NETWORKING TRAINING - MODULE 5

Akash S, embedUR Systems

Qn 1:

Capture and analyze ARP packets using Wireshark. Inspect the ARP request and reply frames when your device attempts to find the router's MAC address. Discuss the importance of ARP in packet forwarding

arp:

- I started by identifying the IP address of my Linux Virtual Box and my Windows Host IP
    - Linux Vbox: **192.168.29.220**
    - Windows IP: **192.168.29.112**
- Confirmed that they are in the same network
- I opened the Wireshark and selected the interface **enp0s3**
- I pinged from my Linux VM to my Windows Host IP
- It worked properly!

What arp does?

- ARP - **Address Resolution Protocol**
- ARP helps in IP-MAC Mapping
- ARP maps IP (Logical Address) and MAC (Physical address)
- We can check the arp cache using the command: **$ arp -a**

Why arp?

- If a device wants to send packets, it should know its MAC address

ARP Request:

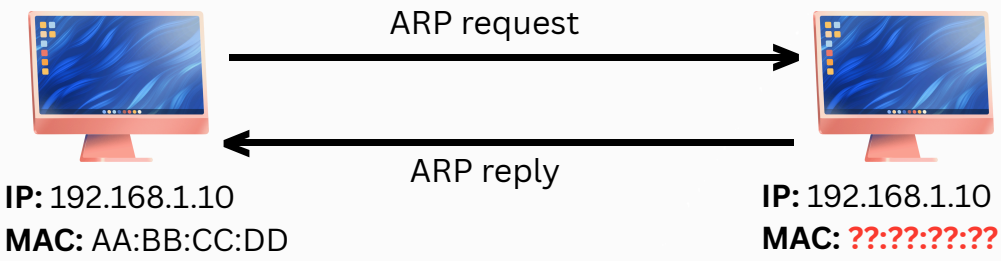
- If the device's MAC is unknown, ARP sends **ARP Request**.
- ARP Request is broadcast, right device will reply.

```
▸ Frame 30: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface enp0s3, id 0
▸ Ethernet II, Src: SkyworthDigi_b6:f2:01 (04:ab:08:b6:f2:01), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▾ Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: SkyworthDigi_b6:f2:01 (04:ab:08:b6:f2:01)
    Sender IP address: 192.168.29.1
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.29.112

0000  ff ff ff ff ff ff 04 ab  08 b6 f2 01 08 06 00 01   ···· ···· ····
0010  08 00 06 04 00 01 04 ab  08 b6 f2 01 c0 a8 1d 01   ···· ···· ····
0020  00 00 00 00 00 00 c0 a8  1d 70 00 00 00 00 00 00   ···· ···· ·p····
0030  00 00 00 00 00 00 00 00  00 00 00 00               ···· ···· ····
```

ARP Reply:

- The target device responds with an ARP reply with its MAC address




```
▸ Frame 3837: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface enp0s3, id 0
▸ Ethernet II, Src: ChongqingFug_34:f1:69 (c8:94:02:34:f1:69), Dst: PCSSystemtec_0e:cd:8c (08:00:2
▾ Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: ChongqingFug_34:f1:69 (c8:94:02:34:f1:69)
    Sender IP address: 192.168.29.112
    Target MAC address: PCSSystemtec_0e:cd:8c (08:00:27:0e:cd:8c)
    Target IP address: 192.168.29.220

0000  08 00 27 0e cd 8c c8 94  02 34 f1 69 08 06 00 01   ··'····· ·4·i····
0010  08 00 06 04 00 02 c8 94  02 34 f1 69 c0 a8 1d 70   ········ ·4·i···p
0020  08 00 27 0e cd 8c c0 a8  1d dc 00 00 00 00 00 00   ··'····· ········
0030  00 00 00 00 00 00 00 00  00 00 00 00               ···· ···· ····
```

```
arp && (arp.src.proto_ipv4 == 192.168.29.112 || arp.dst.proto_ipv4 == 192.168.29.112)
No.   Time          Source                Destination           Protocol Length Info
   30 19.179404058  SkyworthDigi_b6:f2:…  Broadcast             ARP      60 Who has 192.168.29.112? Tell 192.168.29.1
 1440 47.165081530  SkyworthDigi_b6:f2:…  Broadcast             ARP      60 Who has 192.168.29.112? Tell 192.168.29.1
 1572 48.145886970  SkyworthDigi_b6:f2:…  Broadcast             ARP      60 Who has 192.168.29.112? Tell 192.168.29.1
 2619 82.675038465  SkyworthDigi_b6:f2:…  Broadcast             ARP      60 Who has 192.168.29.112? Tell 192.168.29.1
 2631 86.838855768  SkyworthDigi_b6:f2:…  ChongqingFug_34:f1:…  ARP      60 192.168.29.1 is at 04:ab:08:b6:f2:01
 2684 90.465810883  PCSSystemtec_0e:cd:…  Broadcast             ARP      42 Who has 192.168.29.112? Tell 192.168.29.220
 2685 90.466011893  ChongqingFug_34:f1:…  PCSSystemtec_0e:cd:…  ARP      60 192.168.29.112 is at c8:94:02:34:f1:69
 2698 95.335264961  ChongqingFug_34:f1:…  PCSSystemtec_0e:cd:…  ARP      60 Who has 192.168.29.220? Tell 192.168.29.112
 2699 95.335291789  PCSSystemtec_0e:cd:…  ChongqingFug_34:f1:…  ARP      42 192.168.29.220 is at 08:00:27:0e:cd:8c
 3133 109.836595775 SkyworthDigi_b6:f2:…  ChongqingFug_34:f1:…  ARP      60 192.168.29.1 is at 04:ab:08:b6:f2:01
 3222 121.622090790 SkyworthDigi_b6:f2:…  Broadcast             ARP      60 Who has 192.168.29.112? Tell 192.168.29.1
 3571 146.697027690 SkyworthDigi_b6:f2:…  Broadcast             ARP      60 Who has 192.168.29.112? Tell 192.168.29.1
 3583 147.645405805 SkyworthDigi_b6:f2:…  Broadcast             ARP      60 Who has 192.168.29.112? Tell 192.168.29.1

▸ Frame 1572: 60 bytes on wire (480 bits), 60 bytes
▸ Ethernet II, Src: SkyworthDigi_b6:f2:01 (04:ab:08
▾ Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: SkyworthDigi_b6:f2:01 (04:a
    Sender IP address: 192.168.29.1
    Target MAC address: ChongqingFug_34:f1:69 (c8:9
    Target IP address: 192.168.29.112

0000  ff ff ff ff ff ff 04 ab  08 b6 f2 01 08 06 00 01
0010
0020
0030

akash@akash: ~
akash@akash: ~   akash@akash: ~
akash@akash:~$ ping 192.168.29.112
PING 192.168.29.112 (192.168.29.112) 56(84) bytes of data.
64 bytes from 192.168.29.112: icmp_seq=1 ttl=128 time=0.385 ms
64 bytes from 192.168.29.112: icmp_seq=2 ttl=128 time=0.892 ms
64 bytes from 192.168.29.112: icmp_seq=3 ttl=128 time=0.719 ms
64 bytes from 192.168.29.112: icmp_seq=4 ttl=128 time=0.369 ms
64 bytes from 192.168.29.112: icmp_seq=5 ttl=128 time=0.308 ms
64 bytes from 192.168.29.112: icmp_seq=6 ttl=128 time=0.370 ms
```

Qn 2:

Manually configure static routes on a router to direct packets to different subnets.
Use the ip route command and verify connectivity using ping and traceroute.

