BIRZEIT UNIVERSITY

Faculty of Engineering and Technology

Electrical & Computer Engineering Department

Computer networks ENCS3320

Project #2

Prepared by:

Name: mahmoud nobani **Id:**1180729

Instructor: Dr. Abdelkarim awad

Section: 1

Date:6/30/2023

Part1:

- DHCP (dynamic host configuration protocol): is a protocol that distributes IPs within a network, as when a device connects onto a network it asks an IP from the DHCP server and then DHCP assigns one to it, this whole procedure simplify the of IP assignment.
- DNS (Domain name system): it aims to map domain names into an IP machine can understand, thus allowing people to access website using easy to remember names instead of the machine IPs.
- ICMP (internet control message protocol): it’s used for diagnoses and feedbacks as it provides networks a way to exchange control or error messages.

---

1) Sniffing DHCP packets:

First, we have to perform two commands on the CMD which are:

- Ipconfig /release: which release the current IP address from the network
- Ipconfig /renew: which request a new IP address from the DHCP server.

Wireshark was able to sniff the following packets:

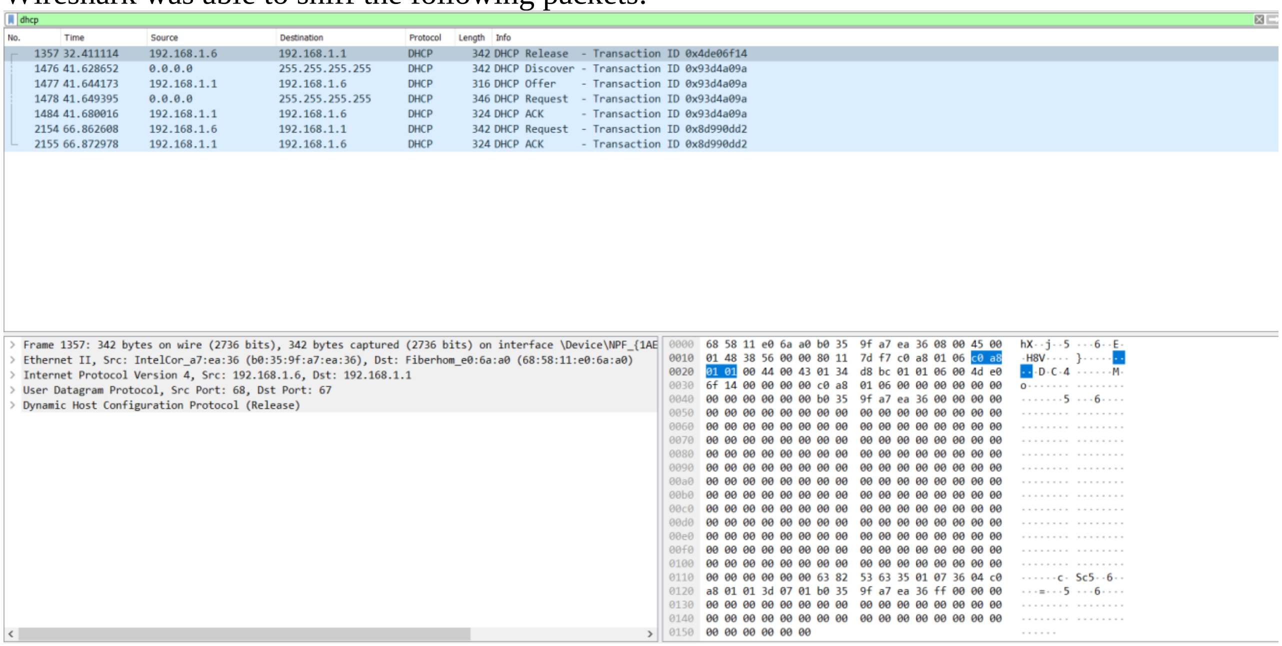

Figure 1: sniff DHCP.

As we can see we have 7 packets but the most notable ones are the first one (DHCP release) and the 4 and 6 ones (DHCP request) in which the IP address was given.

