NETWORKING TRAINING - MODULE 5

Akash S, embedUR Systems

Task 1:

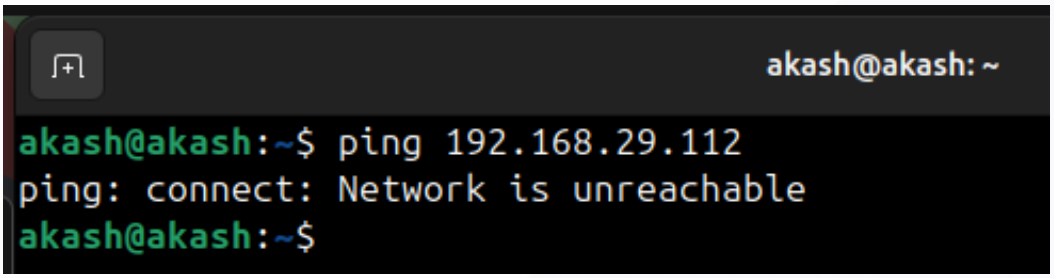
Capture and analyze ARP packets using Wireshark. Inspect the ARP request and reply frames, and discuss the role of the sender's IP and MAC address in these packets.

Explanation:

- 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**

Attempt 1:

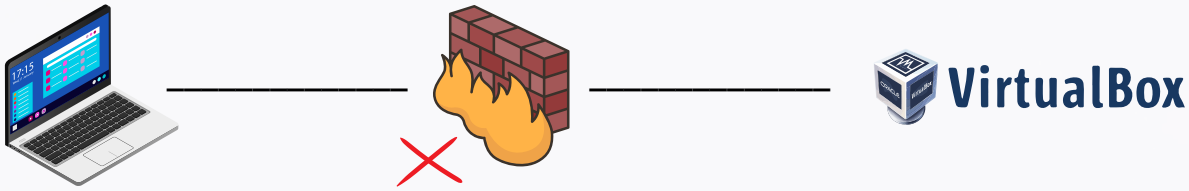
- When I pinged from my VM to Windows, it showed **“Not reachable”**
- I checked solving the issue by several ways!



It failed!

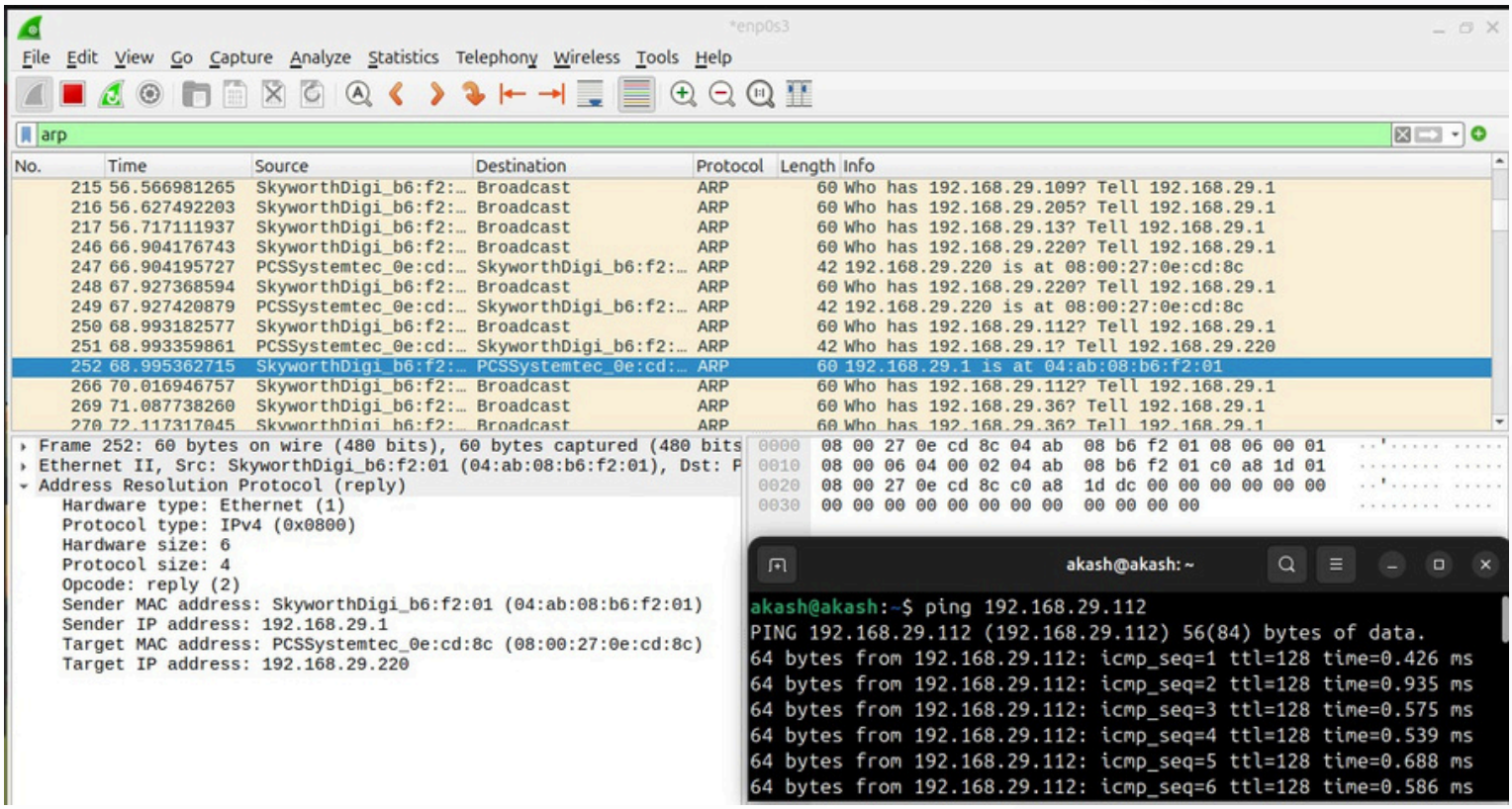
Attempt 2:

- I finally found why my network is not reachable.
- It is because of the firewall settings of my Windows which is blocking the ICMP requests



Attempt 3:

- I allowed the incoming ICMP requests by configuring firewall **rules**
- Enabled the blocking rules
- Tried pinging again, it worked!



Successful!

What I learnt?

- How arp works
- IP and MAC address relationship
- If two systems want to communicate, then we always have to check for the firewall if any rules are blocked.

Task 2:

Using Packet Tracer, simulate an ARP spoofing attack. Analyze the behavior of devices on the network when they receive a malicious ARP response.

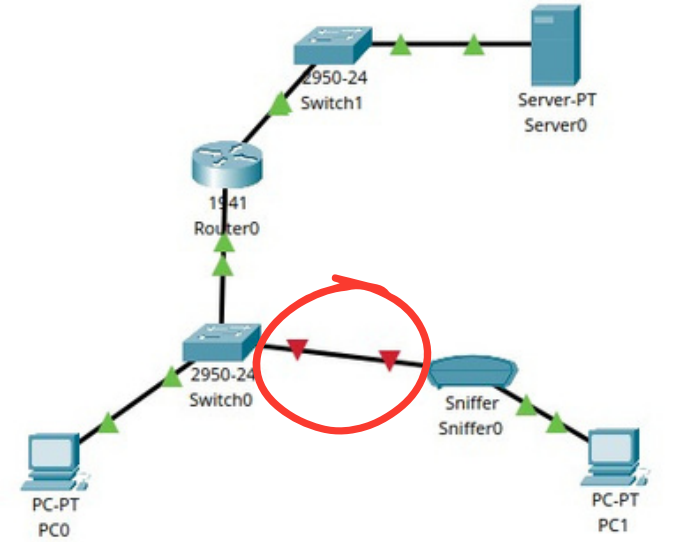
Trial & Errors:

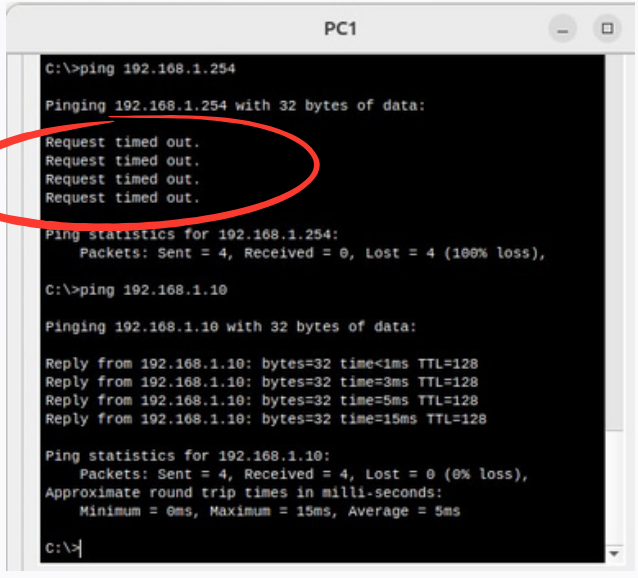
Trial 1:

- I started by connected two PCs and a switch
    - PC1: **192.168.1.10**
    - PC2: **192.168.1.20**
- I added a network sniffer between the switch and the PC
- I added a second switch, this time it is for a different network and I added a Web Server to it - **192.168.2.100**
- I added a Generic Router (1941) to connect both the networks of different subnet

So what's the problem?

