CSCI 5010 – Fundamentals of Data Communications

Lab 7

Applications:
DHCP and DNS

University of Colorado Boulder
Department of Computer Science
Network Engineering

Levi Perigo, Ph.D.

Objectives

- Learn DHCP configuration and concepts.
- Learn DNS basic configuration and concepts.

Summary:

As networks scale, we need applications to help manage our IP addresses and configuration of devices. Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS) are applications used for better network management. You will use GNS3 in this lab to implement these protocols. You will be examining the messages on Wireshark at the packet level to get a deeper understanding about the protocol mechanics. This lab covers many interview questions that you will be asked when you're applying for internships and jobs. The main goal of the lab is to help you gain protocol knowledge and basic implementation skills to be able to configure these services on networking devices.

Objective-1: Getting started with DHCP

1. Startup GNS3 and initialize the following topology:

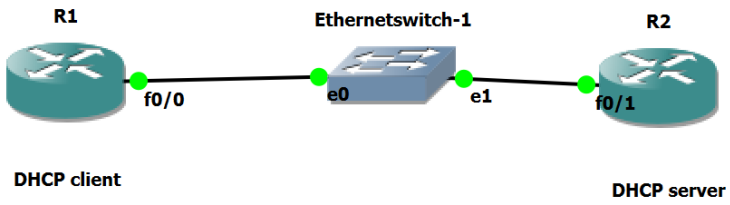
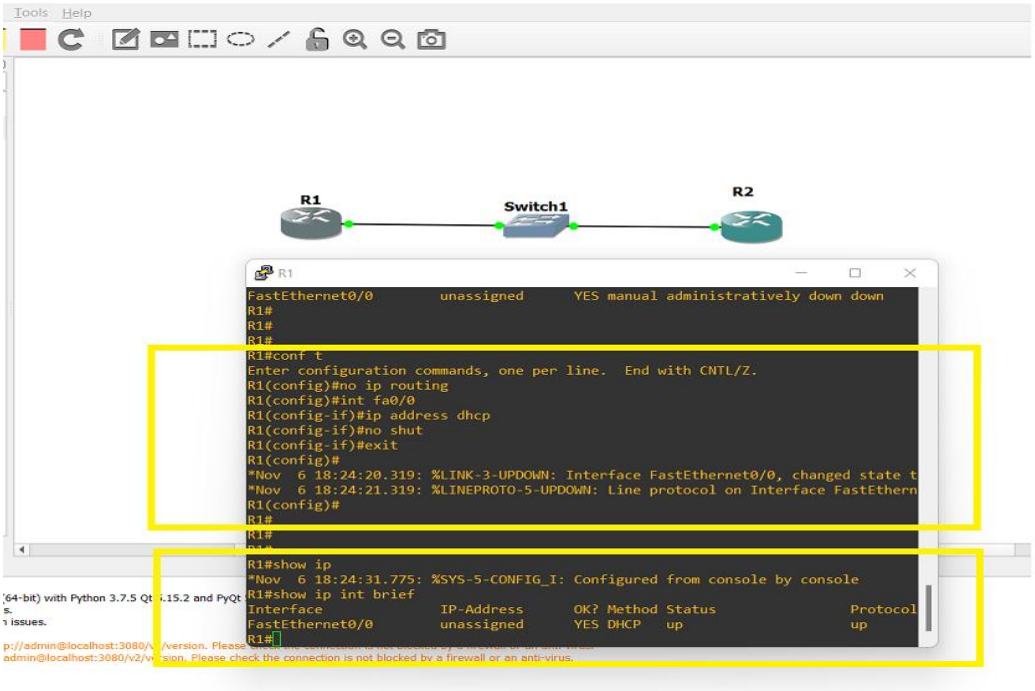


Fig.1

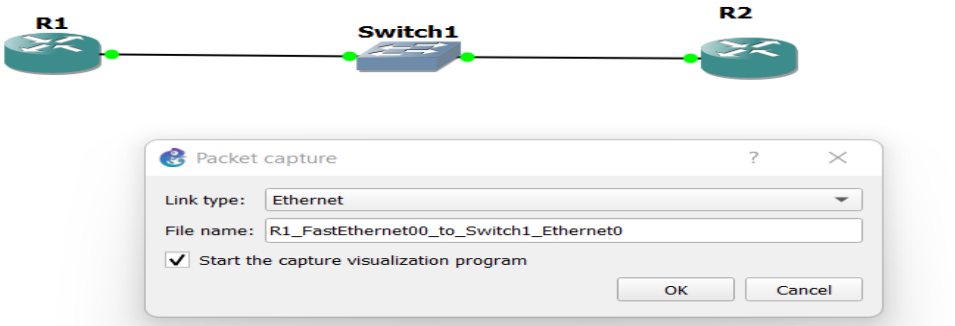
2. Configure R1's f0/0 interface to obtain its IP address from DHCP. Paste a screenshot of the interface configuration. **[3 points]**