Looking at the first packet more, we can see that we have the following info about it:

The source: 192.168.1.6 which is my IP

The destination: 192.168.1.1 which is the DHCP IP

The length: 342 which is the number if bits on the datagram message shown

Protocol: which is the protocol used (DHCP)

Info: which specify the type of command

Another interesting property we can point out Is the User Datagram protocol (UDP) source port 68 and destination 67 as DHCP uses the UDP protocol.

2) Sniffing ICMP:

Using the command ping google.com, I got the following result:

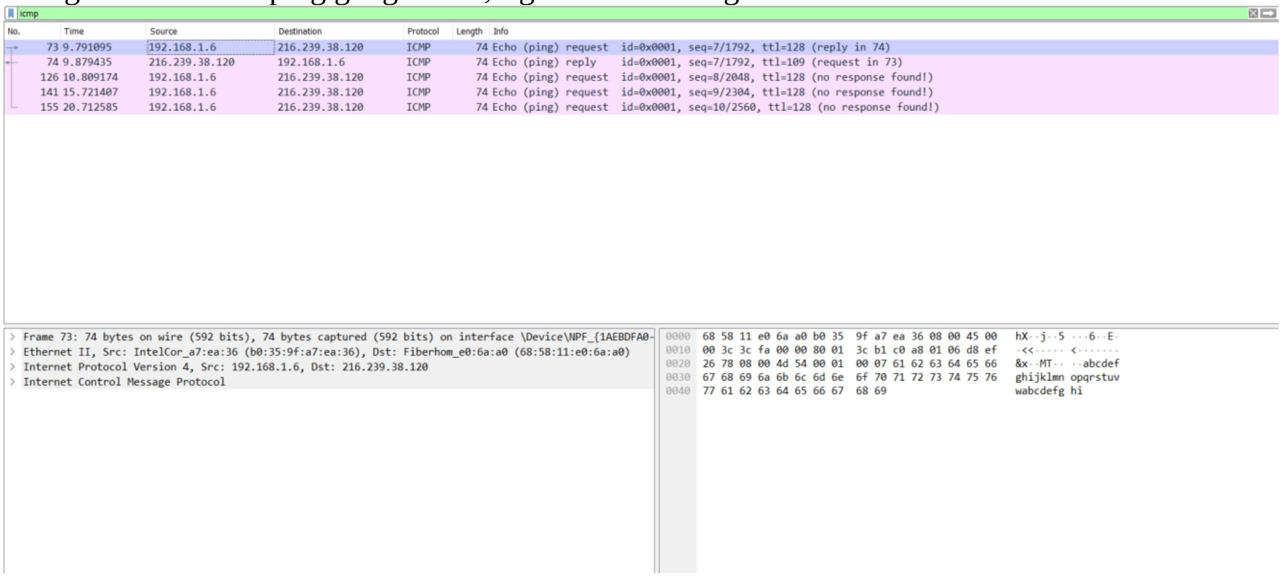

Figure 2: sniff ICMP.

As we can see, we have 5 packets, the 4 request ping commands usually make, and only one replay, now if we look as the CMD output:

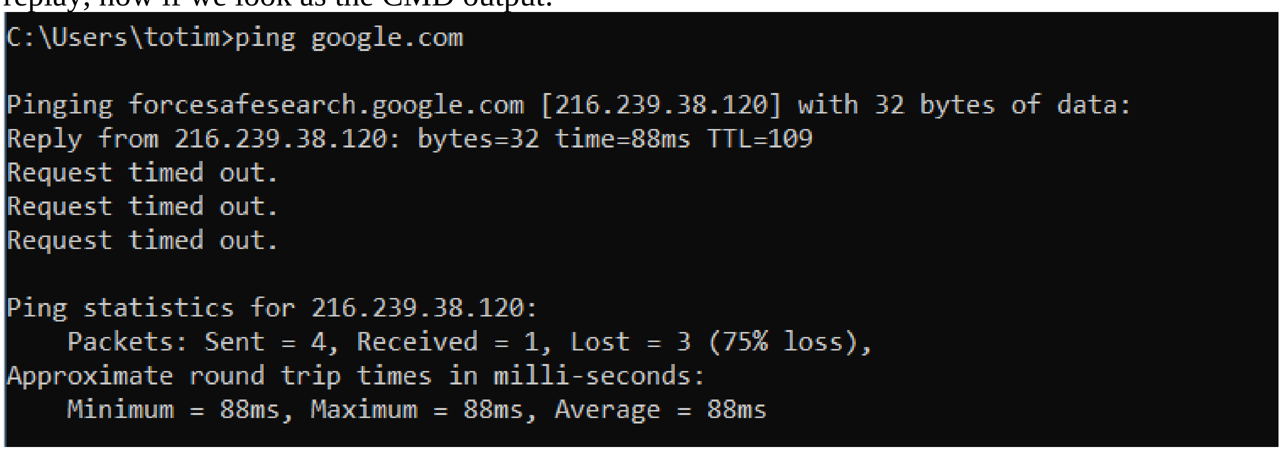

Figure 3: ping google.com

We can see that three requests were timed out; thus, it makes sense to have only 5 packets.
Now going back to figure 2, lets analyze the first packet more, as it has some interesting fields:
The most notable ones are:
Source: 192.168.1.6 which is my IP
Destination: 216.239.38.210 which is google.com IP address
The length was 74 and protocol used is IMCP, with the info specifying the command as request.

3) Sniffing DNS:

To sniff DNS, we used nslookup google.com and got the result below:

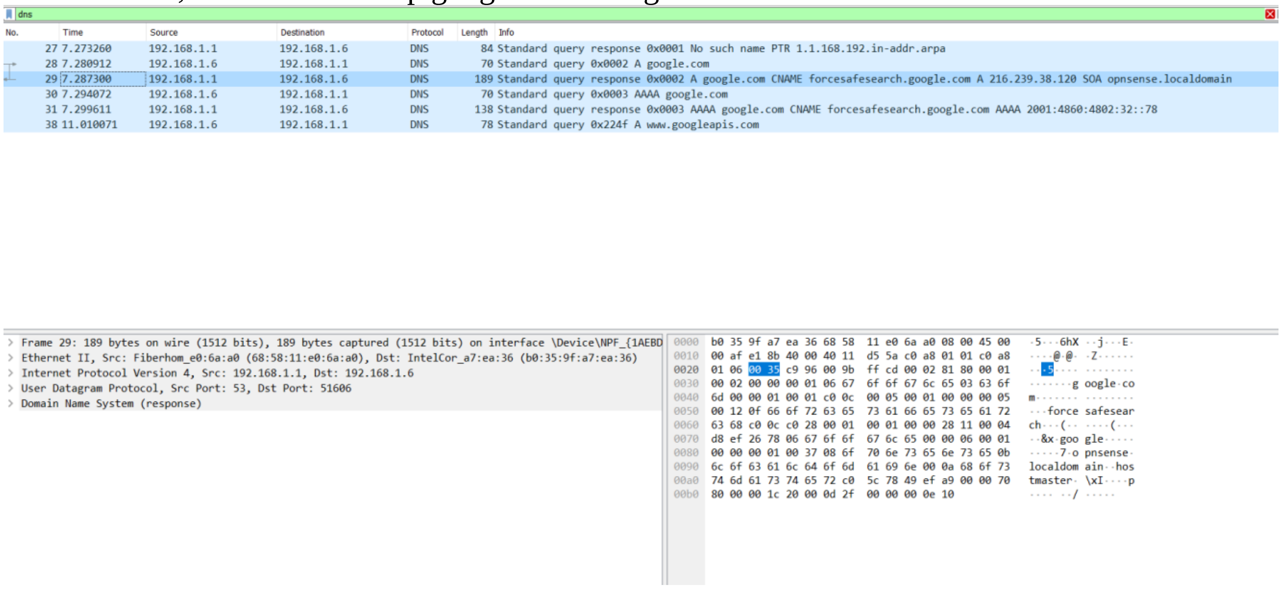

Figure 4:sniff DNS

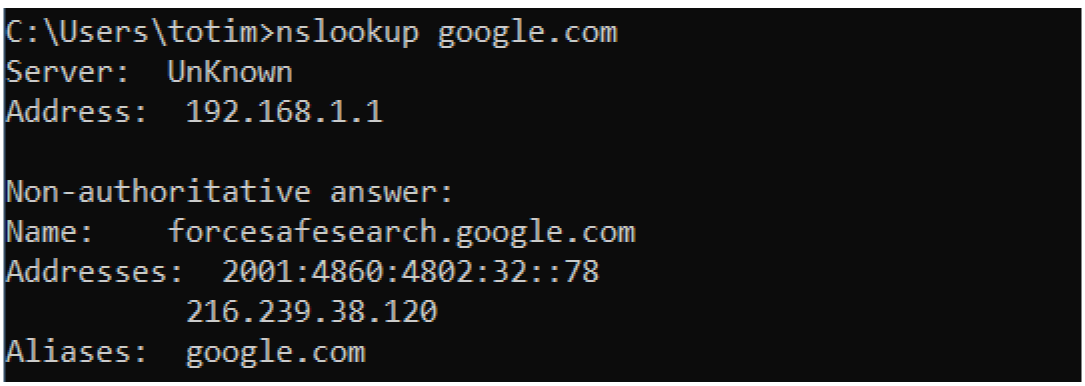

Figure 5: nslookup google.com

Now looking at figure one, the most interesting packet for me is the third, first it has the typical stuff a source of 192.168.1.1 which is the home IP, as it sends to my IP the destination IP 192.168.1.6, and info of:

"Standard query response 0x0002 A google.com CNAME forcesafesearch.google.com A 216.239.38.120 SOA opnsense.localdomain"

The info field has CNAME record, which has the alias and the canonical name.

Other fields are the protocol field (DNS) and length field (189)

Another interesting thing I want to point out, is that a udp protocol is being used with a src port of 53 (the dns default port) as shown in the figure.

Part2:

In this part we will use cisco packet tracer to build a network that contains 4 routers, 2 switches, 5 PCs, a **webserver,** And **DNS server,** at least one subnet will use DHCP.
To connect the different networks/subnets together, OSPF routing algorithm will be used,
For my network the IP address will be 205.0.7.0000 0000/24

The following figure is a representation of the whole network and its components:

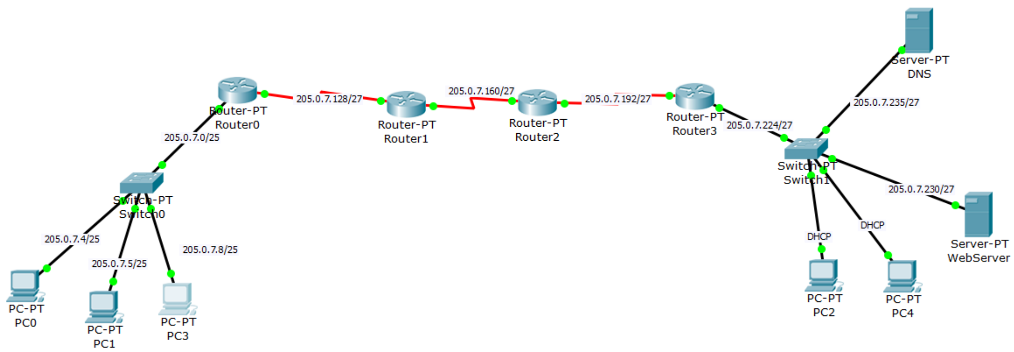

Figure 6: the network with its own components.

