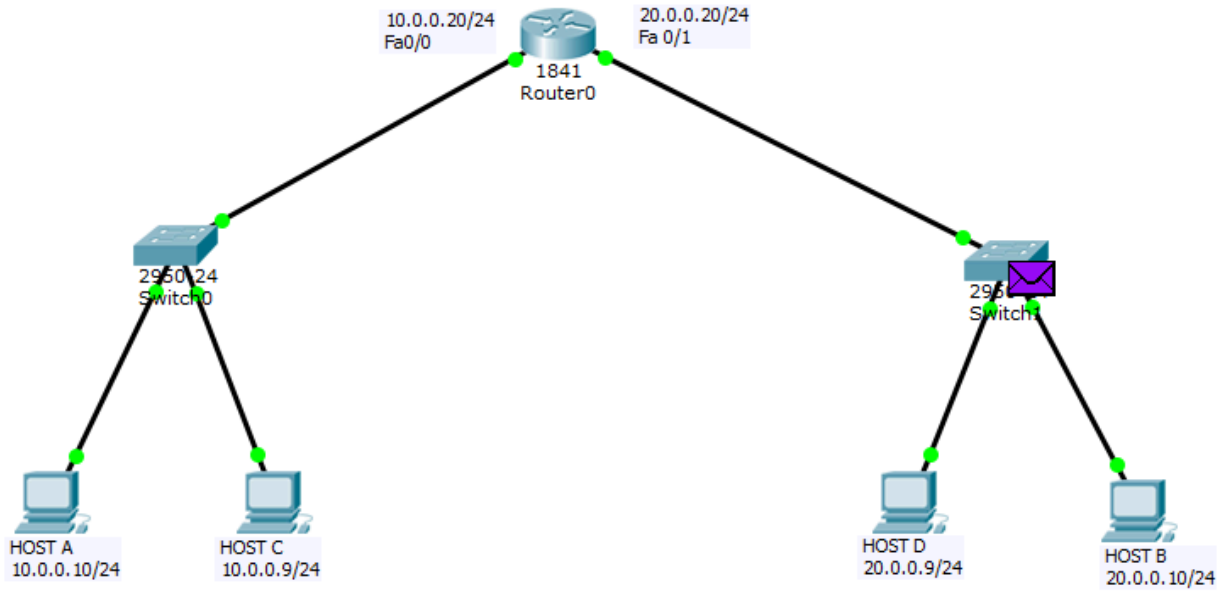
Note – Here, the target MAC address is the MAC address of host B (000B.BE8E.5202). Target MAC address is the MAC address of a device that the host wants to know through its ARP request to resolve ARP.