Ans : IP ADDRESS DHCP

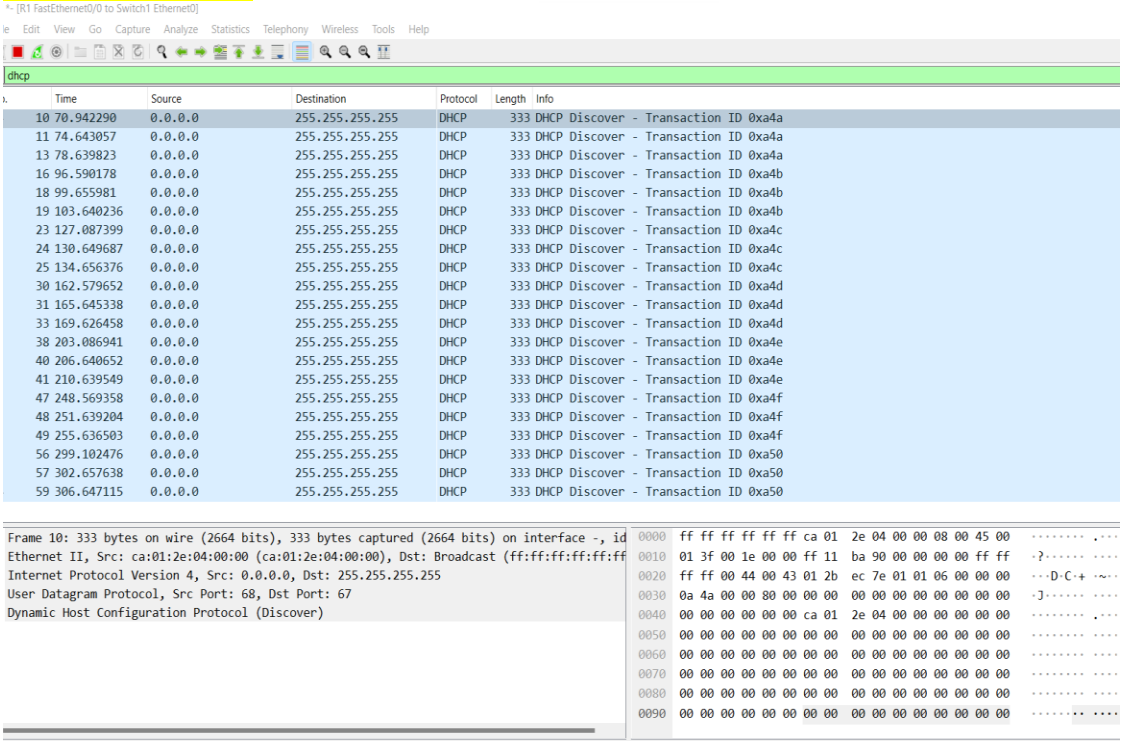


3. Start a Wireshark capture in this step to capture all DHCP messages that will be exchanged in the next step. In the above topology, where would you initiate a Wireshark capture? **[1 point]**
(Hint: To start a capture on Wireshark, right click on an interface and click start capture)

Ans :: By right clicking on Fa0/0 interface, it will open a Wireshark window where DHCP capture can be done. Pls find the screenshot below:



Wireshark Capture



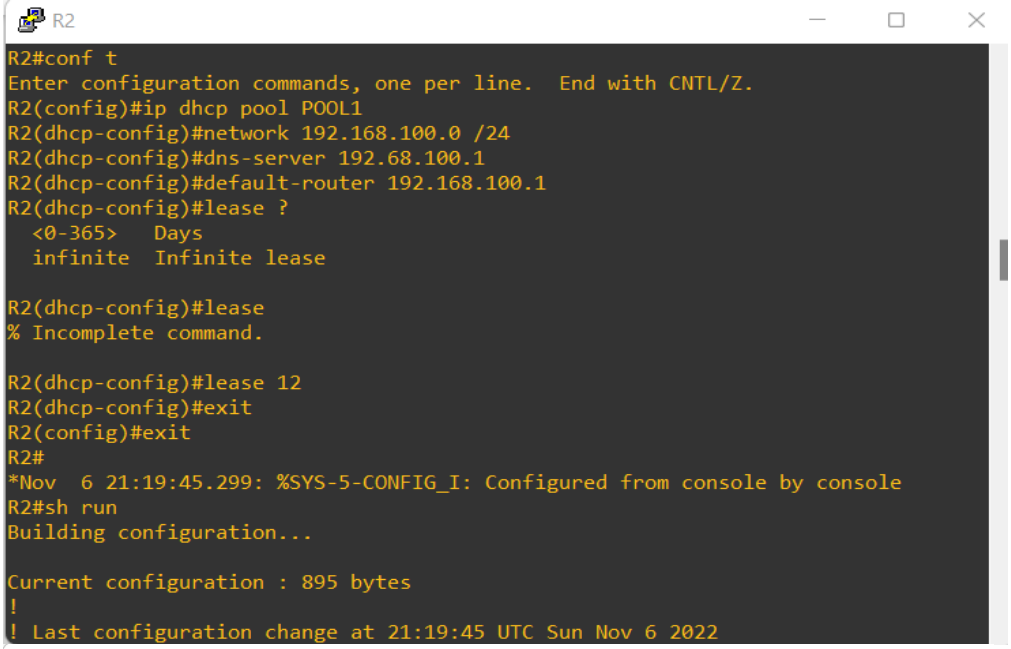
4. If R2 is only a DHCP server, do you need any other basic configuration on R2 besides the configuration of a DHCP pool? Explain if the f0/1 interface of R2 needs to have an IP address. Justify your answer. **[5 points]**

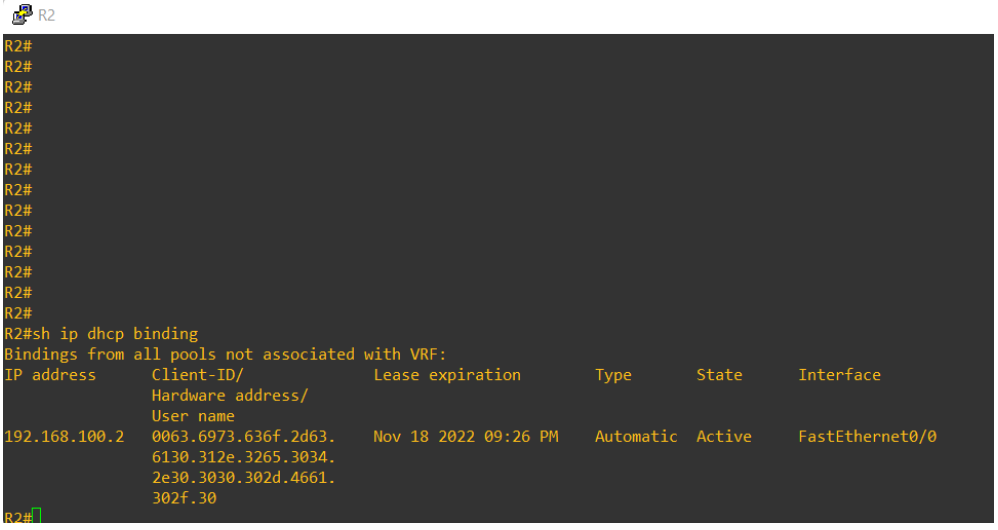
Yes. From what Levi taught in DHCP class, technically the DHCP Server must have a manually set static IP address for the server to communicates its IP address to the client during the initial discovery packet binding process. This address usually needs to be known when it starts up so that's pretty much static. For example: The server itself must have a fixed IP address usually in the subnet where it is going to distribute IP addresses. if the local subnet will be 192.168.10.0/24, usually (this is not a rule but a recommendation), the servers get the lower number IPs. In our case, 192.168.10.1/24 is usually the router/gateway. 192.168.10.2 could be the DNS server and 192.168.10.3 could be the DHCP server.Then we can use the remaining addresses, from 192.168.10.10 to let's say 192.168.10.250 as an address pool configured on the DHCP server. These addresses will be distributed to client devices if they request an IP.