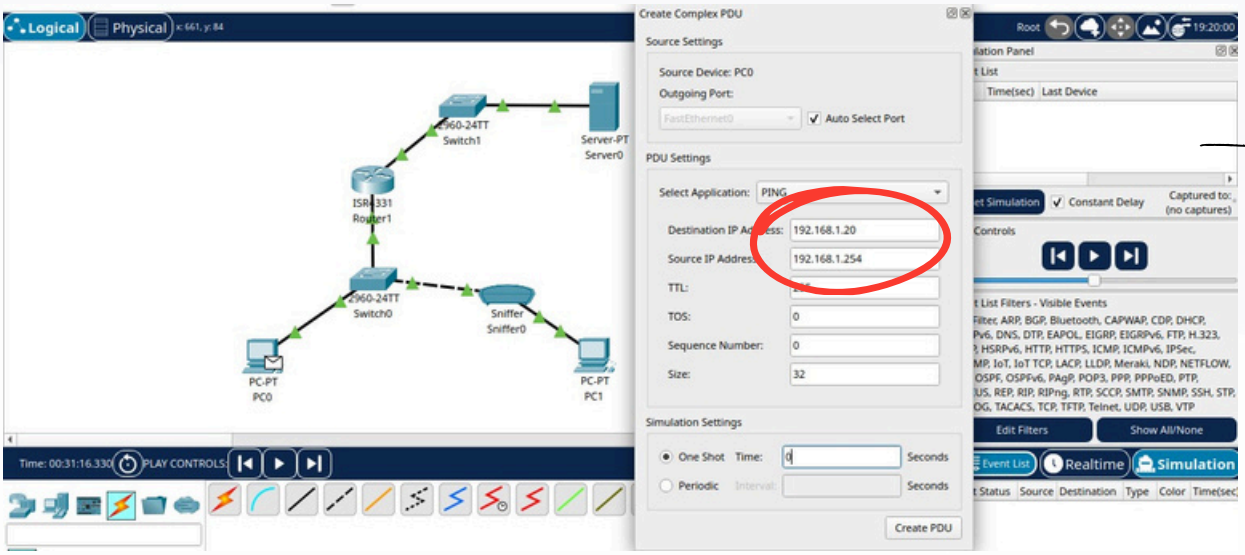
- The sniffer that is connected to switch is not working.
- I took several steps to make the connection UP, but nothing worked.
- What happened??
    - Oh, is it because the sniffer's package is not installed?
        - I checked, no issues, it is installed properly
    - Ok, is the firewall blocking the access?
        - No, everything is perfect!
    - Any problem with the interface?
        - No, they are on different ports, i don't think so!





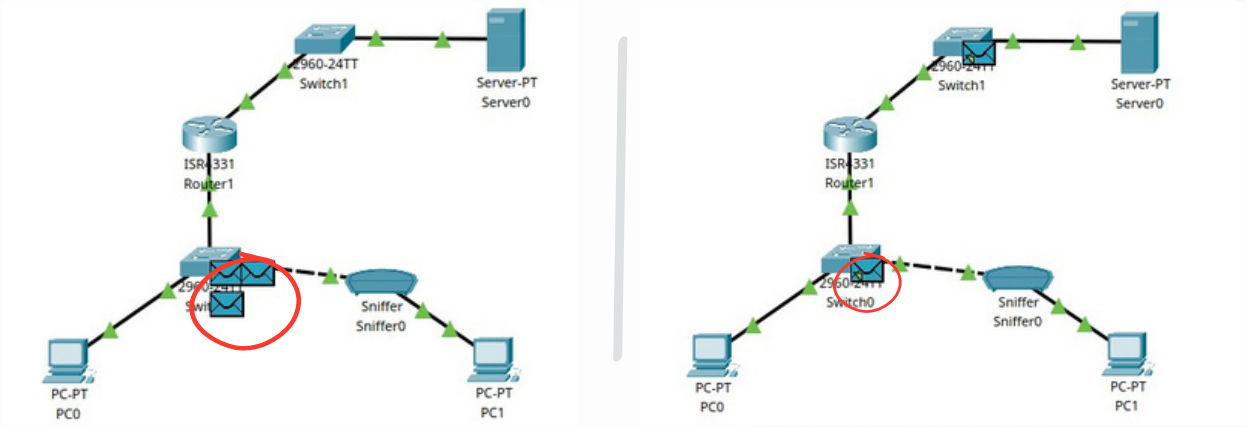
Trial 2:

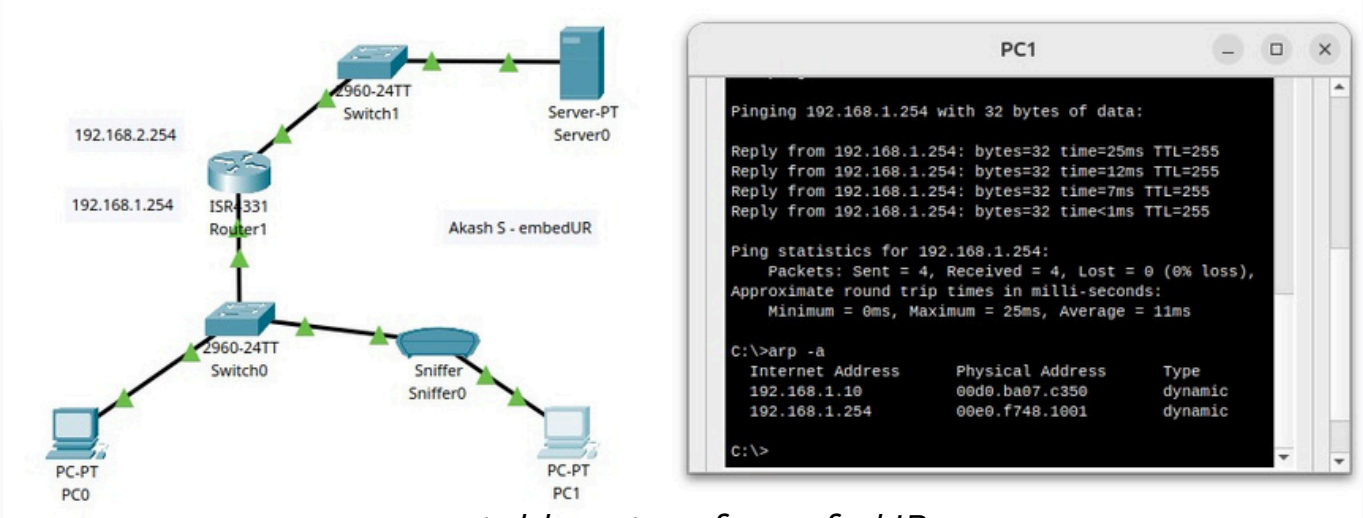
- Let me try deleting the interface and adding a proper one.
- Last time I left the Default Gateway as default - **0.0.0.0**
- But this time, I have given the router's address as default gateway



→ Dest: PC1 - **192.168.1.20**
Src: PC2 - **192.168.1.10**
but spoofed as **192.168.1.254**