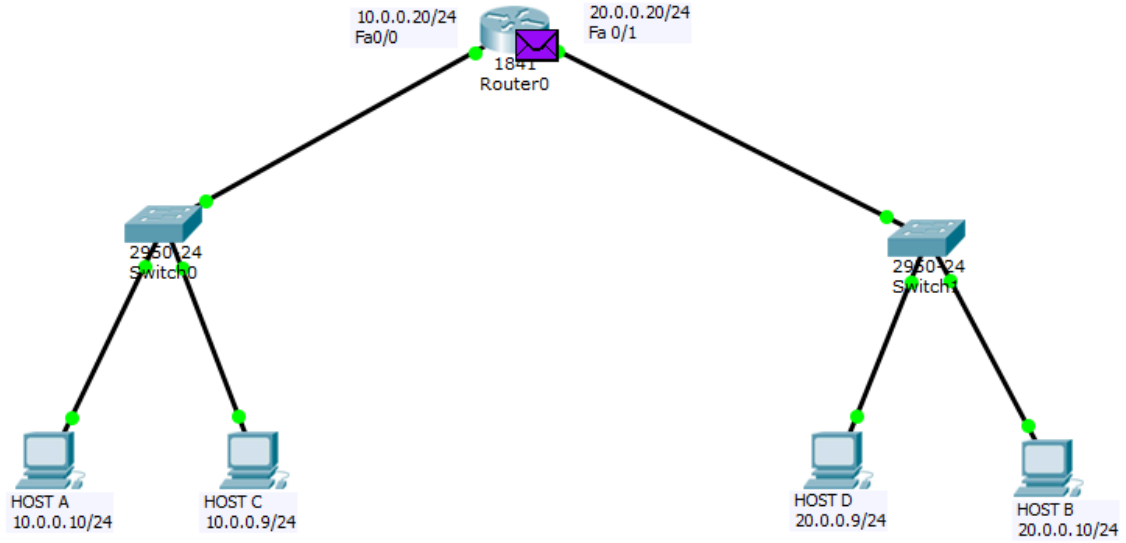

10.0.0.20/24
Fa0/0
20.0.0.20/24
Fa 0/1
1841
Router0
2950-24
Switch0
2950-24
Switch1
HOST A
10.0.0.10/24
HOST C
10.0.0.9/24
HOST D
20.0.0.9/24
HOST B
20.0.0.10/24

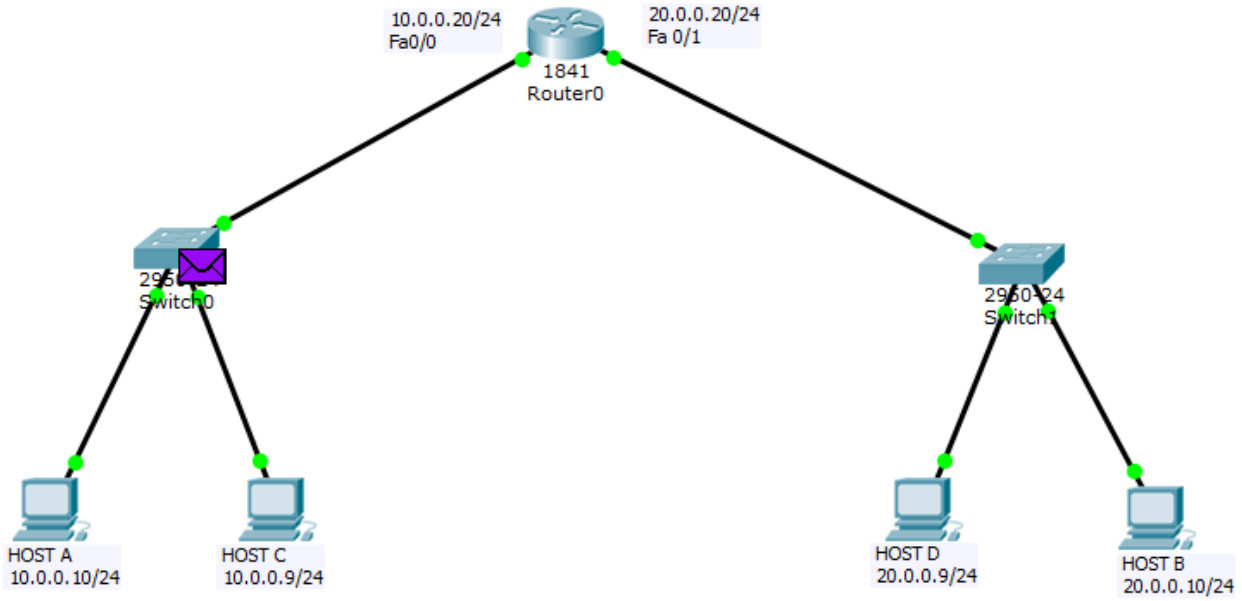

10.0.0.20/24
Fa0/0
20.0.0.20/24
Fa 0/1
1841
Router0
2950-24
Switch0
Switch1
HOST A
10.0.0.10/24
HOST C
10.0.0.9/24
HOST D
20.0.0.9/24
HOST B
20.0.0.10/24