Network topology:

- I created a network with two subnets
- Two PCs and two Routers
    - PC1: **192.168.1.2**
    - PC2: **192.168.2.2**
- Default gateway
    - Router 1: **192.168.1.1**
    - Router 2: **192.168.2.1**
- Static route:
    - Router1: **10.0.0.1**
    - Router2: **10.0.0.2**

How it works?

PC1 - Router

- We need PC1 to Communicate with PC2
- PC1 finds that PC2 is in different subnet
- So it first sends the packet to the default gateway

Router1 - Router2:

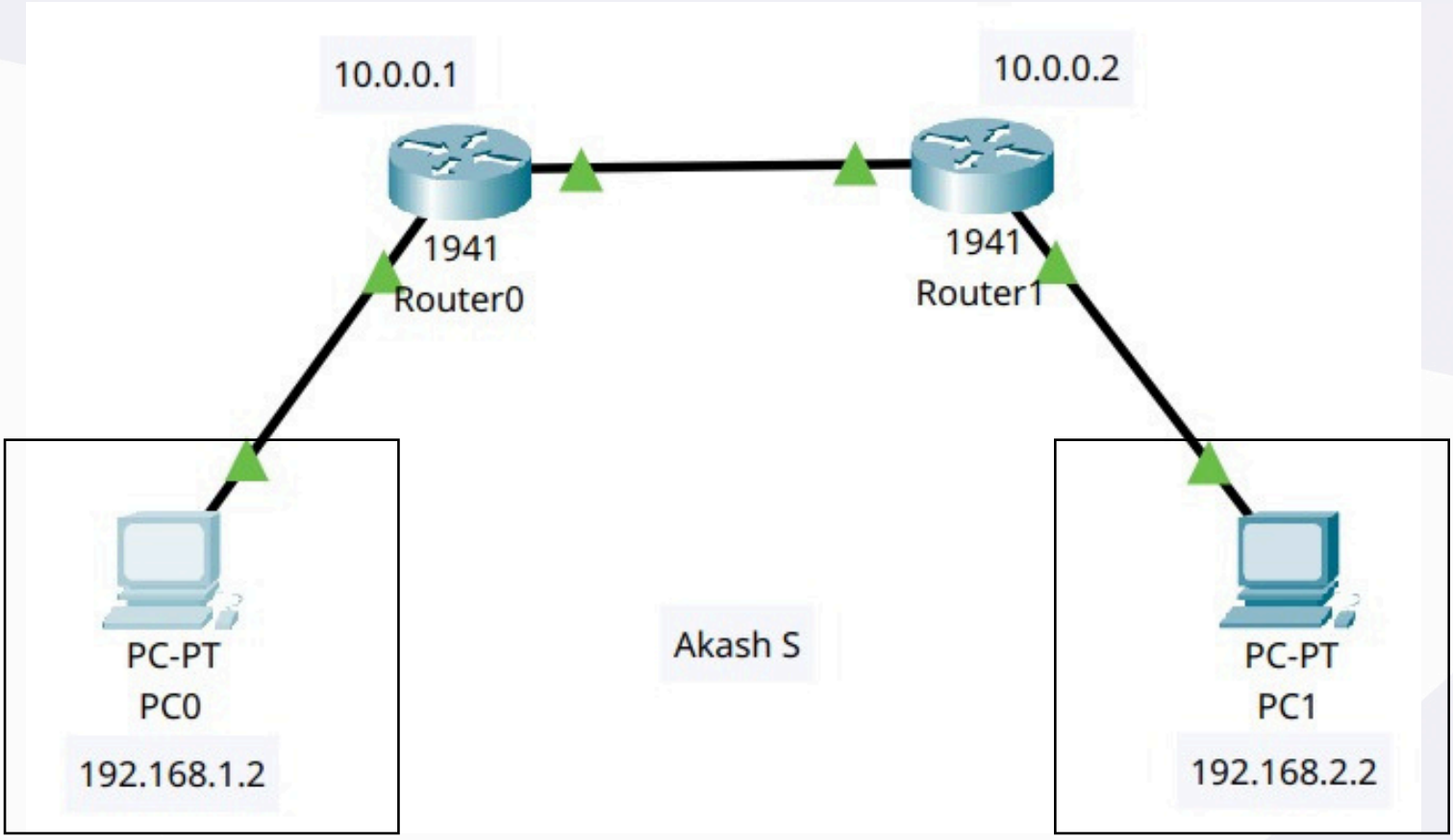
- Router checks the routing table
- It finds that PC2 is not directly connected
- So it checks for **Static ip route**
- Static ip tells to forward the packets to **10.0.0.2** (Router2)

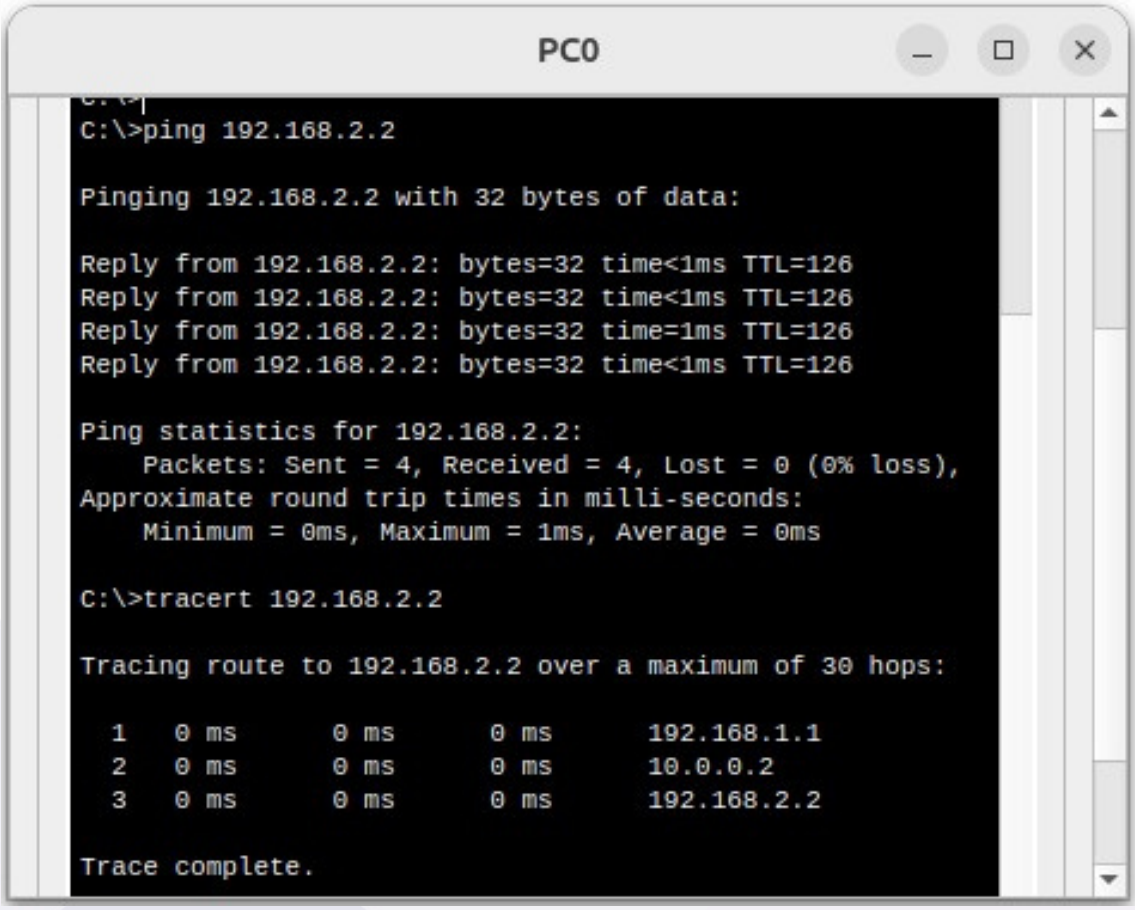
Router2 - PC2:

- Router2 checks the routing table and finds PC2 is directly connected
- So it sends the packet directly and PC2 sends a reply

Output:

Network topology:





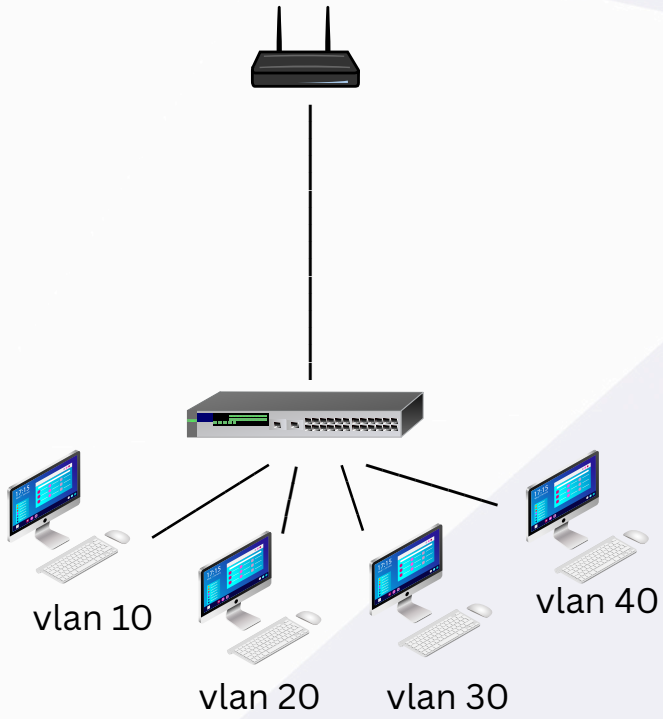
ping and traceroute

Qn 3:

Given a network address of 10.0.0.0/24, divide it into 4 equal subnets.
Calculate the new subnet mask.
Determine the valid host range for each subnet.
Assign IP addresses to devices in Packet Tracer and verify connectivity

Network topology:

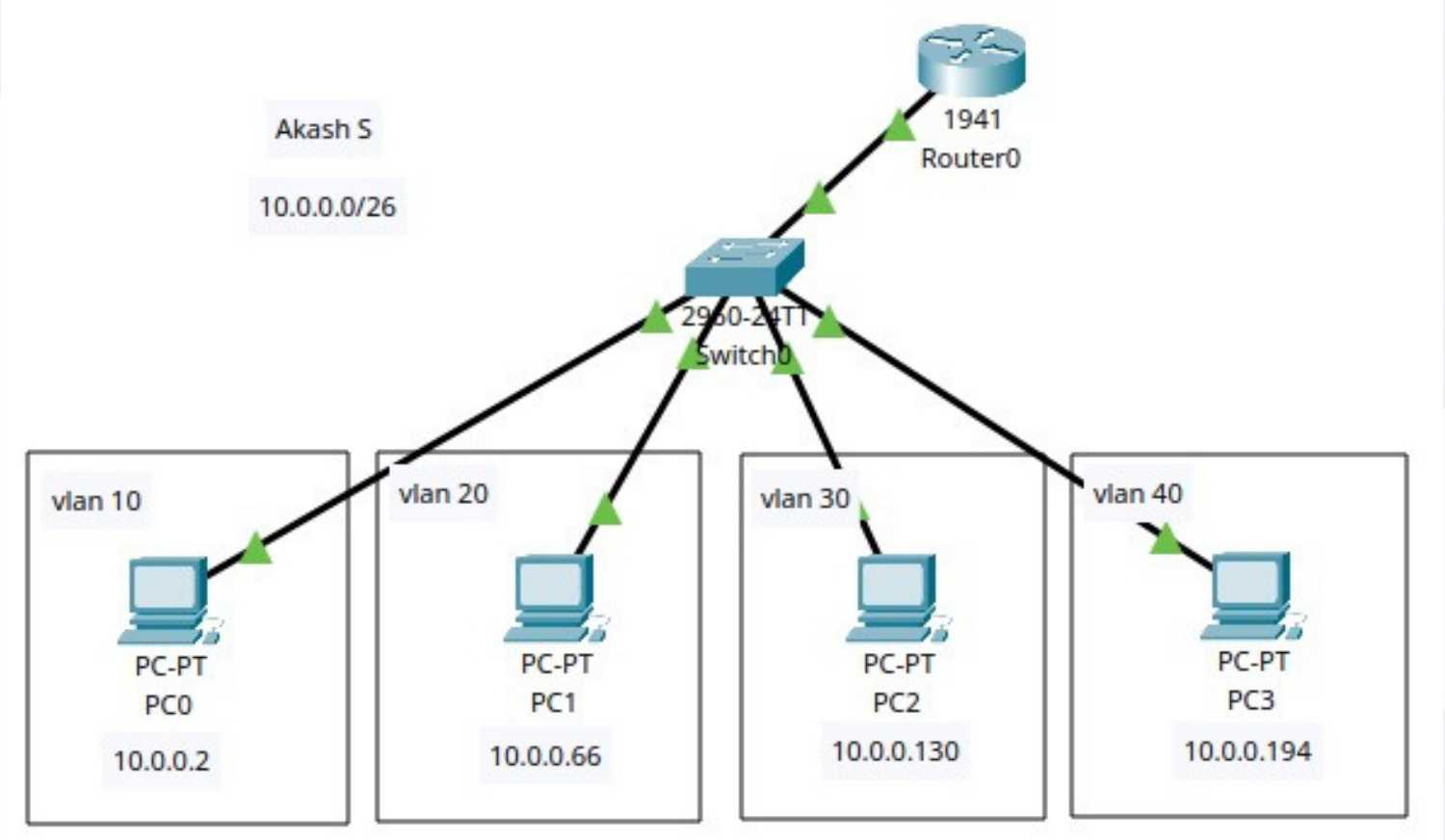
- I created a network with 4 PCs, 1 Switch and 1 Router
- PC IP Configuration
    - PC1: **10.0.0.2**
    - PC2: **10.0.0.66**
    - PC3: **10.0.0.130**
    - PC4: **10.0.0.194**
- Configuring VLANs in switch
    - PC1 - VLAN 10
    - PC2 - VLAN 20
    - PC3 - VLAN 30
    - PC4 - VLAN 40
- Router configuration
    - Added the RIP - **10.0.0.0**
- The whole network is **10.0.0.0/26**