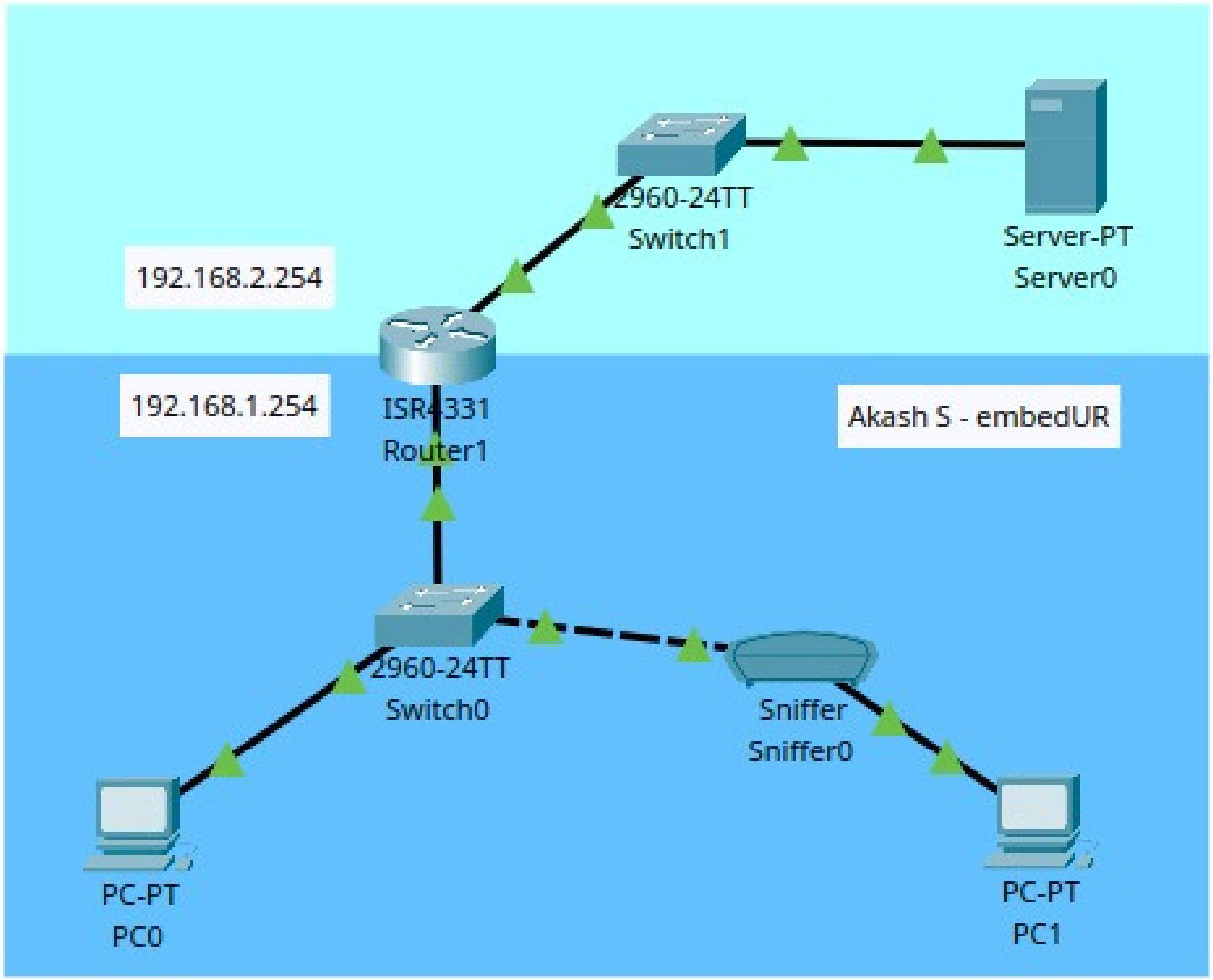
- Here the attacker spoofs the Destination with Router's IP instead of its own IP.
- So, PC0 thinks that the packets are sent from the Router but it is actually not!





arp table entry of spoofed IP

FINAL OUTPUT: 😍



Two networks connected through a router

Learnings from errors:

- Check whether everything is configured before simulating
- Enable the ports for the router when you add - **ON**
- Ping at while you connect so that you won't end up deleting the whole network at the last
- Once I knew there was a problem, I connected one device at a time, pinged and then connected the other one and it worked!

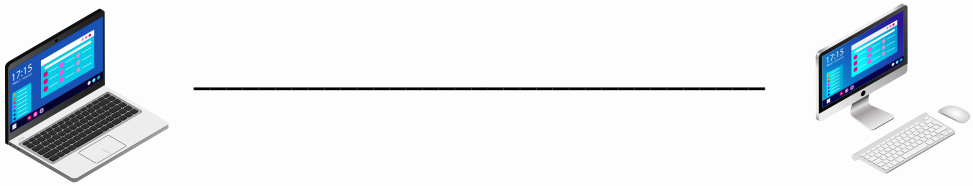
Task 3:

Manually configure static IPs on the client devices(like Pc or your mobile phone) and verify connectivity using ping.

Explanation:

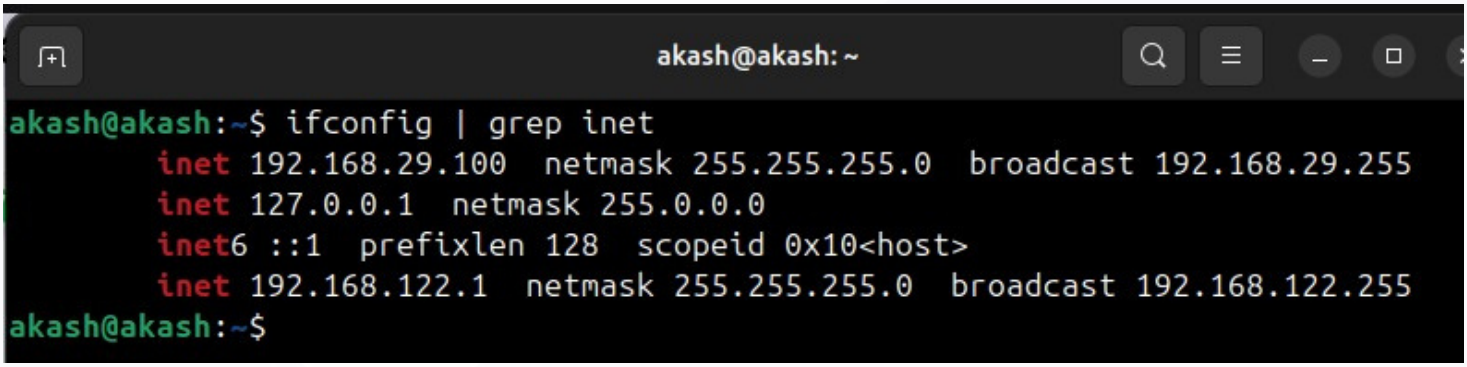
Static IP:

- Created a clone of a Virtual Machine for this task - **embedUR VM**
- First, the interface is found using the command
  $ **ifconfig**
- In my case, the interface is **enp0s3.**
- I assigned a static IP - **192.168.1.100**
- Once it is set, we can confirm the static IP using the command,
  $ **ifconfig enp0s3 | grep inet**