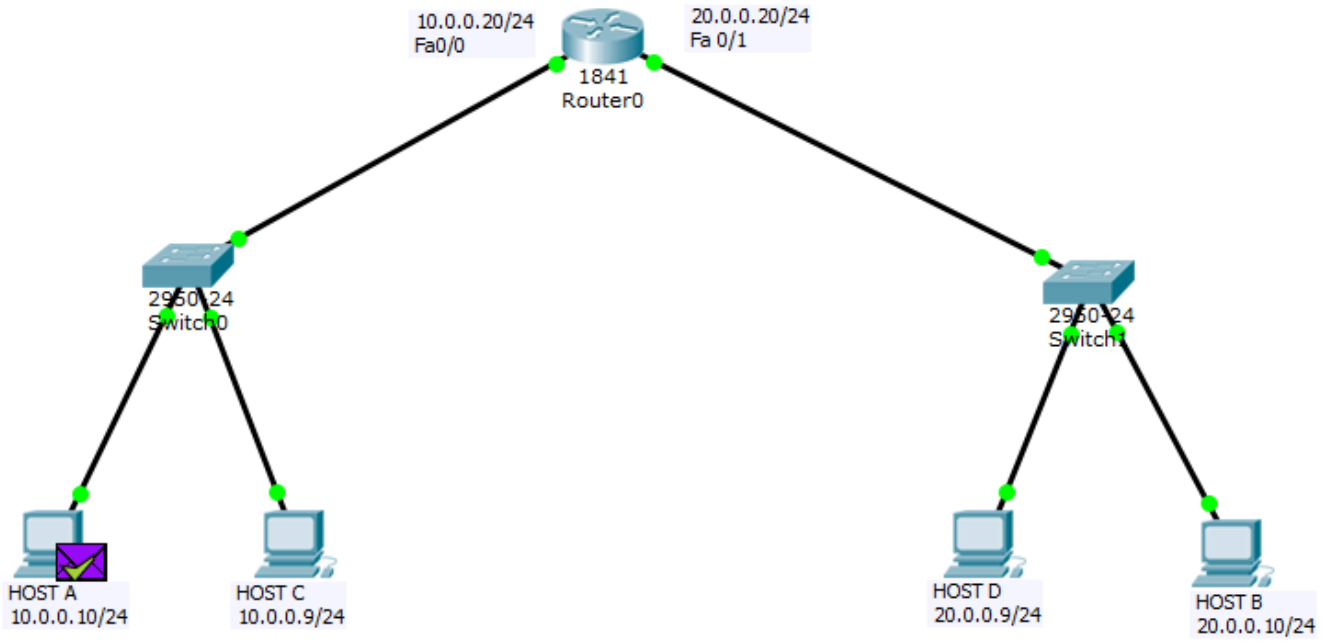


Now the ICMP echo-request packet will be unicast to the host B as shown in the above 3 figures.




10.0.0.20/24
Fa0/0
20.0.0.20/24
Fa 0/1
1841
Router0
2950-24
Switch0
2950-24
Switch1
HOST A
10.0.0.10/24
HOST C
10.0.0.9/24
HOST D
20.0.0.9/24
HOST B
20.0.0.10/24


10.0.0.20/24
Fa0/0
20.0.0.20/24
Fa 0/1
1841
Router0
2950-24
Switch0
2950-24
Switch1
HOST A
10.0.0.10/24
HOST C
10.0.0.9/24
HOST D
20.0.0.9/24
HOST B
20.0.0.10/24



Host B will generate an ICMP echo reply in response to the ICMP echo request for host A which will be delivered to the 20.0.0.20 (router's interface IP address) first and then unicast to host A.

How does the MAC address never crosses its broadcast domain?

PDU Information at Device: PC0

OSI Model | Outbound PDU Details

PDU Formats

Ethernet II

0	4	8	14	19 Byt
PREAMBLE: 101010...1011		DEST MAC: 000B.BE8E.5201	SRC MAC: 000D.BD22.7C22	
TYPE: 0x800	DATA (VARIABLE LENGTH)		FCS: 0x0	