New Subnet Mask Calculation:

- We need 4 subnets. This means we have to borrow bits from the host portion of the address.
- The original /24 subnet has 8 host bits
- To create 4 subnets, we need to borrow 2 bits from the host portion
    - 2^2 = **4**
- Borrowing 2 bits from the host portion increases the subnet mask:
    - /24 + 2 = **/26**
- So the new subnet mask is: 🟢 **255.255.255.192**

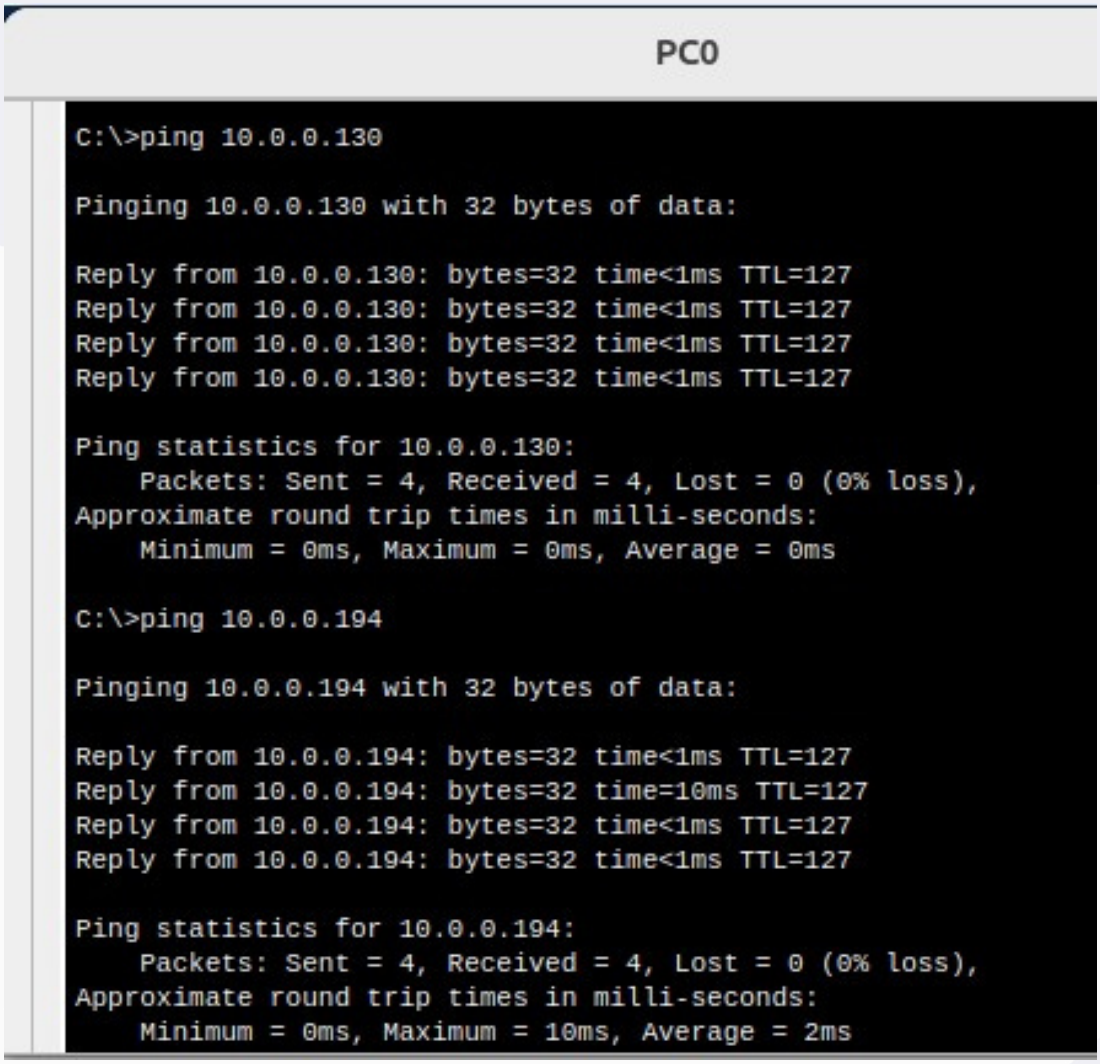
Network Topology:

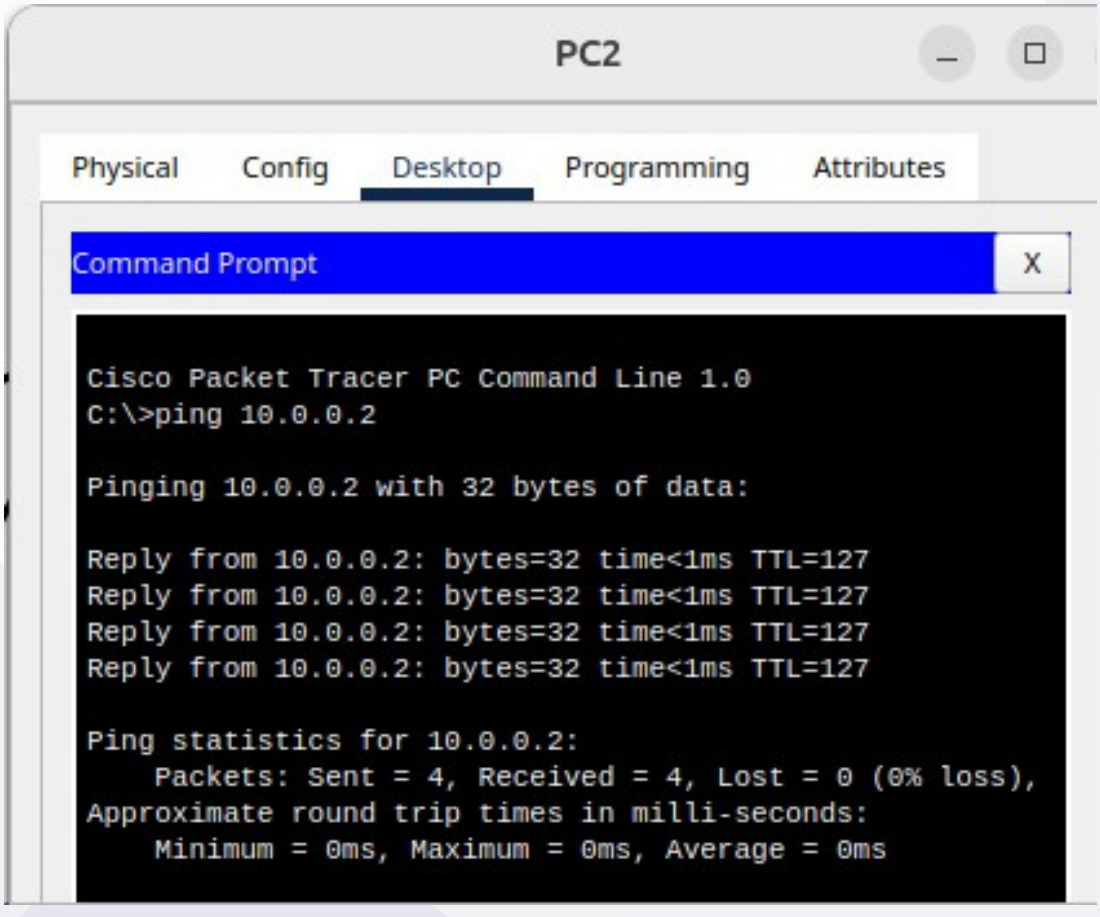


Subnet Classification:

Subnet 1	10.0.0.0/26	10.0.0.1	10.0.0.62	10.0.0.63
Subnet 2	10.0.0.64/26	10.0.0.65	10.0.0.126	10.0.0.127
Subnet 3	10.0.0.128/26	10.0.0.129	10.0.0.190	10.0.0.191
Subnet 4	10.0.0.192/26	10.0.0.193	10.0.0.254	10.0.0.255

Connectivity Tests:





Qn 4:

You are given three IP addresses: 192 168.10.5, 172.20 15 1, and 8.8.8.8.
Identify the class of each IP address.
Determine if it is private or public.
Explain how NAT would handle a private IP when accessing the internet.

Documentation:

Identification:

1. IP Address: **192.168.10.5**
   Class: **C**
   Type: Private IP

2. IP Address: **172.20.15.1**
   Class: **B**
   Type: Private IP

3. IP Address: **8.8.8.8**
   Class: **A**
   Type: Public IP

Class Range:

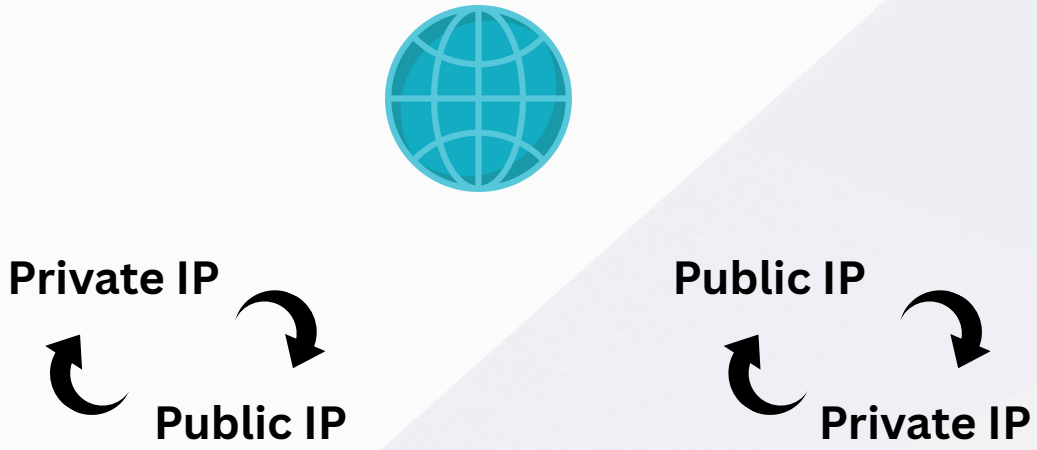
Class	Starting IP	Ending IP	Default Subnet Mask
A	1.0.0.0	126.255.255.255	255.0.0.0
B	128.0.0.0	191.255.255.255	255.255.0.0
C	192.0.0.0	223.255.255.255	255.255.255.0

NAT:

- NAT - Network Address Translation
- NAT allows Private IP devices to communicate in the network
- NAT translates Private IP to Public IP assigned by routers
- By this way, multiple Private IPs share a single Public IP

Process:

- A device with a private IP sends a request to the internet.
- Router replaces the Private IP with a public IP before forwarding it.
- When the response returns, the router translates the public IP back to the original private IP.
- This allows multiple devices to share a single public IP while maintaining unique connections.



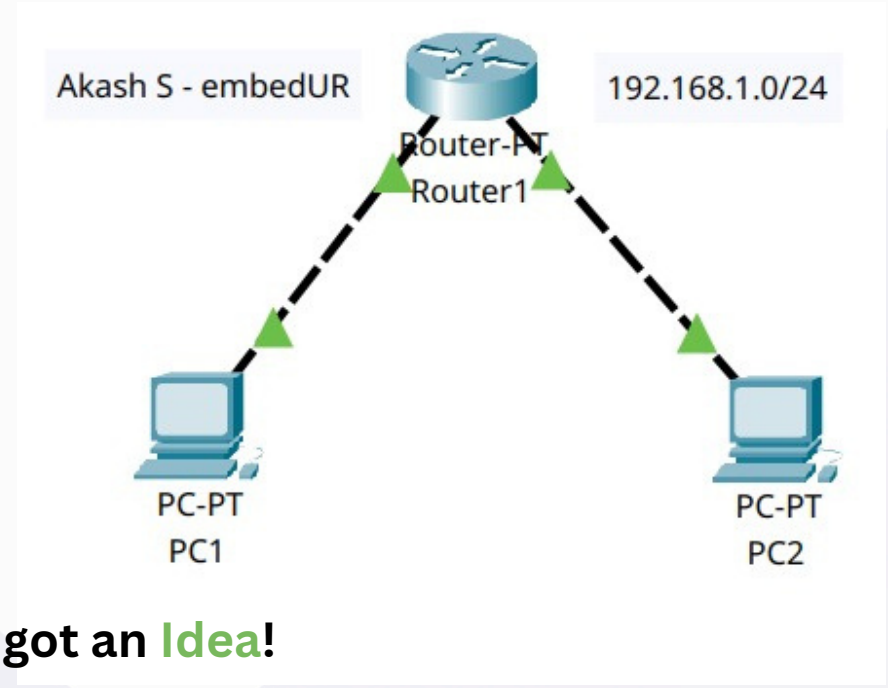
Qn 5:

In Cisco Packet Tracer, configure NAT on a router to allow internal devices (192.168.1.x) to access the internet.
Test connectivity by pinging an external public IP
Capture the traffic in Wireshark and analyze the source IP before and after NAT translation

Attempt 1:

- I connected two VPCs and a router
    - PC1: **192.168.1.10**
    - PC2: **192.168.2.20**
- Enabled the ports in router
- Pinging from PC1 to Router worked ✓
- Pinging from PC2 to Router didn't work ✗
- Pinging from PC1 to PC2 also didn't work ✗

There's a problem!



Finally got an Idea!

- I haven't added **RIP** in the router so that PC2 is not configured in the router.
- Finally it worked!