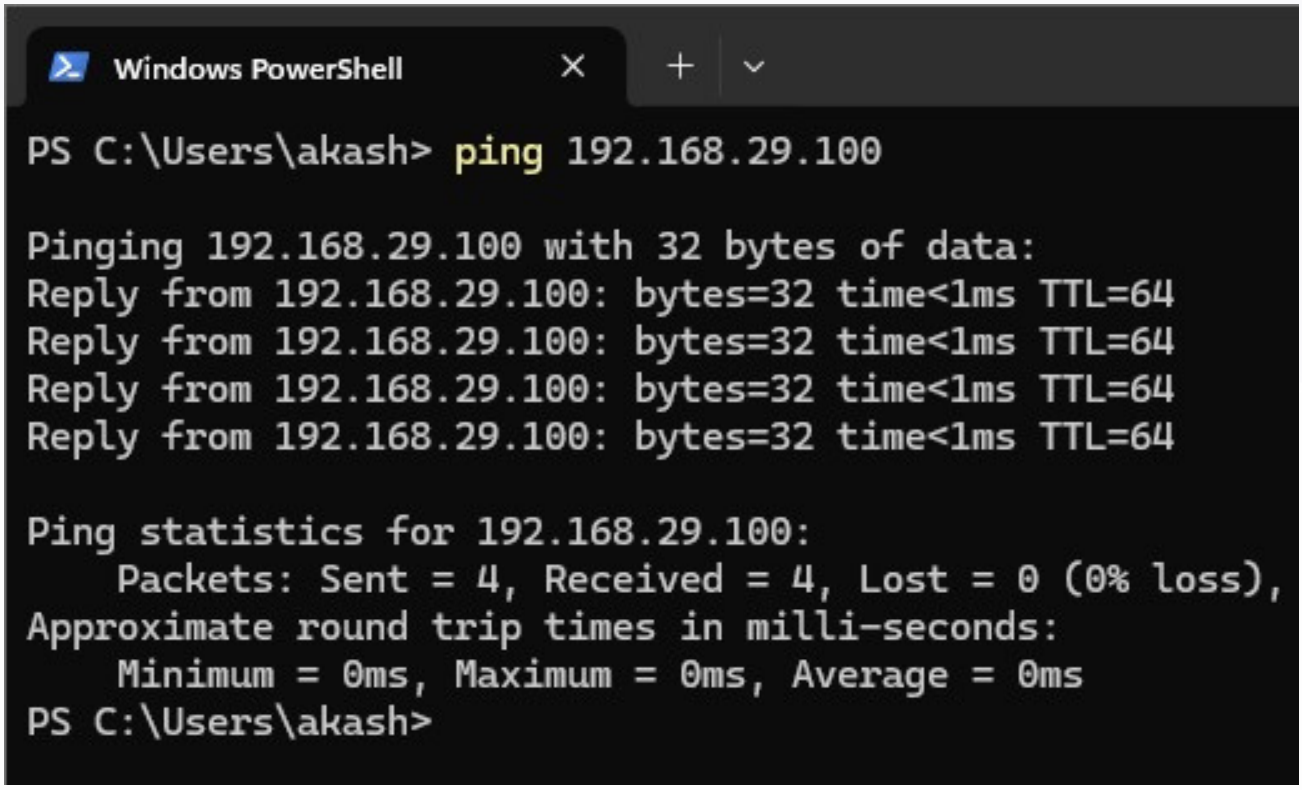


Windows Client Machine

Checking whether Static IP is set properly:

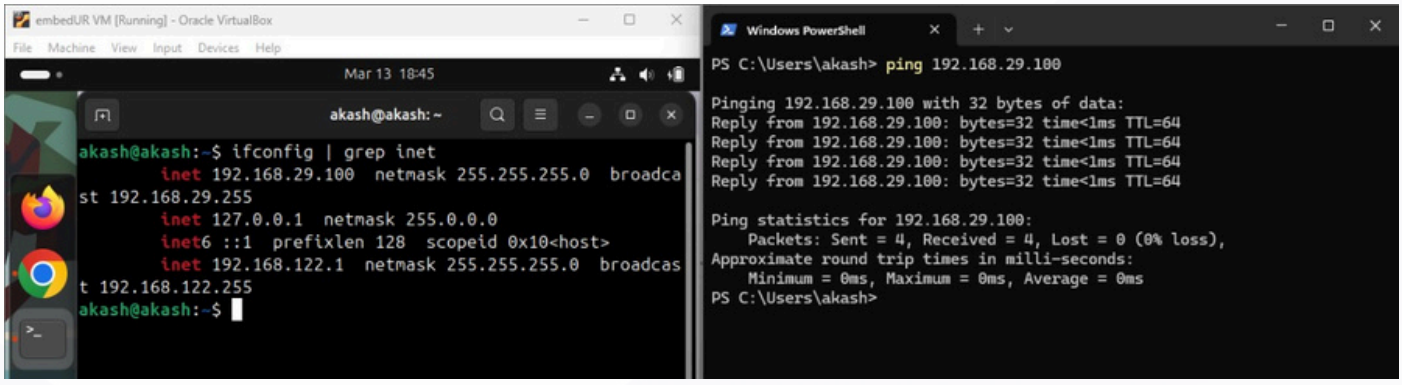


Outputs:



Checking connectivity through ping

Combined:



Client (Virtual box)

Windows (host)

Task 4:

Use Wireshark to capture DHCP Discover, Offer, Request, and Acknowledge messages and explain the process.

Explanation:

DHCP:

- Configured the VM to DHCP mode to facilitate the DORA process
- DHCP dynamically assigned an IP to my VM
    - **192.168.29.166**
- Ensured that DHCP is active
- Now captured the packets using Wireshark in **enp0s3** interface
- I triggered DHCP using,
    - sudo nmcli con down "netplan-enp0s3"
    - sudo nmcli con up "netplan-enp0s3"
- Filtered only **dhcp** in wireshark search

Attempt 1:

- I only saw Request and Acknowledgement in Wireshark
- What happened?
    - Since the VM already had an IP **192.168.29.220,** DHCP Discover and Offer didn’t work
- How did I solve?
    - I flushed the IP
    - This time the VM has only a single IP that DHCP assigned
    - Now triggered again.

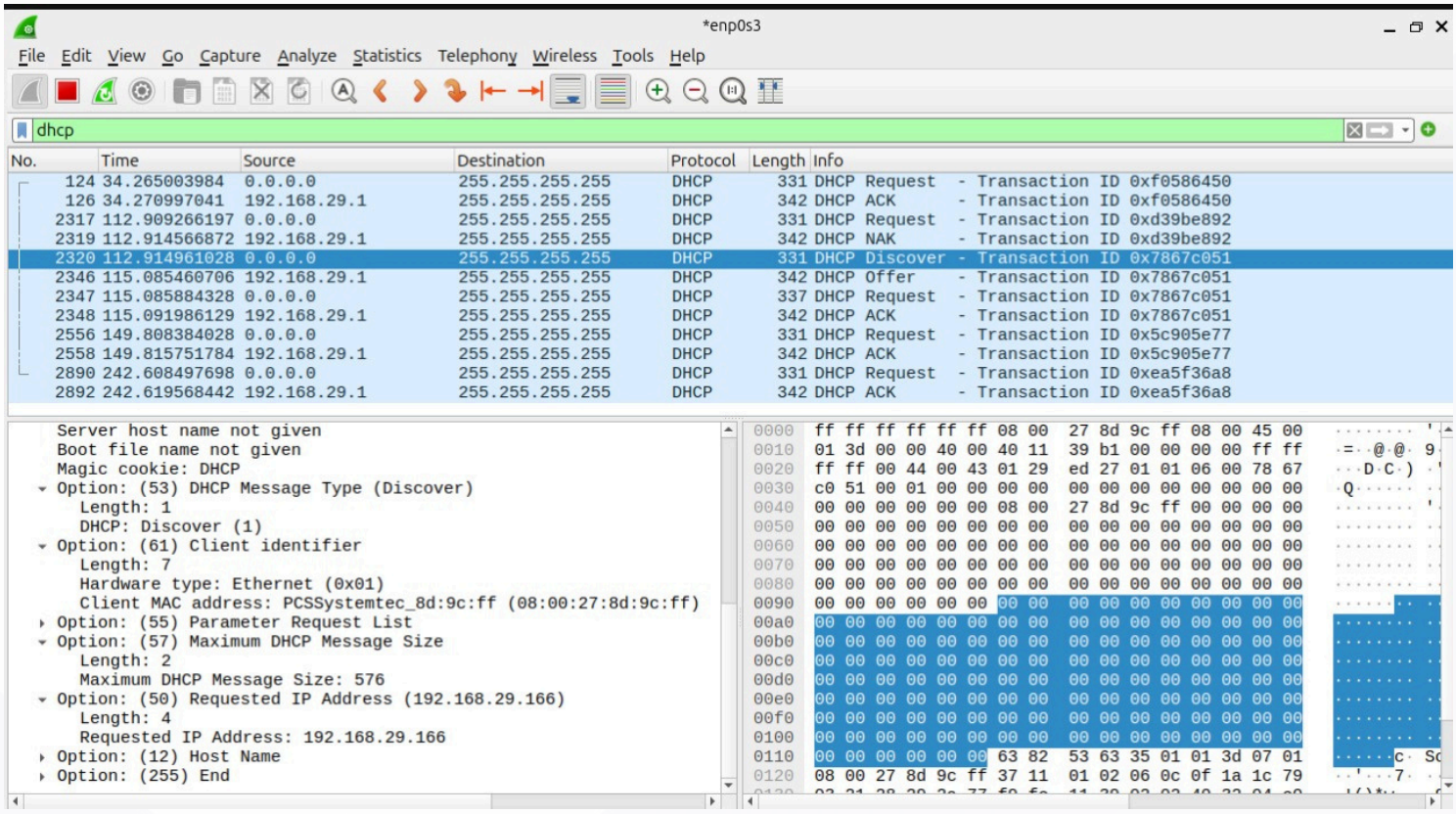
Attempt 2:

- When I filtered **dhcp** packets alone in Wireshark, it worked.
- I identified all packets like Discover, Offer, Request, Acknowledgement

All DHCP Messages:

	Protocol	Length	Info
255.255	DHCP	331	DHCP Request - Transaction ID 0xf0586450
255.255	DHCP	342	DHCP ACK - Transaction ID 0xf0586450
255.255	DHCP	331	DHCP Request - Transaction ID 0xd39be892
255.255	DHCP	342	DHCP NAK - Transaction ID 0xd39be892
255.255	DHCP	331	DHCP Discover - Transaction ID 0x7867c051
255.255	DHCP	342	DHCP Offer - Transaction ID 0x7867c051
255.255	DHCP	337	DHCP Request - Transaction ID 0x7867c051
255.255	DHCP	342	DHCP ACK - Transaction ID 0x7867c051
255.255	DHCP	331	DHCP Request - Transaction ID 0x5c905e77
255.255	DHCP	342	DHCP ACK - Transaction ID 0x5c905e77
255.255	DHCP	331	DHCP Request - Transaction ID 0xea5f36a8
255.255	DHCP	342	DHCP ACK - Transaction ID 0xea5f36a8

Detailed Capture:



Complete capture in wireshark

Task 5:

Given an IP address range of 192.168.1.0/24, divide the network into 4 subnets.
Task: Manually calculate the new subnet mask and the range of valid IP addresses for each subnet.
Assign IP addresses from these subnets to devices in Cisco Packet Tracer and verify connectivity using ping between them.

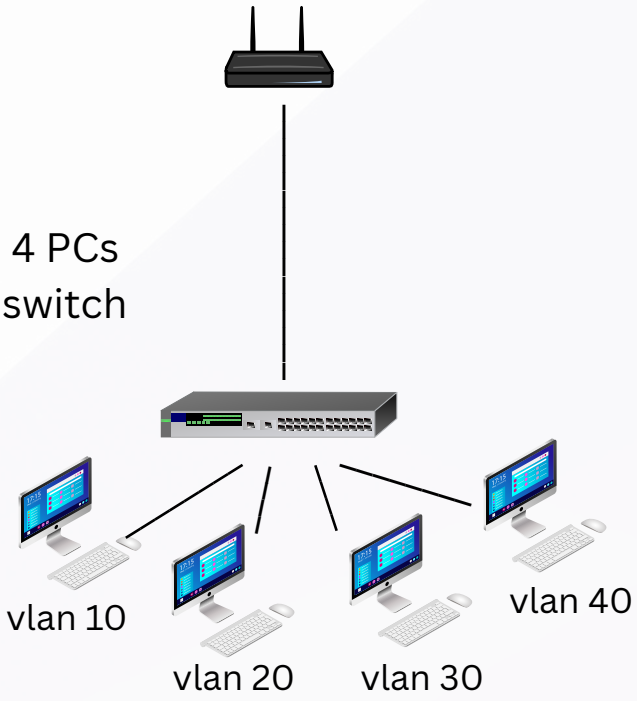
Calculation:

- Original: /24 (255.255.255.0) = 256 addresses.
- Need 4 subnets: $2^2 = 4$, so borrow 2 bits from the host portion.
- New subnet mask: /24 + 2 = /26 (255.255.255.192).
- Addresses per subnet: 256 ÷ 4 = 64 (2^6).

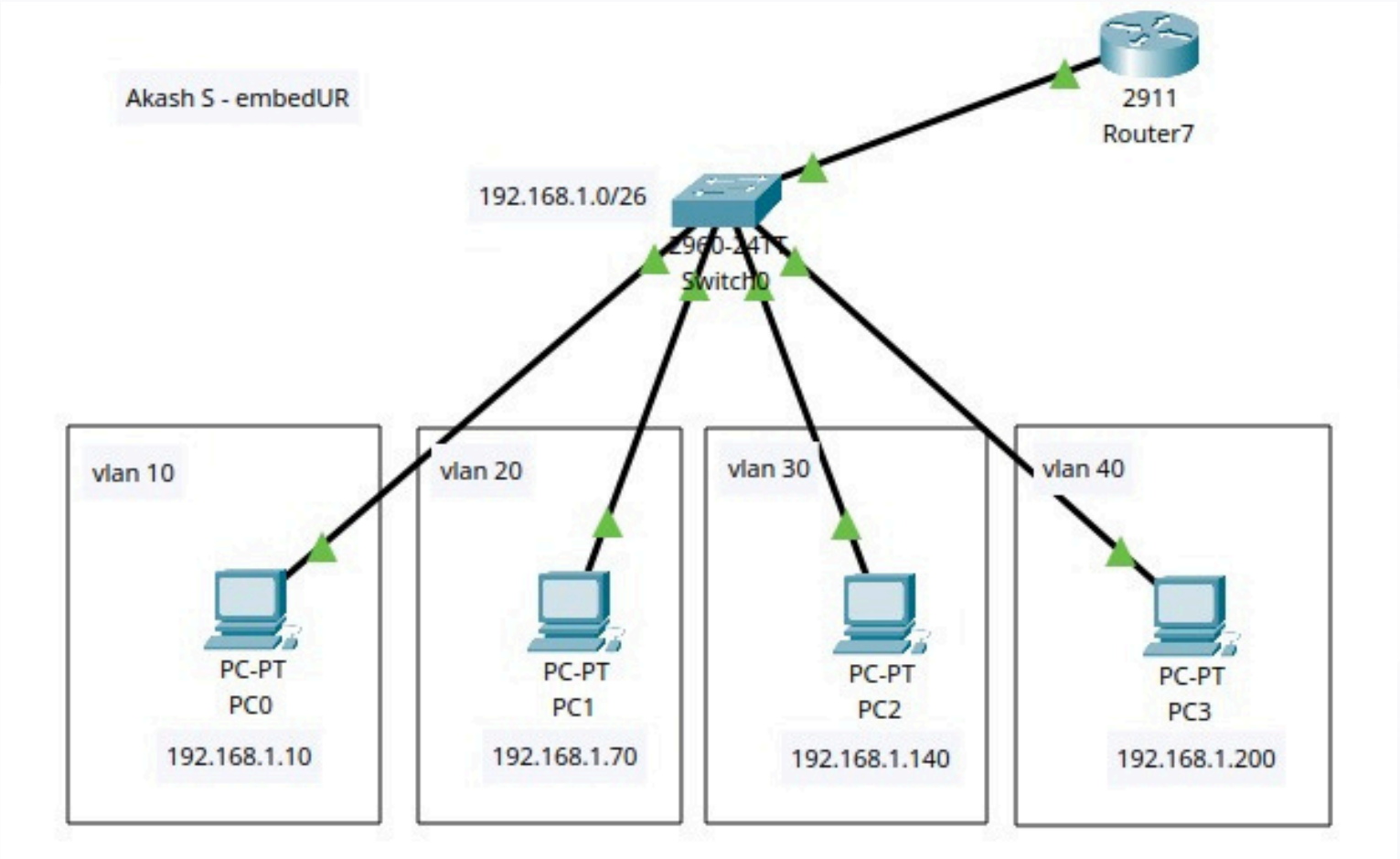
So we need to add **64** to configure new address

Practical Network:

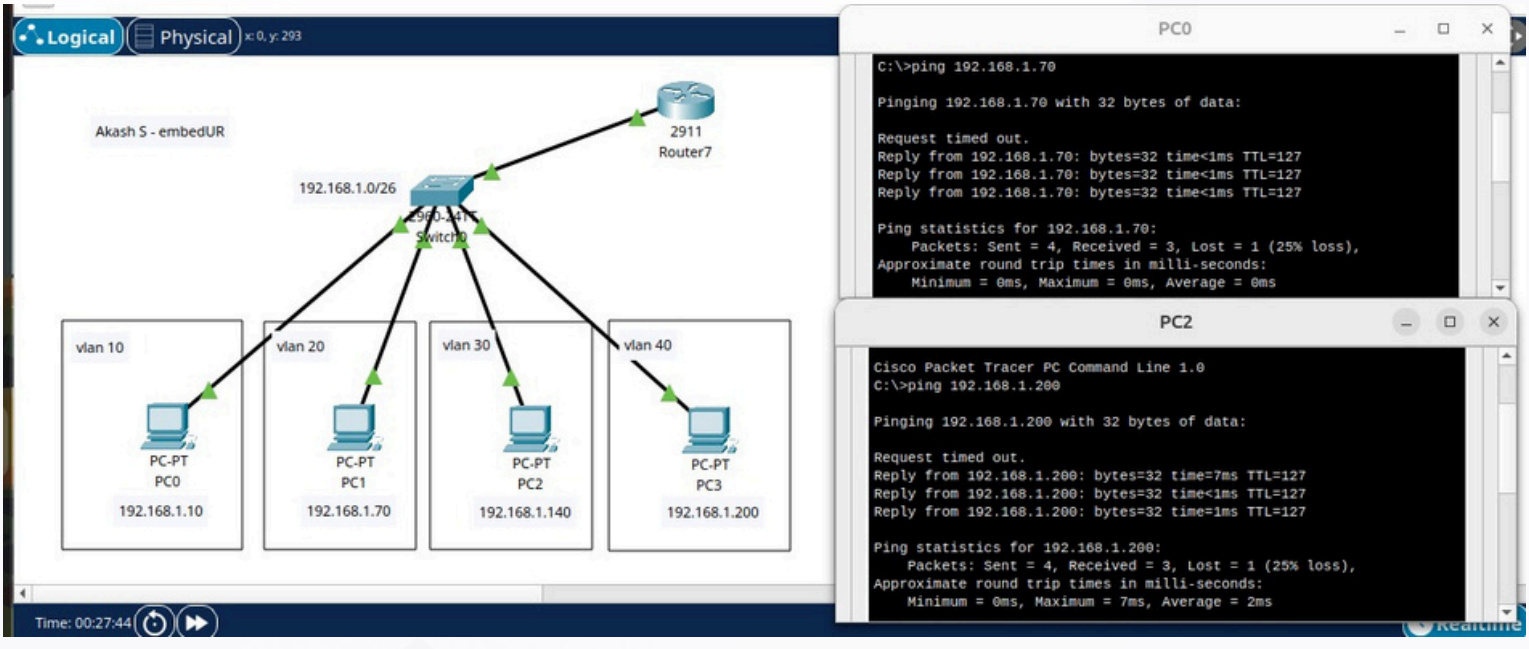
- I added 4 PCs,
    - PC1: **192.168.1.10**
    - PC2: **192.168.1.70**
    - PC3: **192.168.1.140**
    - PC4: **192.168.1.200**
- I added a switch to connect to all these 4 PCs
- I added a router which connects to the switch
- I configured each PC as a VLAN
    - PC1: **vlan 10**
    - PC2: **vlan 20**
    - PC3: **vlan 30**
    - PC4: **vlan 40**
- I tested all the VLANs connectivity by randomly pinging a VLAN device



Outputs:



network topology



Connectivity test

Task 6:

Describe how you would configure a basic LAN interface using the ip command in Linux (kernel.org).

Documentation:

Identification:

1. IP Address: 10.1.1.1
   Class: A
   Default Subnet Mask: 255.0.0.0
   IP Range: 10.0.0.0 – 10.255.255.255

2. IP Address: 172.16.5.10
   Class: B
   Default Subnet Mask: 255.255.0.0
   IP Range: 172.16.0.0 – 172.31.255.255

3. IP Address: 192.168.1.5
   Class: C
   Default Subnet Mask: 255.255.255.0
   IP Range: 192.168.0.0 – 192.168.255.255

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

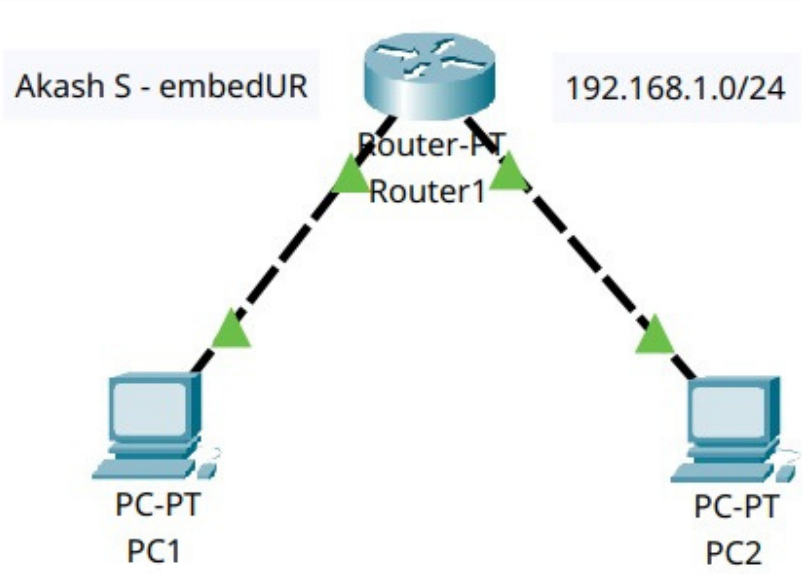
Task 7:

In Cisco Packet Tracer, create a small network with multiple devices (e.g., 2 PCs and a router). Use private IP addresses (e.g., 192.168.1.x) on the PCs and configure the router to perform NAT to allow the PCs to access the internet.
Task: Test the NAT configuration by pinging an external IP address from the PCs and capture the traffic using Wireshark.
What is the source IP address before and after NAT?

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!

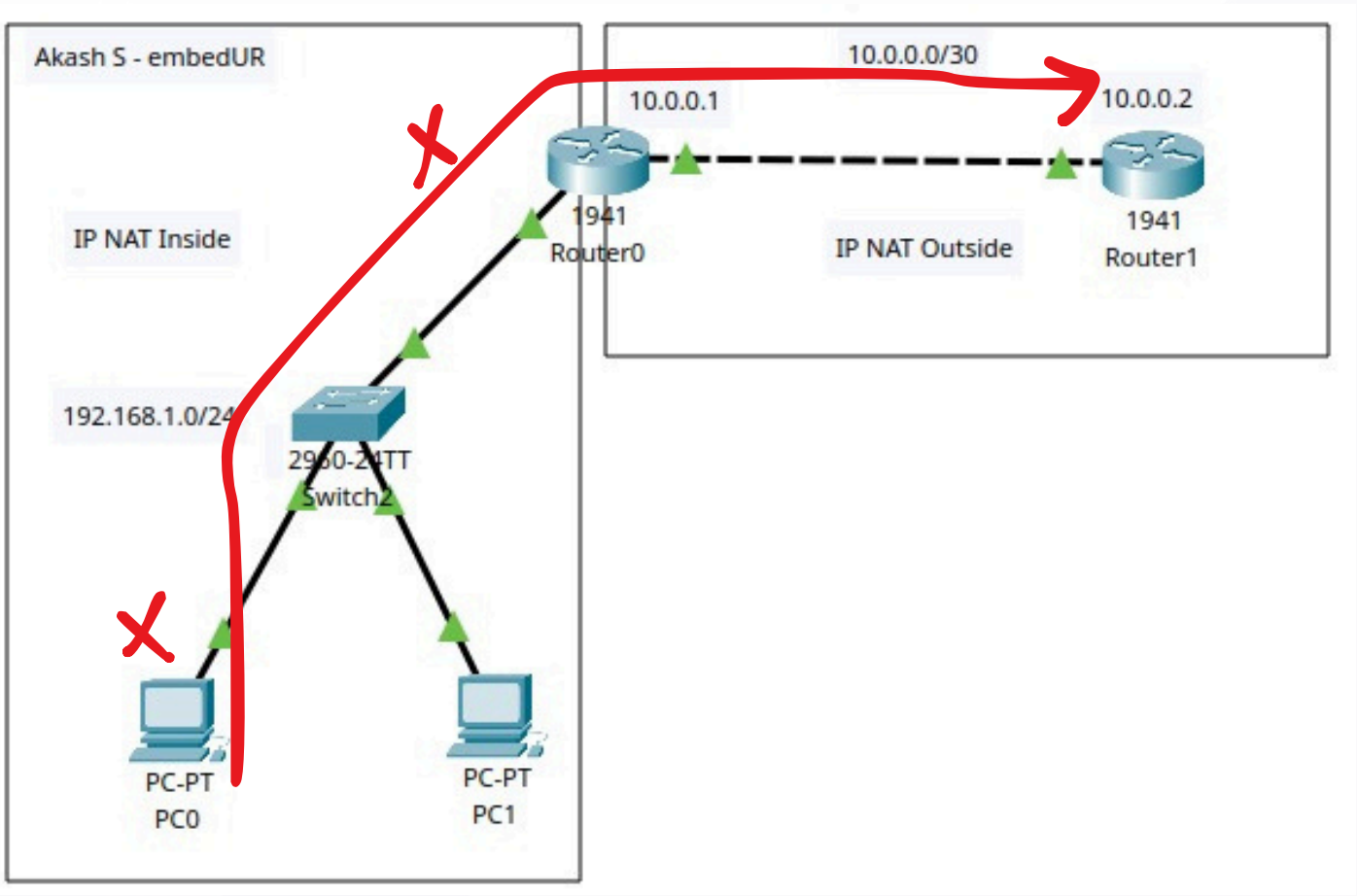
Diving into the actual network:

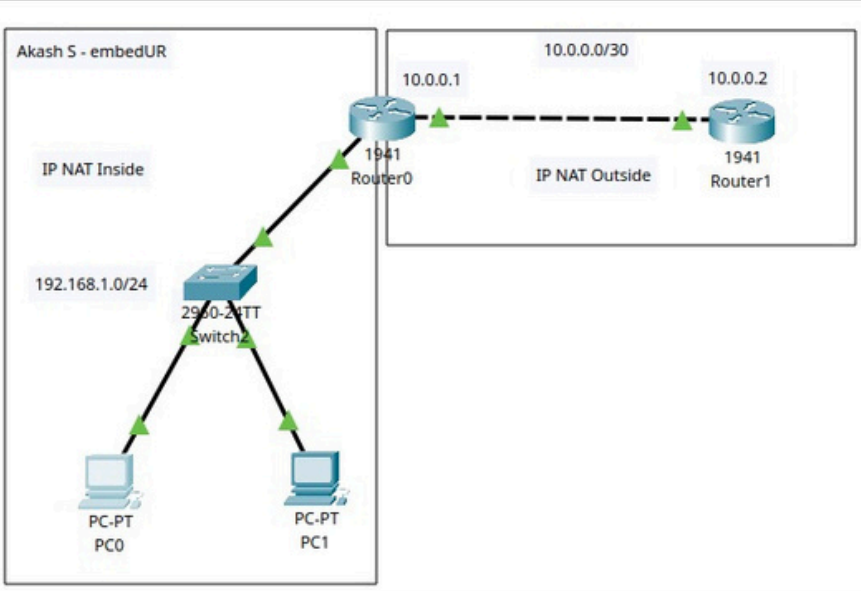
Attempt 2:

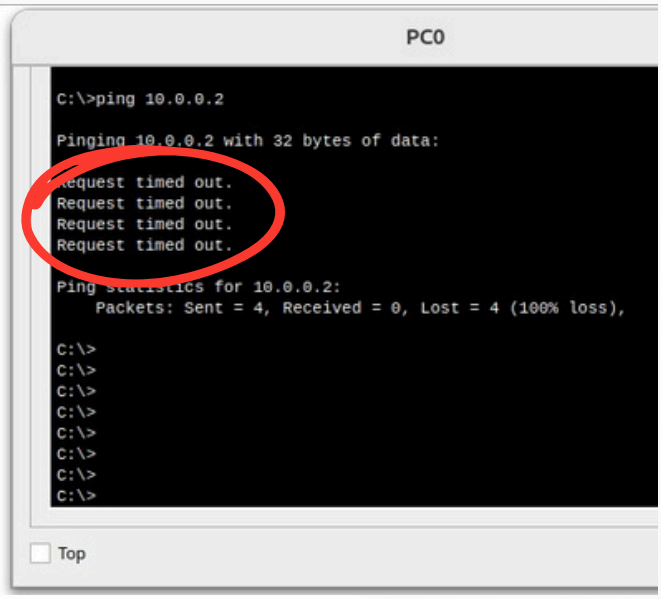
- I created two NAT networks
    - IP NAT Inside
    - IP NAT Outside
- In IP NAT Inside, I have connected 2 VPCs, a switch and a Router
    - PC1: **192.168.10.10**
    - PC2: **192.168.10.20**
    - Router1: **10.0.0.1**
- In IP NAT Outside, I have connected a router (external)
    - Router2: **10.0.0.2**
- 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

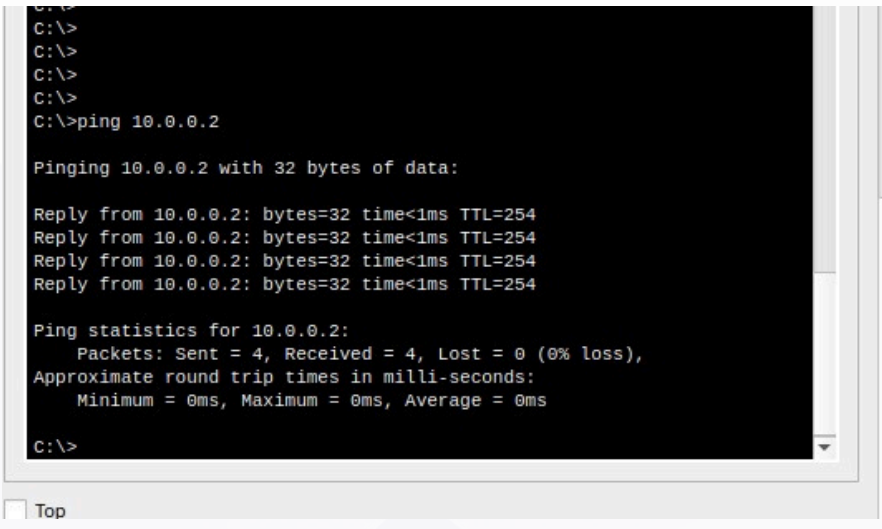
Attempt 4:

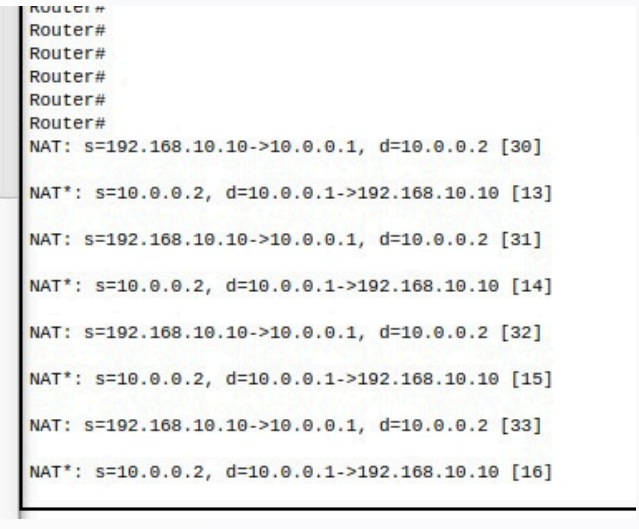
- Now I can’t reach Router 2 from **PC1**
- I tried debugging through Router’s CLI and PC1's Command prompt
- Configured
    - **IP NAT Inside**
    - **IP NAT Outside**