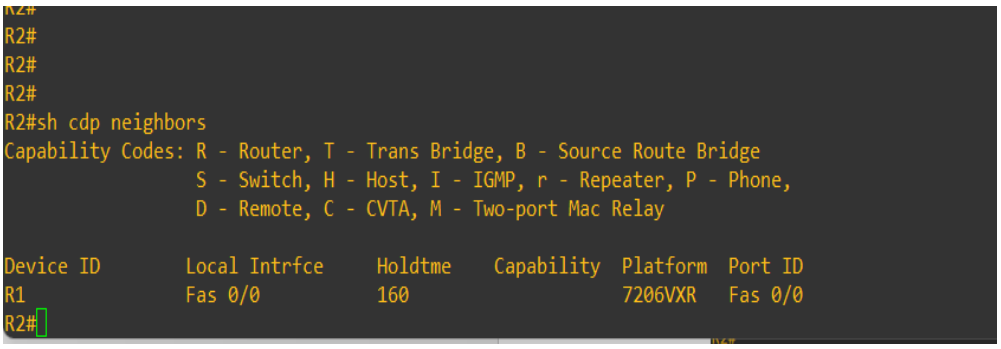
Also, A DHCP server must have a configured IP address so that it can know which scopes are locally attached to physical interfaces, and which Scopes can only be served via a DHCP relay

5. Having made sure you started Wireshark capture in Step 3, now configure R2 to be a DHCP server. Paste a screenshot of the configuration you made on R2. **[5 points]**

SHOW IP DHCP BINDING, SHOW CDP NEIGHBOURS







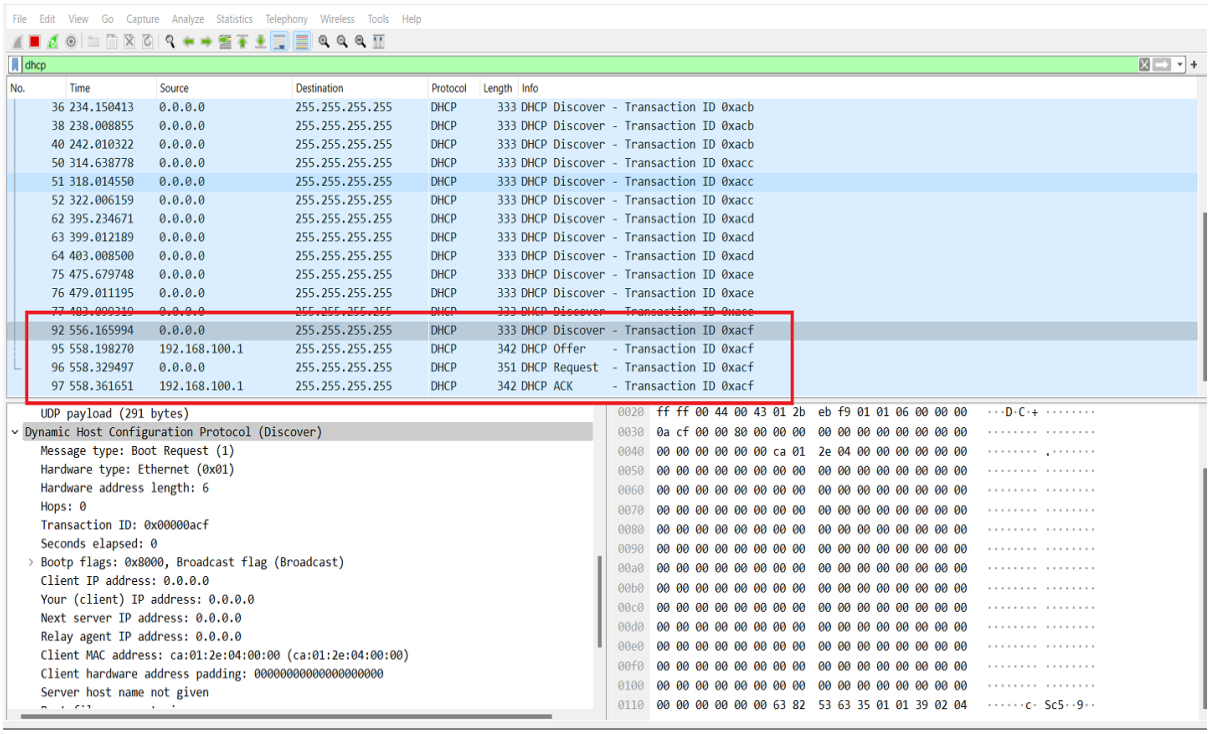
6. Did you get an IP address on R1? Indicate from its CLI that it got a DHCP address. How do you know this? **[2 points]**

Using the SHOW IP INT BRIEF command and also sh CDP Neighbors on R2

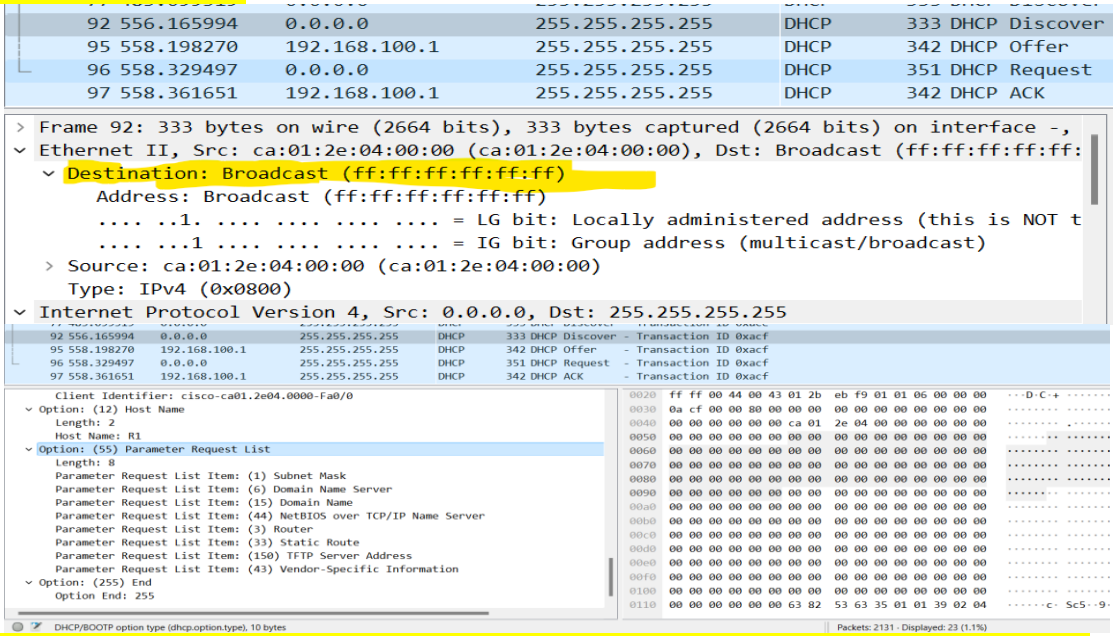
```
R1
*Nov  6 18:32:30.319: %SYS-5-CONFIG_I: Configured from console by console
R1#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        unassigned      YES manual up                    up
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int fa0/0
R1(config-if)#ip address dhcp
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#exit
R1#
*Nov  6 18:33:10.131: %SYS-5-CONFIG_I: Configured from console by console
R1#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        unassigned      YES DHCP   up                    up
R1#
*Nov  6 21:26:03.315: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned
 DHCP address 192.168.100.2, mask 255.255.255.0, hostname R1

R1#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.100.2   YES DHCP   up                    up
R1#
```