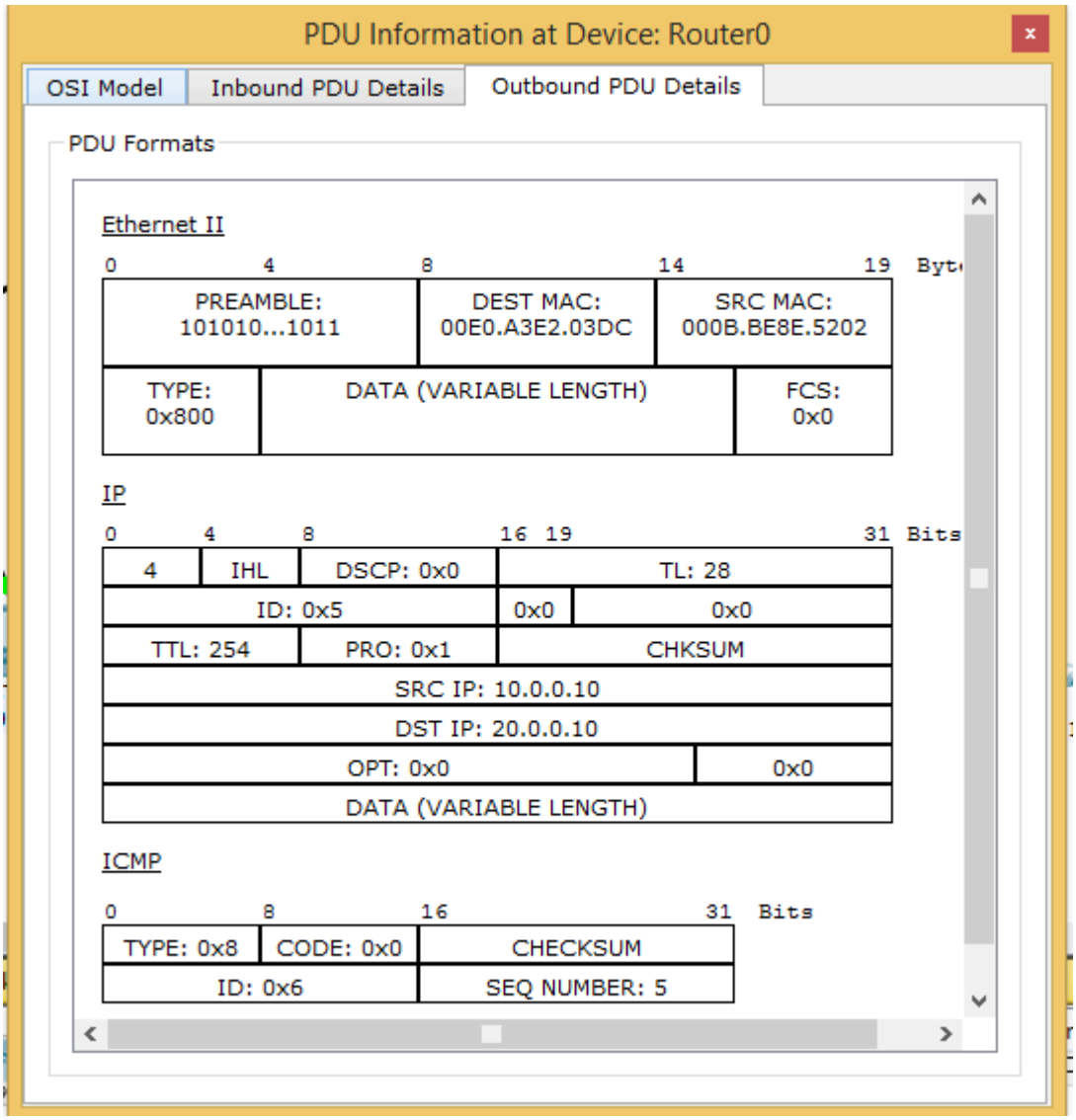
IP

0	4	8	16 19	31 Bits
4	IHL	DSCP: 0x0	TL: 28	
ID: 0x5		0x0	0x0	
TTL: 255		PRO: 0x1	CHKSUM	
SRC IP: 10.0.0.10				
DST IP: 20.0.0.10				
OPT: 0x0			0x0	
DATA (VARIABLE LENGTH)				

ICMP

0	8	16	31 Bits
TYPE: 0x8	CODE: 0x0	CHECKSUM	
ID: 0x6		SEQ NUMBER: 5	

This is the IP and Ethernet header when host A forwards the ICMP echo request to its default gateway. Therefore source IP is 10.0.0.10 and destination IP is 10.0.0.20, source MAC address is 000D.BD22.7C22 (host A MAC address) and destination MAC address is 000B.BE8E.5201 (router's fa0/0 interface MAC address).



But now when the ICMP echo request message is forwarded from the router's fa0/1 interface to host B then the source MAC address is changed to 000B.BE8E.5202 (router's fa0/1 interface MAC address) and destination MAC address is 00E0.A3E2.03DC (host B MAC address).

Here router's fa0/0 interface MAC address is not used as the source MAC address, instead the fa0/1 MAC address is used as a MAC address. Therefore, fa0/0 is not used in other broadcast domains (20.0.0.0/24 network) therefore MAC address never crosses its broadcast domain. IN this way, PING is performed in 2 different networks.