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



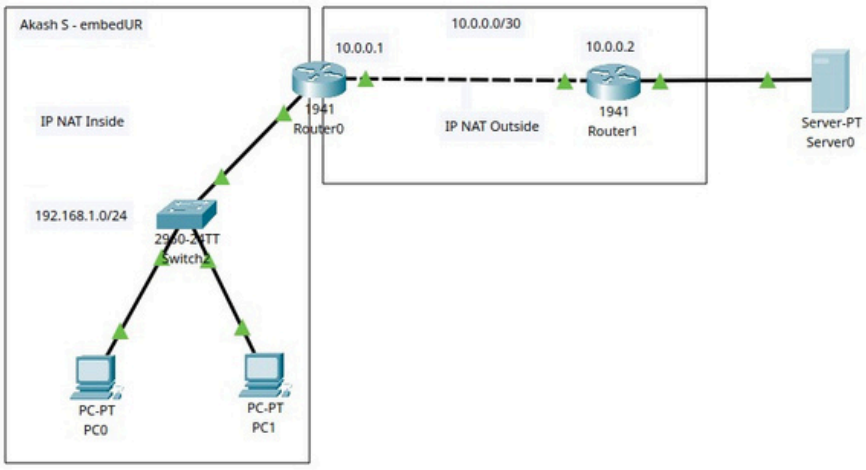


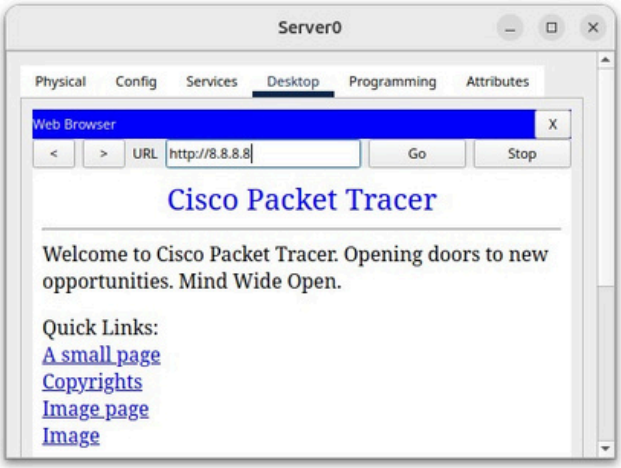
Final Output:

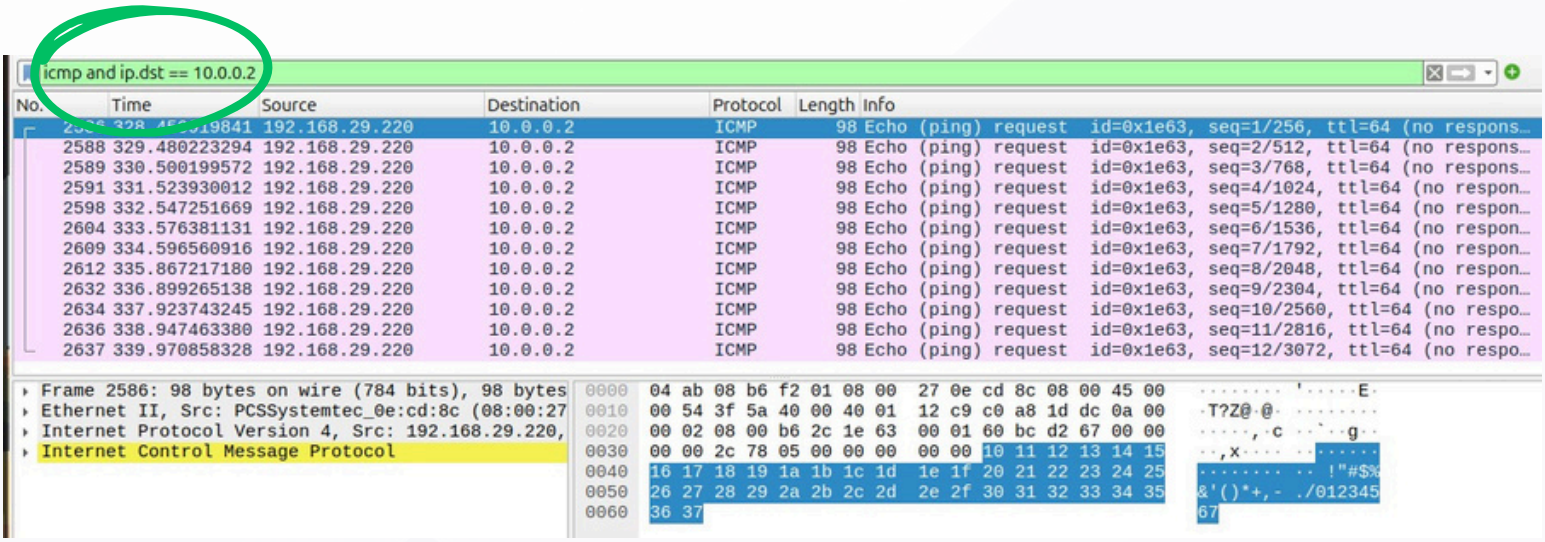
- I can now connect to the Web Server
- Google.com / 8.8.8.8

Finally it is successful!

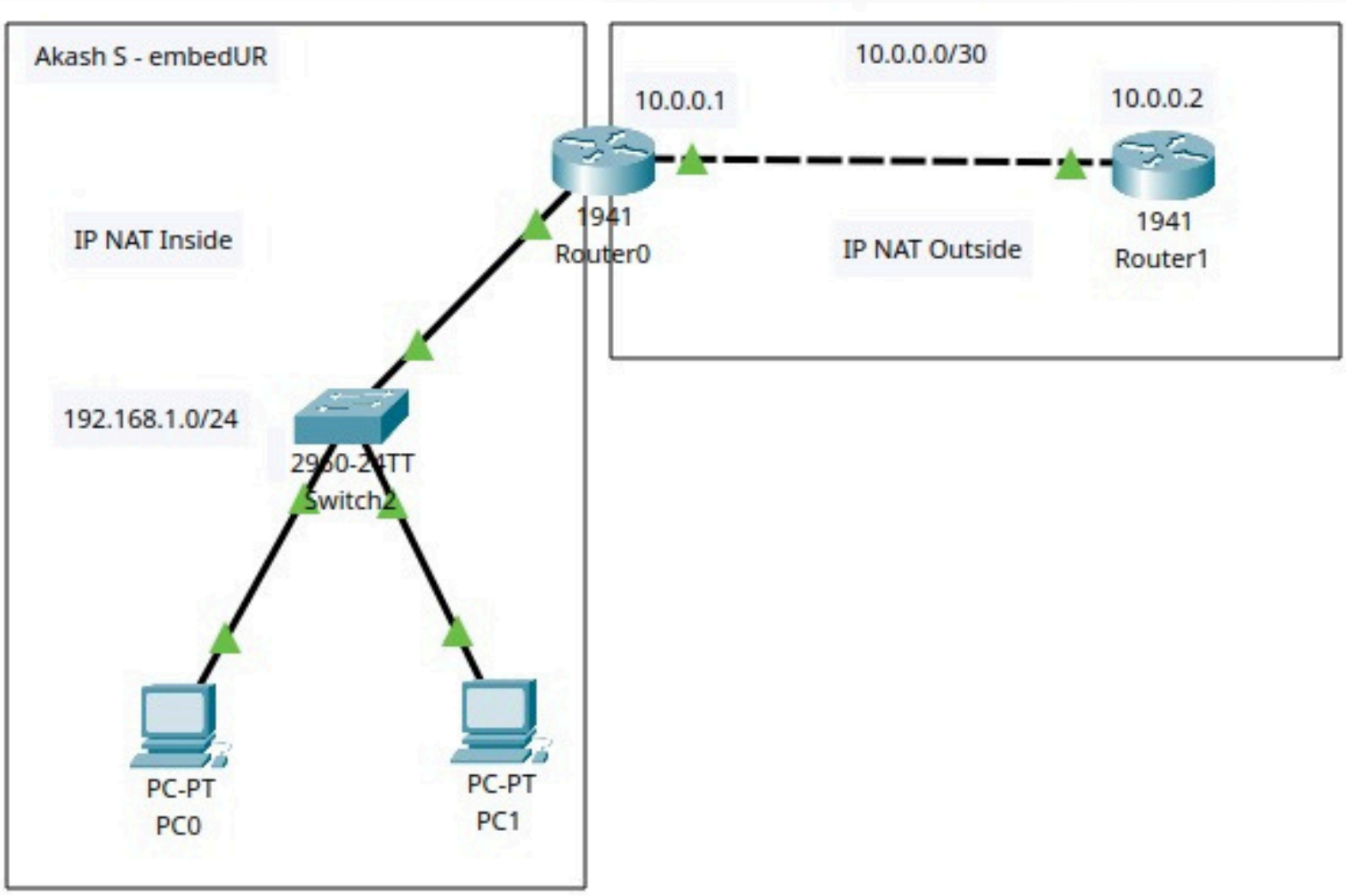
Fire	Last Status	Source	Destination	Type	Color	Time(sec)
	Successful	PC0	Router1	ICMP		0.000
	Successful	PC1	Router1	ICMP		0.000
	Successful	PC0	Router0	ICMP		0.000
	Successful	PC1	Router0	ICMP		0.000







FINAL SUCCESSFUL NETWORK:



NAT inside and NAT outside

THE END

AKASH S