7. In the above step, capture the DHCP messages that were exchanged. Explain in detail the four messages. For each of these messages, mention the Source IP, Destination IP, Source MAC and Destination MAC that you see. **[10 points]**

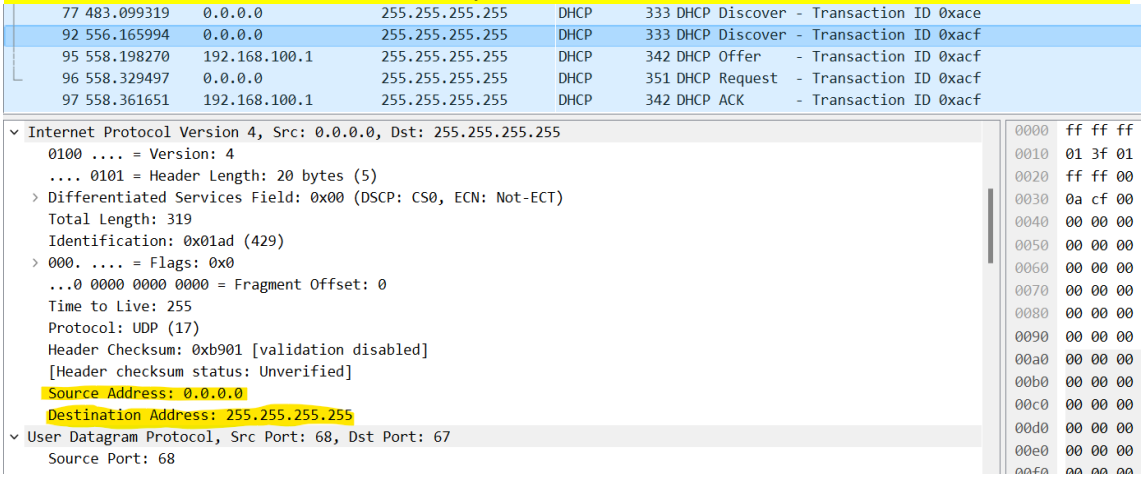
DHCP Discover :: In lamen terms, a DHCP client uses a technique called "broadcasting" using the destination limited broadcast address (255.255.255.255/ff:ff:ff:ff:ff:ff) which sends out a message that basically says "Can someone give me an IP address?" [This is called a DHCPDiscover].



The 291 byte DHCP Discover request constitutes of "Boot Request" message type, client IP address, Next server IP address, Relay agent IP Address, Client Mac Address, and a Parameter Request list having"Subnet Mask, DNS, Domain Name, Static Route"



DHCP Discover Source IP – 0:0:0:0, DHCP Discover Destination IP – 255.255.255.255

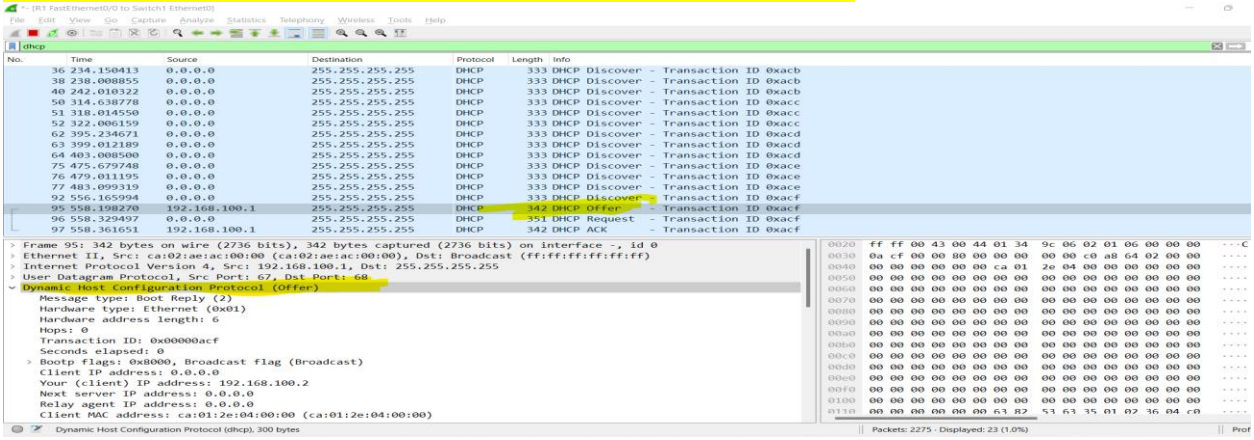


DHCP Discover Source MAC – ca: 01:2e: 04:00:00 DHCP Destination MAC – ff:ff:ff:ff:ff:ff

```
    77 483.099319    0.0.0.0          255.255.255.255    DHCP    333 DHCP Discover - Transaction ID 0xace
    92 556.165994    0.0.0.0          255.255.255.255    DHCP    333 DHCP Discover - Transaction ID 0xacf
    95 558.198270    192.168.100.1    255.255.255.255    DHCP    342 DHCP Offer    - Transaction ID 0xacf
    96 558.329497    0.0.0.0          255.255.255.255    DHCP    351 DHCP Request  - Transaction ID 0xacf
    97 558.361651    192.168.100.1    255.255.255.255    DHCP    342 DHCP ACK      - Transaction ID 0xacf
```

```
> Frame 92: 333 bytes on wire (2664 bits), 333 bytes captured (2664 bits) on interface -, id 0
v Ethernet II, Src: ca:01:2e:04:00:00 (ca:01:2e:04:00:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  v Destination: Broadcast (ff:ff:ff:ff:ff:ff)
      Address: Broadcast (ff:ff:ff:ff:ff:ff)
      .... ..1. .... .... .... .... = LG bit: Locally administered address (this is NOT the factory default)
      .... ...1 .... .... .... .... = IG bit: Group address (multicast/broadcast)
  v Source: ca:01:2e:04:00:00 (ca:01:2e:04:00:00)
      Address: ca:01:2e:04:00:00 (ca:01:2e:04:00:00)
      .... ..1. .... .... .... .... = LG bit: Locally administered address (this is NOT the factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
    Type: IPv4 (0x0800)
```

b) **DHCP OFFER** : **On an IP network, the job of the DHCP server is to notice the MAC (Media Access Control) hardware address that has sent that request, work out what IP address to dish out, and send a packet back to that physical address with the offer of an IP address. [This is called a DHCPOffer]**