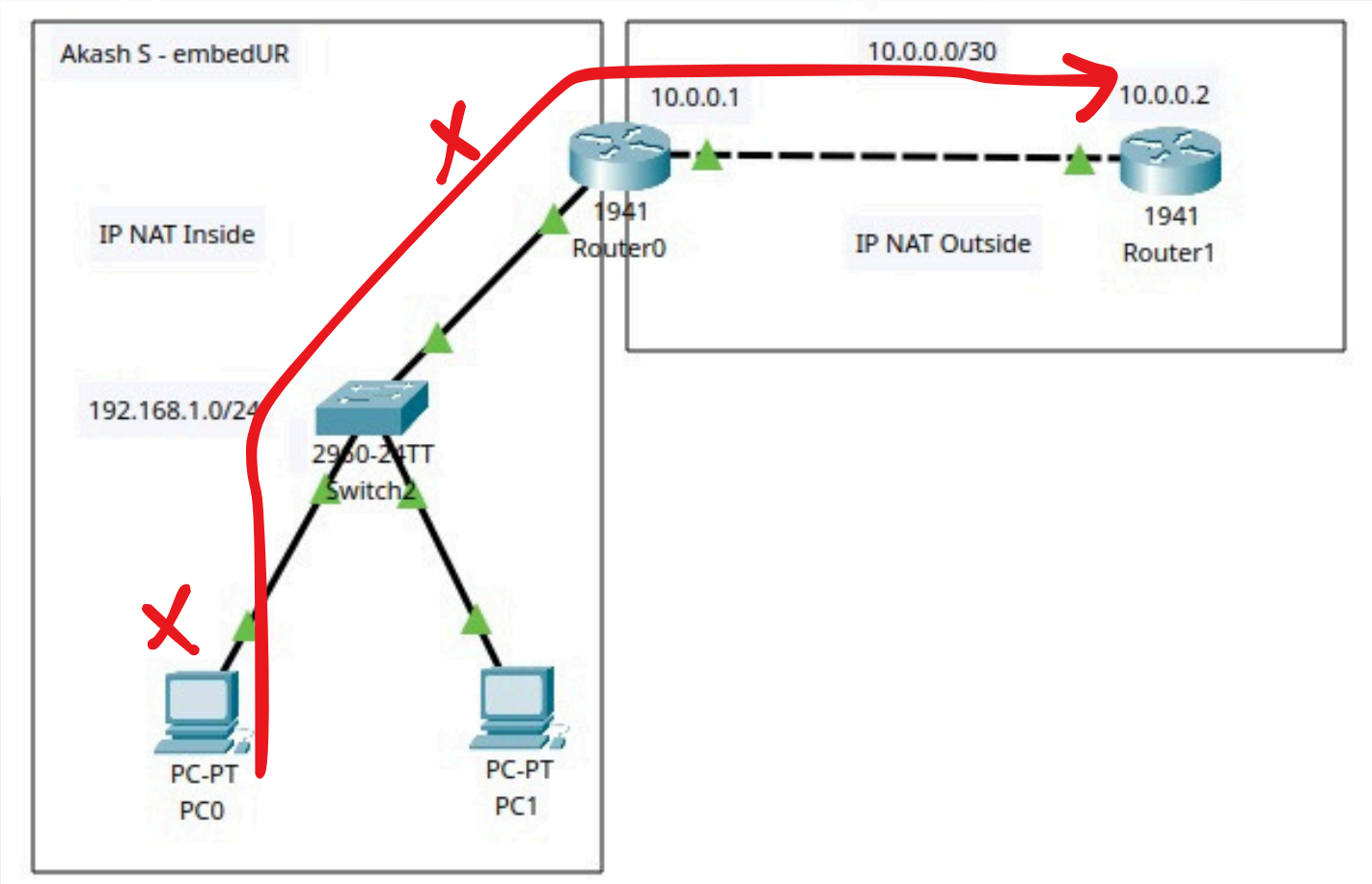
Diving into the actual network:

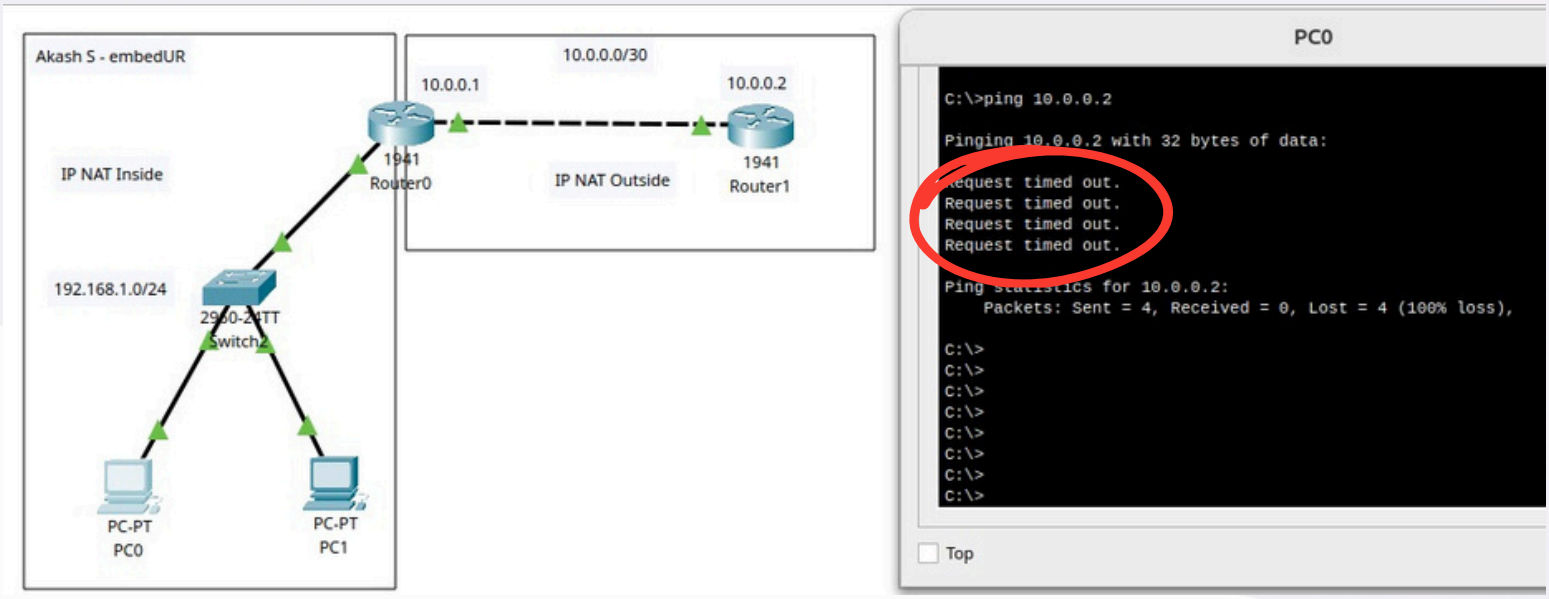
Attempt 2:

- I created two NAT networks
    - IP NAT Inside
    - IP NAT Outside
- In IP NAT Inside, I have connected 2 VPCs, a switch and a Router
    - PC1: **192.168.10.10**
    - PC2: **192.168.10.20**
    - Router1: **10.0.0.1**
- In IP NAT Outside, I have connected a router (external)
    - Router2: **10.0.0.2**
- Enabled both ended ports of routers.
- Checked connectivity, it failed!