Figure 6, shows all the component with their IPs, so now I will show you how we get to there. First, we had a network IP of 205.0.7.0/24, our network has 5 subnets, and the closest number of subnet bits to represent is 3 (with 8 subnets) so I combined the first four and left the last four as they are, and got the following result:

- **First subnet:**
  Subnet: 205.0.7.0/25
  Subnet mask: 255.255.255.128

- **Second subnet**
  Subnet: 205.0.7.128/27
  Subnet mask: 255.255.255.224

- **Third subnet**
  Subnet: 205.0.7.160/27
  Subnet mask: 255.255.255.224

- **Fourth subnet**
  Subnet: 205.0.7.192/27
  Subnet mask: 255.255.255.224

- **Fifth subnet**
  Subnet: 205.0.7.224/27
  Subnet mask: 255.255.255.224

After calculating the 5 subnets, the following configuration we applied to each router:

Connected Element	Connection Type	IP/subnet mask
Router 0	Fast Ethernet 0/0	205.0.7.1/255.255.255.128
	Serial 2/0	205.0.7.129/255.255.255.244
Router 1	Serial 2/0	205.0.7.130/255.255.255.244
	Serial 3/0	205.0.7.161/255.255.255.224
Router 2	Serial 2/0	205.0.7.193/255.255.255.224
	Serial 3/0	205.0.7.162/255.255.255.224
Router 3	Fast Ethernet 0/0	205.0.7.225/255.255.255.224
	Serial 2/0	205.0.7.194/255.255.255.224

Table 1: routers configurations

Where Fast Ethernet 0/0 represent the ethernet connection to a device (used with LAN)
And Serial 2/0 represents the serial connection to a device (used with WAN)
And Serial 3/0 same as above, but with another device.