The 300-byte DHCP Offer request constitutes of "Boot Reply" message type, client IP address, and a Parameter Request list having "DHCP Server Identifier, Lease Time, Renewal time, Binding time, Subnet Mask, Domain Name Server and Router"DHCP Offer Source IP – 192.168.100.1, DHCP offer Destination IP – 255.255.255.255

```
    75 475.679748    0.0.0.0          255.255.255.255    DHCP    333 DHCP Discover - Transaction ID 0xace
    76 479.011195    0.0.0.0          255.255.255.255    DHCP    333 DHCP Discover - Transaction ID 0xace
    77 483.099319    0.0.0.0          255.255.255.255    DHCP    333 DHCP Discover - Transaction ID 0xace
    92 556.165994    0.0.0.0          255.255.255.255    DHCP    333 DHCP Discover - Transaction ID 0xacf
    95 558.198270    192.168.100.1    255.255.255.255    DHCP    342 DHCP Offer    - Transaction ID 0xacf
    96 558.329497    0.0.0.0          255.255.255.255    DHCP    351 DHCP Request  - Transaction ID 0xacf
    97 558.361651    192.168.100.1    255.255.255.255    DHCP    342 DHCP ACK      - Transaction ID 0xacf
```

```
v Internet Protocol Version 4, Src: 192.168.100.1, Dst: 255.255.255.255
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 328
    Identification: 0x0000 (0)
  > 000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 255
    Protocol: UDP (17)
    Header Checksum: 0x95fb [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.100.1
    Destination Address: 255.255.255.255
v User Datagram Protocol, Src Port: 67, Dst Port: 68
    Source Port: 67
    Destination Port: 68
```

```
0000  ff ff ff ff ff ff ca 02
0010  01 48 00 00 00 00 ff 11
0020  ff ff 00 43 00 44 01 34
0030  0a cf 00 00 80 00 00 00
0040  00 00 00 00 00 00 ca 01
0050  00 00 00 00 00 00 00 00
0060  00 00 00 00 00 00 00 00
0070  00 00 00 00 00 00 00 00
0080  00 00 00 00 00 00 00 00
0090  00 00 00 00 00 00 00 00
00a0  00 00 00 00 00 00 00 00
00b0  00 00 00 00 00 00 00 00
00c0  00 00 00 00 00 00 00 00
00d0  00 00 00 00 00 00 00 00
00e0  00 00 00 00 00 00 00 00
00f0  00 00 00 00 00 00 00 00
```

Ethernet (eth), 14 bytes

Packets: 2377 · Displayed: 23 (1.0%)

DHCP Offer Source MAC – ca: 01:2e: 04:00:00, DHCP Offer Dest MAC – ff:ff:ff:ff:ff:ff

```
    92 556.165994    0.0.0.0          255.255.255.255    DHCP    333 DHCP Discover - Transaction ID 0xacf
    95 558.198270    192.168.100.1    255.255.255.255    DHCP    342 DHCP Offer    - Transaction ID 0xacf
    96 558.329497    0.0.0.0          255.255.255.255    DHCP    351 DHCP Request  - Transaction ID 0xacf
    97 558.361651    192.168.100.1    255.255.255.255    DHCP    342 DHCP ACK      - Transaction ID 0xacf

> Frame 95: 342 bytes on wire (2736 bits), 342 bytes captured (2736 bits) on interface -, id 0
v Ethernet II, Src: ca:02:ae:ac:00:00 (ca:02:ae:ac:00:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  v Destination: Broadcast (ff:ff:ff:ff:ff:ff)
      Address: Broadcast (ff:ff:ff:ff:ff:ff)
      .... ..1. .... .... .... .... = LG bit: Locally administered address (this is NOT the factory default)
      .... ...1 .... .... .... .... = IG bit: Group address (multicast/broadcast)
  v Source: ca:02:ae:ac:00:00 (ca:02:ae:ac:00:00)
      Address: ca:02:ae:ac:00:00 (ca:02:ae:ac:00:00)
      .... ..1. .... .... .... .... = LG bit: Locally administered address (this is NOT the factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
    Type: IPv4 (0x0800)
> Internet Protocol Version 4, Src: 192.168.100.1, Dst: 255.255.255.255
v User Datagram Protocol, Src Port: 67, Dst Port: 68
    Source Port: 67
    Destination Port: 68
    Length: 308
    Checksum: 0x9c06 [unverified]

0000  ff ff ff ff ff
0010  01 48 00 00 00
0020  ff ff 00 43 00
0030  0a cf 00 00 80
0040  00 00 00 00 00
0050  00 00 00 00 00
0060  00 00 00 00 00
0070  00 00 00 00 00
0080  00 00 00 00 00
0090  00 00 00 00 00
00a0  00 00 00 00 00
00b0  00 00 00 00 00
00c0  00 00 00 00 00
00d0  00 00 00 00 00
00e0  00 00 00 00 00
00f0  00 00 00 00 00

Ethernet (eth), 14 bytes
Packets: 2388 · Displ
```