There's a problem!

- I can ping from PC0 to Router 1 but I can't ping from PC1 to Router1





Attempt 3:

- I enabled each and every interface manually - **UP**

Now it works!

```
Router>enable
Router#ping 192.168.10.10

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/3 ms

Router#ping 192.168.10.20

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.20, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 0/0/0 ms

Router#ping 10.0.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms
```

Router's CLI

```
C:\>ping 10.0.0.2

Pinging 10.0.0.2 with 32 bytes of data:

Reply from 10.0.0.2: bytes=32 time<1ms TTL=254
Reply from 10.0.0.2: bytes=32 time<1ms TTL=254
Reply from 10.0.0.2: bytes=32 time<1ms TTL=254
Reply from 10.0.0.2: bytes=32 time<1ms TTL=254

Ping statistics for 10.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

PC0's CMD

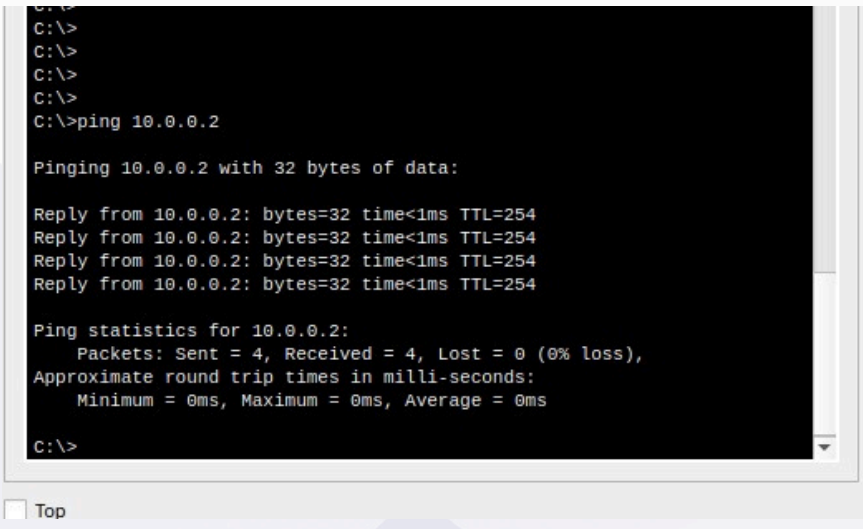
Attempt 4:

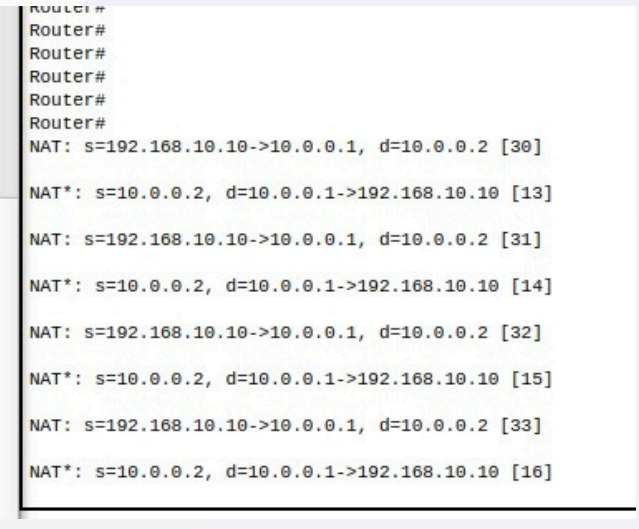
- Now I can’t reach Router 2 from **PC1**
- I tried debugging through Router’s CLI and PC1's Command prompt
- Configured
    - **IP NAT Inside**
    - **IP NAT Outside**

Now it works!