After these configurations were set, an IP of 205.0.7.4 was given to PC0, and using it we ping router 0:

```
PC>ping 205.0.7.1

Pinging 205.0.7.1 with 32 bytes of data:

Reply from 205.0.7.1: bytes=32 time=1ms TTL=255
Reply from 205.0.7.1: bytes=32 time=0ms TTL=255
Reply from 205.0.7.1: bytes=32 time=1ms TTL=255
Reply from 205.0.7.1: bytes=32 time=0ms TTL=255

Ping statistics for 205.0.7.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

Figure 7: PING with PC0 to router 0

Everything is working fine, now we will try to ping router 3:

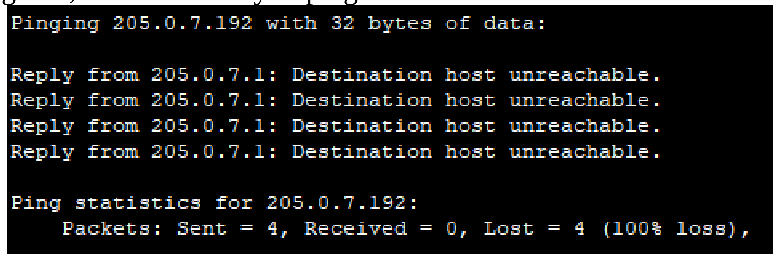

Figure 8: PING with PC0 to router 3 fail

As we can see, we can't reach it as we don't have a routing algorithm, so we will use OSPF (Open Shortest Path First) which is a link-state routing protocol used in computer networks. It determines the shortest path for data packets to travel through an IP network by exchanging routing information among routers and calculating the best routes based on metrics like bandwidth and delay.