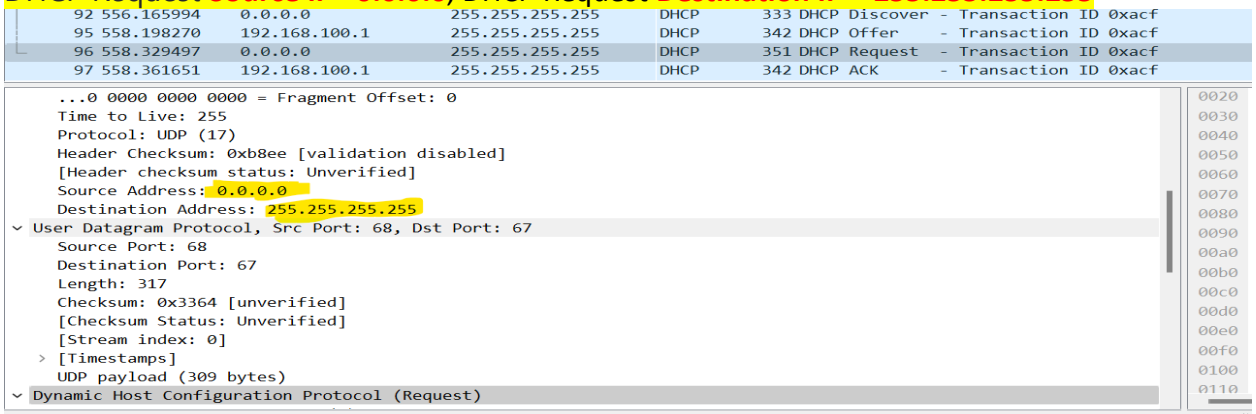
c) **DHCP Request** :: **The DHCP Client then will accept the IP appress offered by the DHCP Server DHCP pool "Yes, OK, I'll take that one please." [This is called a DHCPRequest]**

```
File  Edit  View  Go  Capture  Analyze  Statistics  Telephony  Wireless  Tools  Help
dhcp

No.   Time          Source          Destination          Protocol  Length  Info
   36 234.150413    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xacb
   38 238.008855    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xacb
   40 242.010322    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xacb
   50 314.638778    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xacc
   51 318.014550    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xacc
   52 322.006159    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xacc
   62 395.234671    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xacd
   63 399.012189    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xacd
   64 403.008500    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xacd
   75 475.679748    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xace
   76 479.011195    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xace
   77 483.099319    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xace
   92 556.165994    0.0.0.0         255.255.255.255      DHCP       333 DHCP Discover - Transaction ID 0xacf
   95 558.198270    192.168.100.1   255.255.255.255      DHCP       342 DHCP Offer    - Transaction ID 0xacf
   96 558.329497    0.0.0.0         255.255.255.255      DHCP       351 DHCP Request  - Transaction ID 0xacf
   97 558.361651    192.168.100.1   255.255.255.255      DHCP       342 DHCP ACK      - Transaction ID 0xacf

    [Checksum Status: Unverified]
    [Stream index: 0]
  > [Timestamps]
    UDP payload (309 bytes)
v Dynamic Host Configuration Protocol (Request)
    Message type: Boot Request (1)
    Hardware type: Ethernet (0x01)
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x00000acf
    Seconds elapsed: 0
  > Bootp flags: 0x8000, Broadcast flag (Broadcast)
    Client IP address: 0.0.0.0
    Your (client) IP address: 0.0.0.0
    Next server IP address: 0.0.0.0
    Relay agent IP address: 0.0.0.0
    Client MAC address: ca:01:2e:04:00:00 (ca:01:2e:04:00:00)

0000  ff ff ff ff ff ff ca 01
0010  01 51 01 ae 00 00 ff 11
0020  ff ff 00 44 00 43 01 3d
0030  0a cf 00 00 80 00 00 00
0040  00 00 00 00 00 00 ca 01
0050  00 00 00 00 00 00 00 00
0060  00 00 00 00 00 00 00 00
0070  00 00 00 00 00 00 00 00
0080  00 00 00 00 00 00 00 00
0090  00 00 00 00 00 00 00 00
00a0  00 00 00 00 00 00 00 00
00b0  00 00 00 00 00 00 00 00
00c0  00 00 00 00 00 00 00 00
00d0  00 00 00 00 00 00 00 00
00e0  00 00 00 00 00 00 00 00
00f0  00 00 00 00 00 00 00 00
```

The 309-byte DHCP request constitutes of "Boot Request" message type, client IP address, and "DHCP Server Identifier, Requested IP Address, IP Address Lease

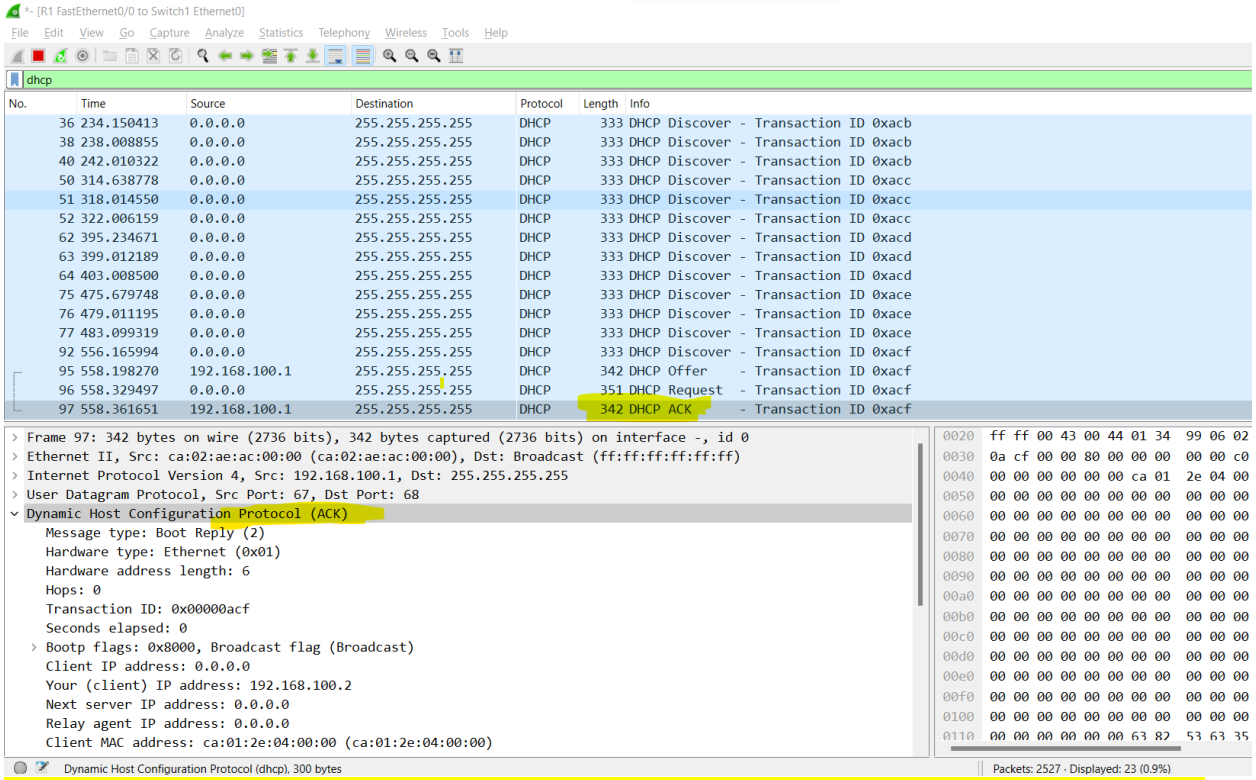
```
    92 556.165994    0.0.0.0          255.255.255.255    DHCP    333 DHCP Discover - Transaction ID 0xacf
    95 558.198270    192.168.100.1    255.255.255.255    DHCP    342 DHCP Offer    - Transaction ID 0xacf
    96 558.329497    0.0.0.0          255.255.255.255    DHCP    351 DHCP Request  - Transaction ID 0xacf
    97 558.361651    192.168.100.1    255.255.255.255    DHCP    342 DHCP ACK      - Transaction ID 0xacf

    Your (client) IP address: 0.0.0.0
    Next server IP address: 0.0.0.0
    Relay agent IP address: 0.0.0.0
    Client MAC address: ca:01:2e:04:00:00 (ca:01:2e:04:00:00)
    Client hardware address padding: 00000000000000000000
    Server host name not given
    Boot file name not given
    Magic cookie: DHCP
  > Option: (53) DHCP Message Type (Request)
  > Option: (57) Maximum DHCP Message Size
  > Option: (61) Client identifier
  > Option: (54) DHCP Server Identifier (192.168.100.1)
  > Option: (50) Requested IP Address (192.168.100.2)
  > Option: (51) IP Address Lease Time
  > Option: (12) Host Name
  > Option: (55) Parameter Request List
  > Option: (255) End

0020  ff ff 00 44 00 43 01 3d  33 64 01 01
0030  0a cf 00 00 80 00 00 00  00 00 00 00
0040  00 00 00 00 00 00 ca 01  2e 04 00 00
0050  00 00 00 00 00 00 00 00  00 00 00 00
0060  00 00 00 00 00 00 00 00  00 00 00 00
0070  00 00 00 00 00 00 00 00  00 00 00 00
0080  00 00 00 00 00 00 00 00  00 00 00 00
0090  00 00 00 00 00 00 00 00  00 00 00 00
00a0  00 00 00 00 00 00 00 00  00 00 00 00
00b0  00 00 00 00 00 00 00 00  00 00 00 00
00c0  00 00 00 00 00 00 00 00  00 00 00 00
00d0  00 00 00 00 00 00 00 00  00 00 00 00
00e0  00 00 00 00 00 00 00 00  00 00 00 00
00f0  00 00 00 00 00 00 00 00  00 00 00 00
0100  00 00 00 00 00 00 00 00  00 00 00 00
0110  00 00 00 00 00 00 63 82  53 63 35 01

Dynamic Host Configuration Protocol (dhcp), 309 bytes
Packets: 2456 · Displayed: 23 (0.9%)
```