```
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface GigabitEthernet0/1
Router(config-if)#ip nat inside
Router(config-if)#exit
Router(config)#
Router(config)#
Router(config)#
Router(config)#interface GigabitEthernet0/0
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#
Router(config)#
Router(config)#ip nat inside source ?
  list    Specify access list describing local addresses
  static  Specify static local->global mapping
Router(config)#ip nat inside source ?
  list    Specify access list describing local addresses
  static  Specify static local->global mapping
Router(config)#ip nat inside source static 192.168.10.10
% Incomplete command.
Router(config)#ip nat inside source static 192.168.10.10 10.0.0.1
Router(config)#
```

Router’s CLI



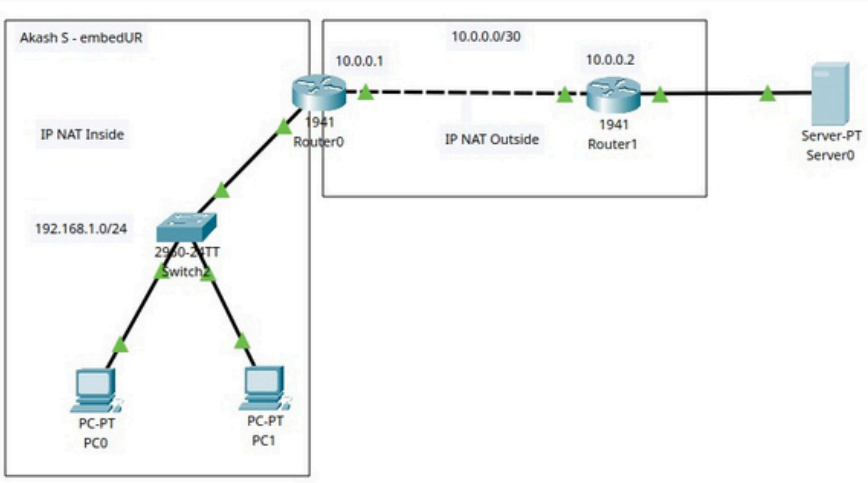


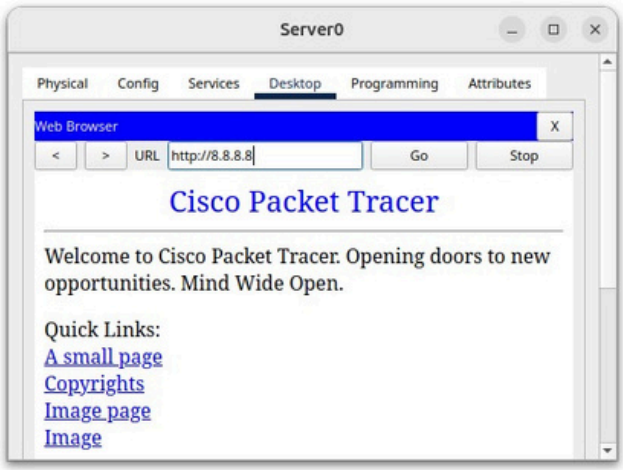
Final Output:

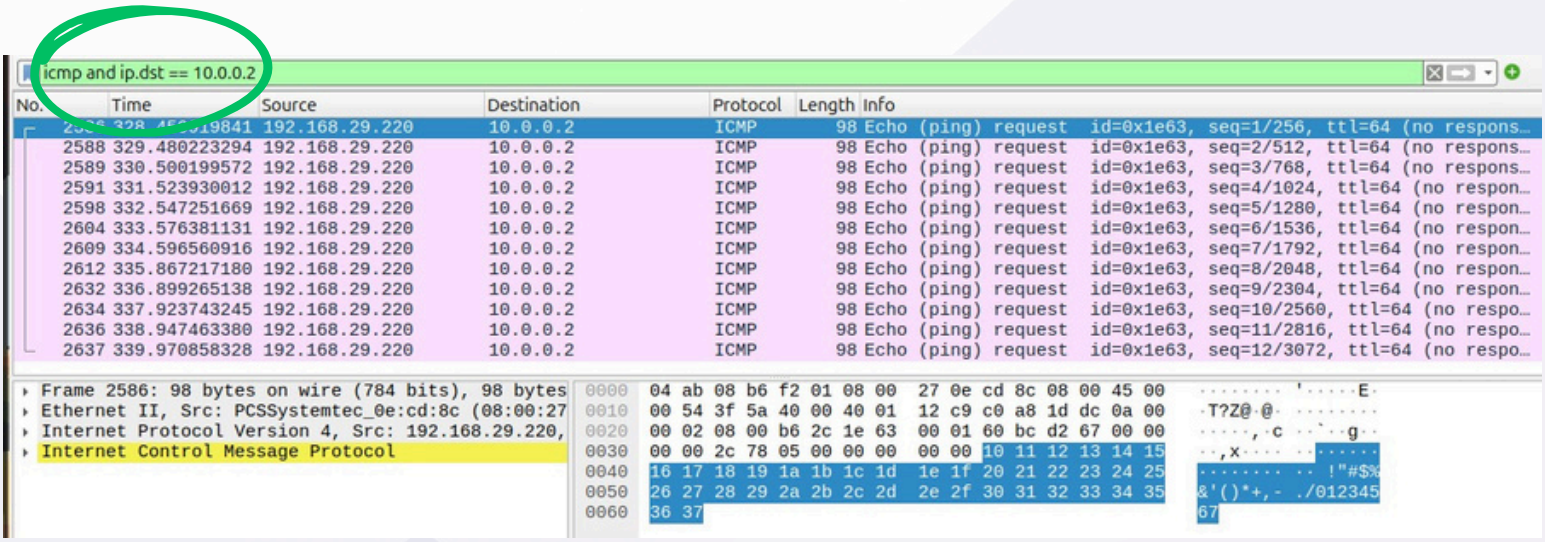
- I can now connect to the Web Server
- Google.com / 8.8.8.8

Finally it is successful!

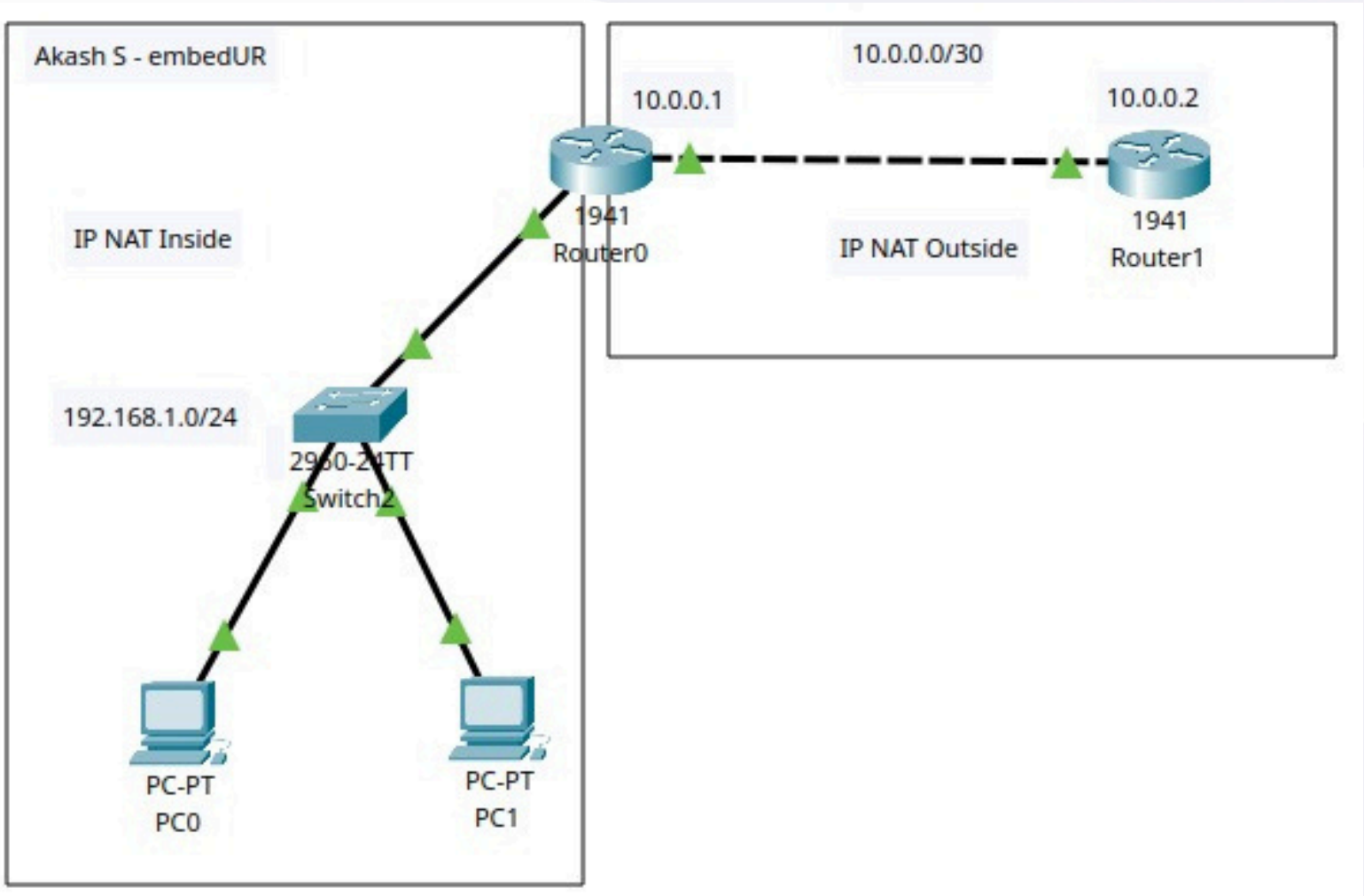
Fire	Last Status	Source	Destination	Type	Color	Time(sec)
	Successful	PC0	Router1	ICMP		0.000
	Successful	PC1	Router1	ICMP		0.000
	Successful	PC0	Router0	ICMP		0.000
	Successful	PC1	Router0	ICMP		0.000







FINAL SUCCESSFUL NETWORK:



NAT inside and NAT outside

THE END

AKASH S