To configure OSPF, these commands has to be put on the CLI of each Router,

Router # conf //to enter configure terminal
Router (config) # router ospf 1 #to start OSPF configuration
Router (config-router) # network ip wildcard (complement of mask) area 0
And so on

so, for example in Router 2:

Router # conf //to enter configure terminal
Router (config) # router ospf 1 #to start OSPF configuration
Router (config-router) # network 205.0.7.160 0.0.0.31 area 0
Router (config-router) # network 205.0.7.192 0.0.0.31 area 0

After OSPF was set, pinging with PC0 to route 3 was successful as shown:

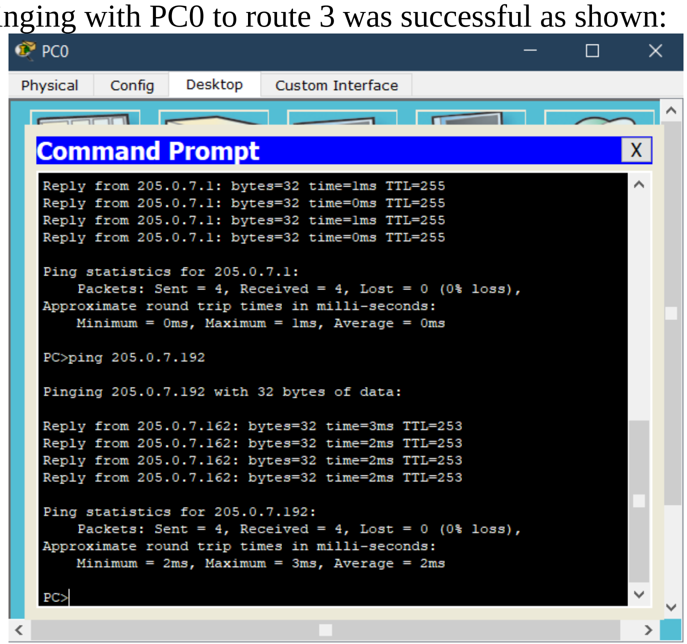

Figure 9: PING with PC0 to router 3 success.

After we added 2 extra PC to 205.0.7.0 connection.
Then add DNS server to 205.0.7.224 connection with IP 205.0.7.235
Then a web server was added to 205.0.7.224, with IP of 205.0.7.230 and it was registered to the DNS server as shown:

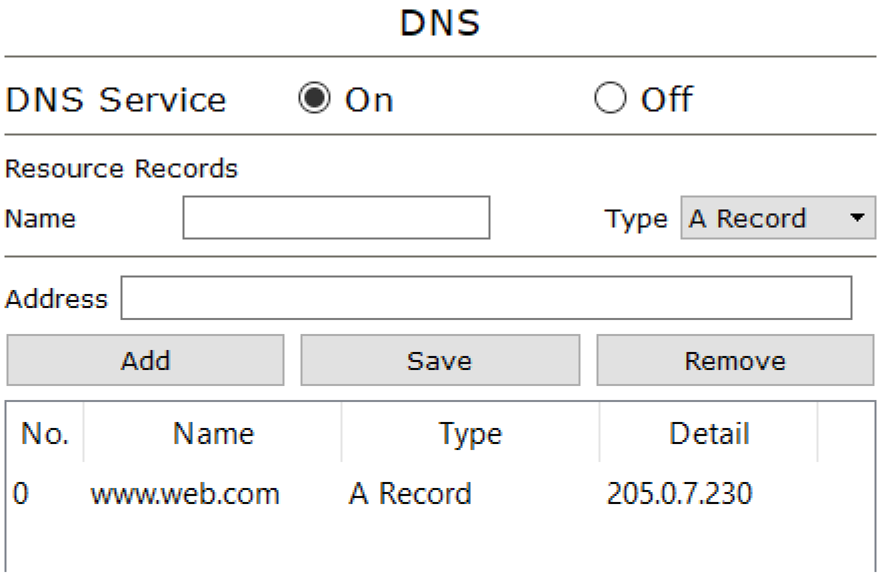

Figure 10: adding web server to DNS

Before adding the last two PC to 205.0.7.224, I want to enable DHCP, to do that we did the following:

Router(config)#
Router(config)#ip dhcp pool MY_LAN
Router(dhcp-config)#network 205.0.7.224 255.255.255.224
Router(dhcp-config)#default-router 205.0.7.225
Router(dhcp-config)#dns-server 205.0.7.235

After adding DHCP to the subnet, we requested and IP for the last two PC as shown:

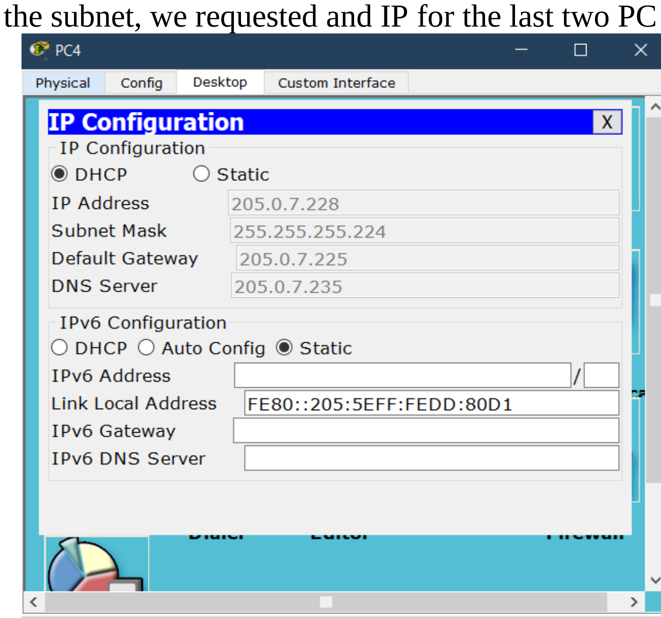

Figure 11: IP requesting for PC4

Now we will try to access the web server using PC4 (same subnet) and PC0 (another subnet):

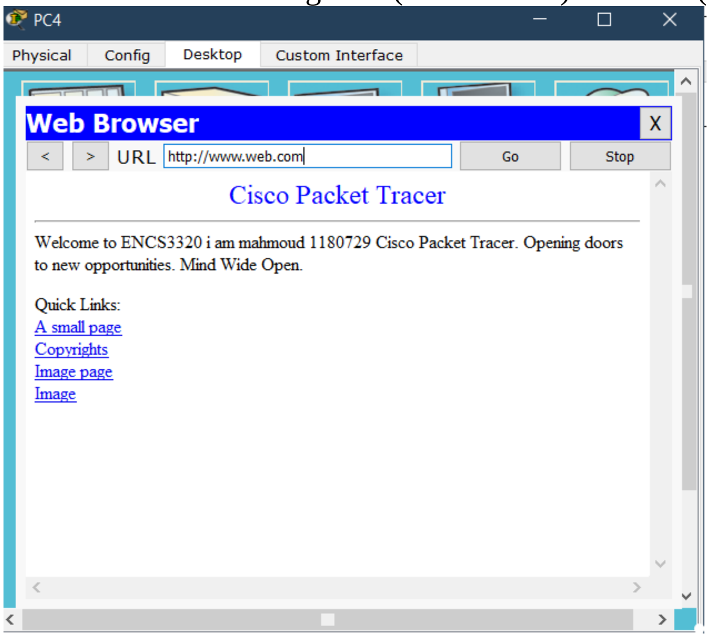

Figure 12: access web server from PC4

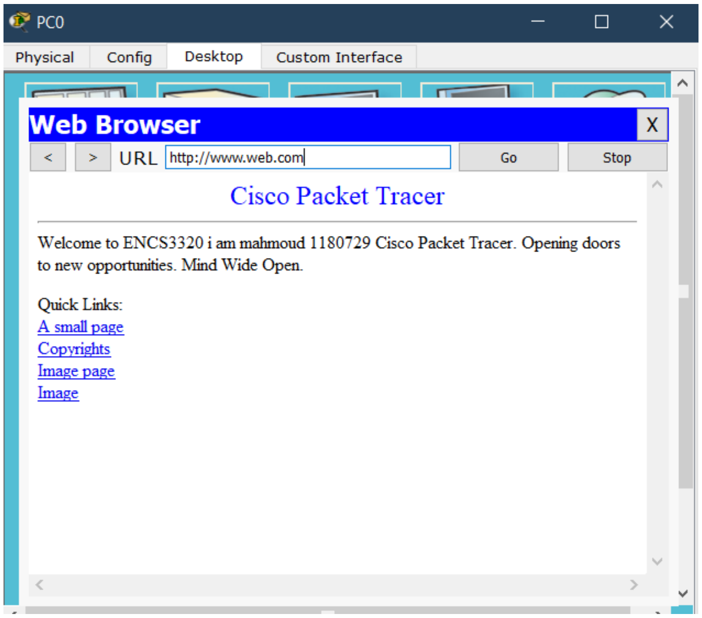

Figure 13: access web server PC0

We can see that it was successful for both, although PC0 took more time (expected)
NOTE: don't forget to add DNS to PC0.

Lastly let’s use tracert command to check the reachability from PC0 to PC4 and from PC0 to 205.0.7.162 (router 2 from right side subnet).

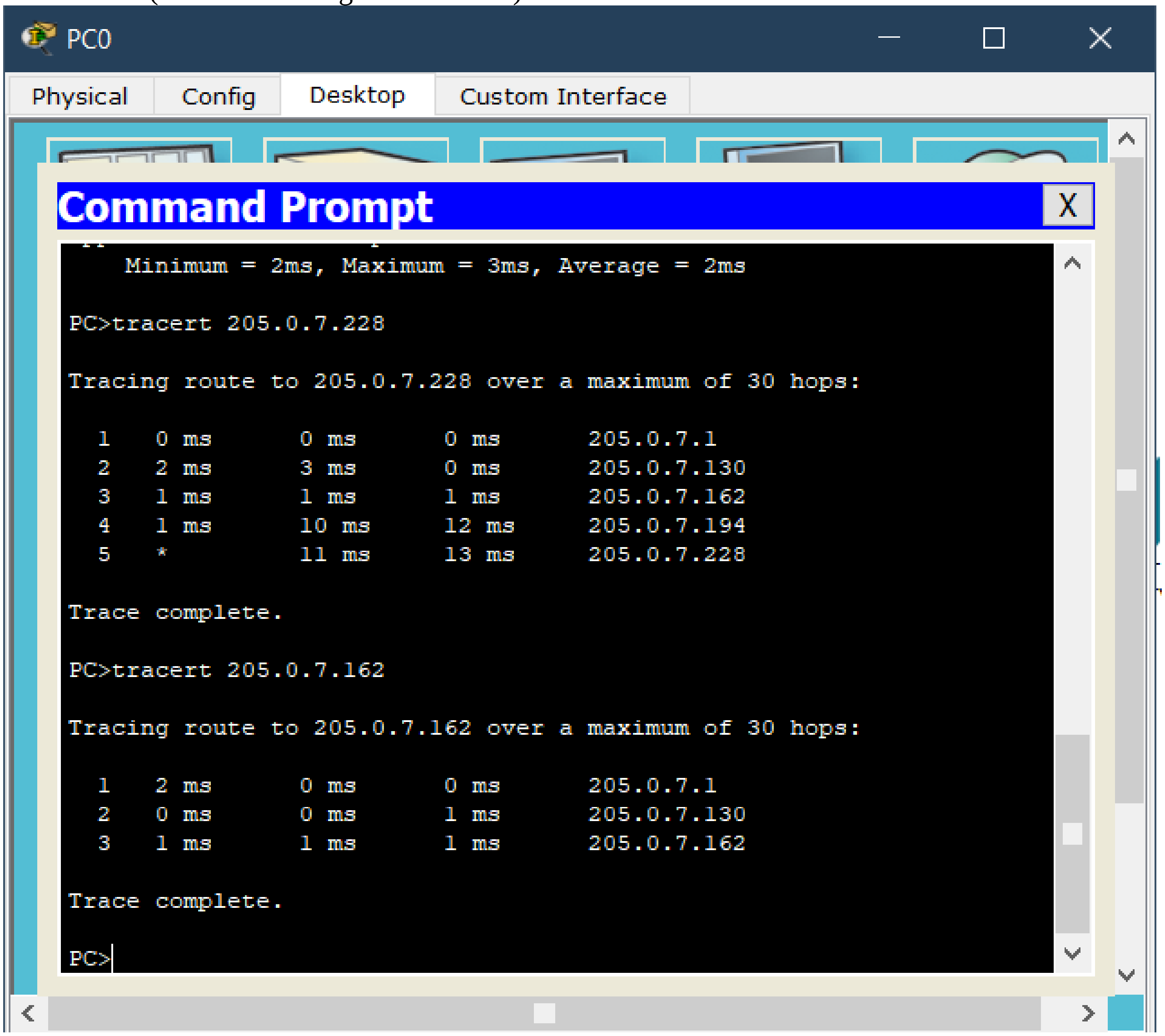

Figure 14: Tracert.

As we can see, packet needed to go through all subnets to reach PC4, but only 3 which is true to our design.