DHCP Request **Source IP –0.0.0.0**, DHCP Request **Destination IP – 255.255.255.255**



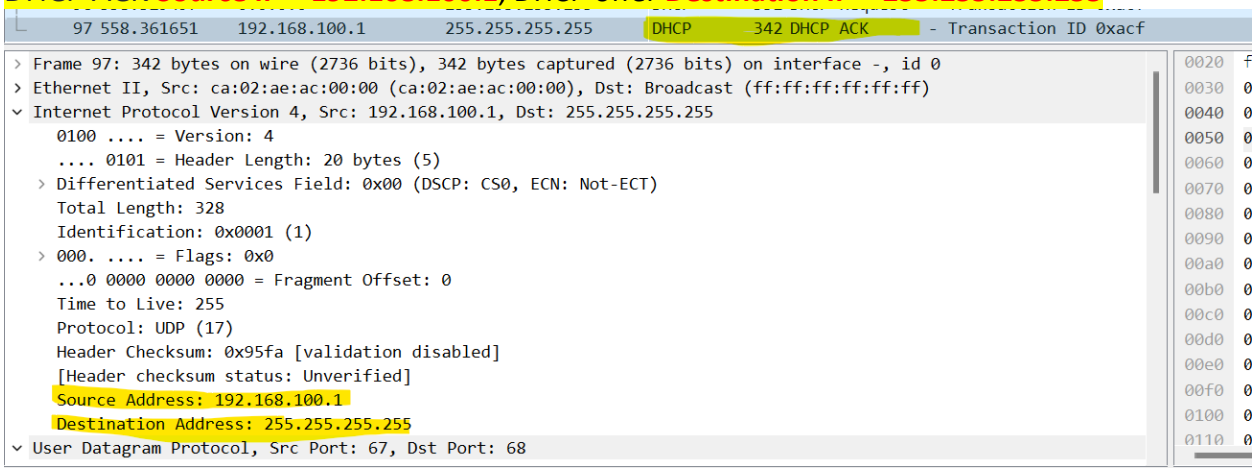
DHCP Request Source MAC – ca: 01:2e: 04:00:00, DHCP Request Dest MAC – ff:ff:ff:ff:ff:ff

DHCP Acknowledgment: the DHCP server says "Agreed" and allocated the requested IP Address to the DHCP client by sending ACK [This is called a DHCPAcknowledgement.]



The 309-byte DHCP Ack constitutes of "Boot Reply" message type, client IP address, and "DHCP Server Identifier, Requested IP Address, IP Address Lease Time, Renewal Time

DHCP ACK **Source IP – 192.168.100.1**, DHCP offer **Destination IP – 255.255.255.255**



DHCP ACK Source MAC – ca: 01:2e: 04:00:00, DHCP ACK Dest MAC – ff:ff:ff:ff:ff:ff

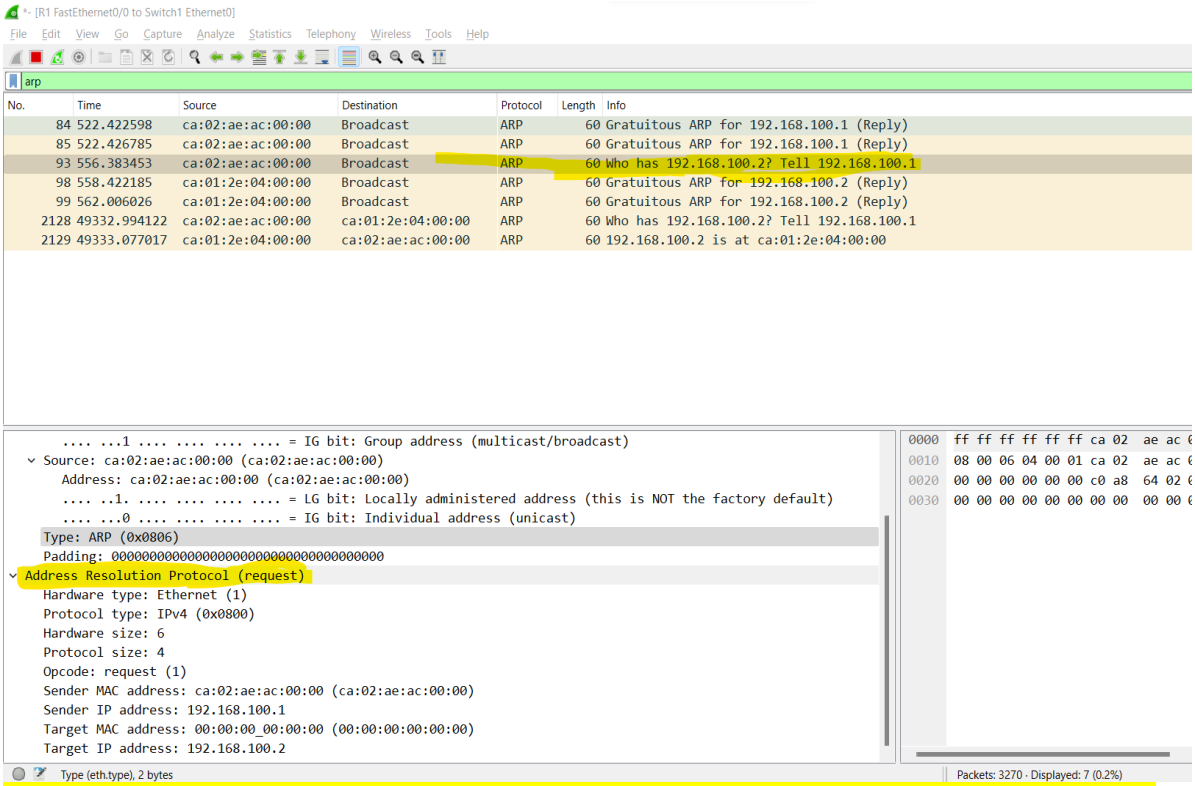
8. Which of the DHCP messages are broadcast at Layer 3? Which of the DHCP messages are broadcast at Layer 2? **[2 points]**

   DHCP Discover is a broadcast at Layer2, Layer3
   DHCP Offer is a broadcast at layer 3 as the server doesn't know client's IP Address
   DHCP Request is a broadcast at layer3
   DHCP Acknowledgment is a broadcast at Layer3.

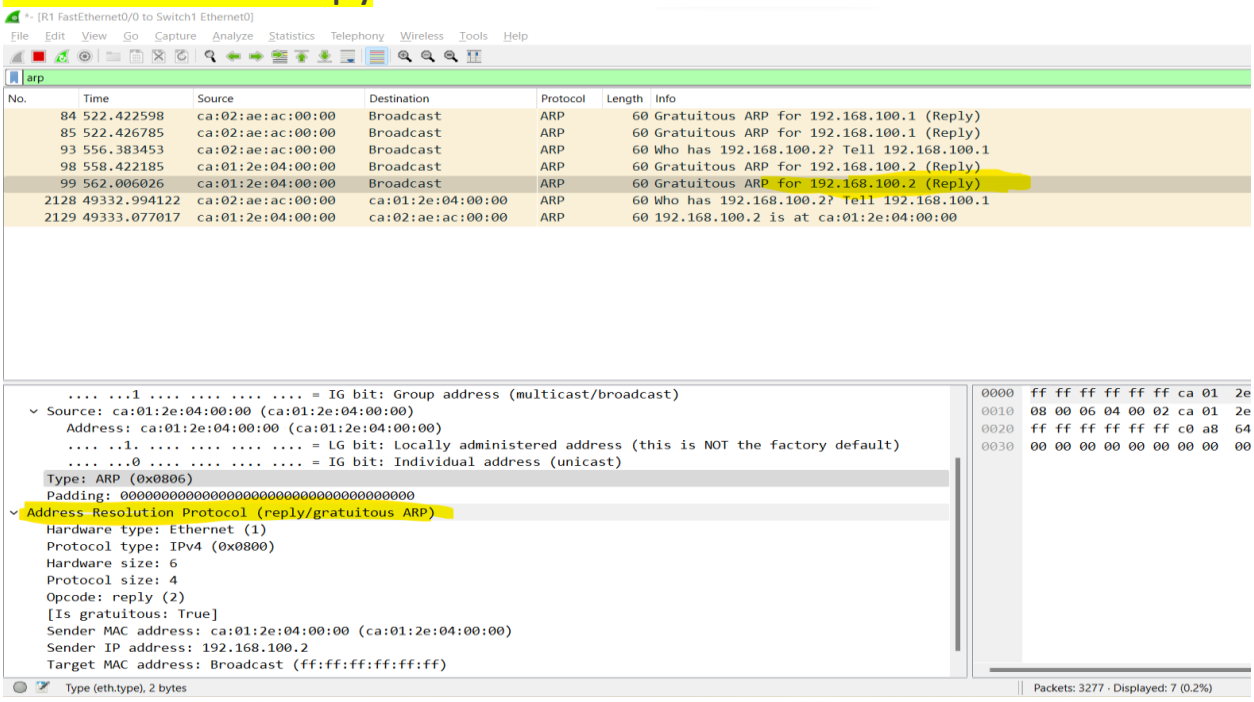
9. Are there any other messages you expect to see during the above process except DHCP messages? (Eg: From your theoretical knowledge of DHCP, postulate if you would see any ARP, ICMP or any other messages. Now verify the same on Wireshark)

   Explain if you see any of these messages. Why or why not? **[5 points]**

   Yes. I see ARP messages populated during the DHCP DORA (Discover, Offer, Request, Ack) process.
   Generally, ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. Here DHCP Client (R1) wants to receive DHCP IP from Server, so it first sends out ARP request by broadcasting at Layer 2 using (ff:ff:ff:ff:ff:ff) to map MAC address in its ARP cache. Server then shoots a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the R1.



Ideally a gratuitous ARP request is an ARP request packet where the source and destination IP are both set to the IP of the machine issuing the ARP packet and the destination MAC is set to the broadcast address ff:ff:ff:ff:ff:ff. Here in the request the server is broadcasting for "192.168.100.2" to hosts asking who has 192.168.100.2 and receives a reply from the respective host that has allocated with the 100.2/24 IP address as Gratuitous Reply.



Objective-2: DHCP server with multiple clients

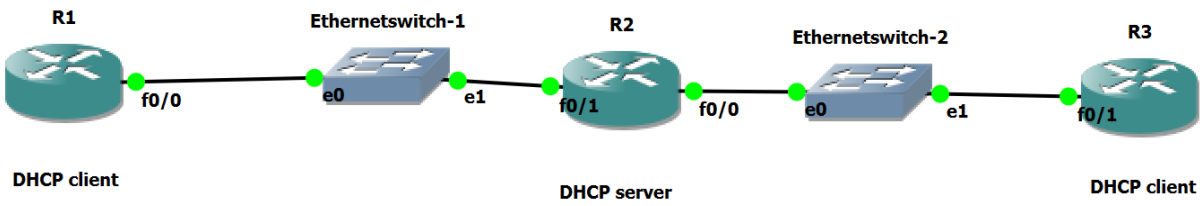


Fig.2

1. Refer to figure 2. Could this network design work? Can a single DHCP server serve two different DHCP clients as shown in the figure? If yes, explain what configuration changes you will need to do on R2 to make this work, and why you would have to make these modifications. Paste the configuration change you made on R2 to make it work.
   [10 points]

Yes. A single DHCP server can multiple clients, and an DHCP server can support more clients in an environment with longer leases, or an environment that consists of constantly connected devices such as cable modems.

Basically, I have configured R2 setup in a way that DHCP will assign ip addresses from different pools depending on which port/interface it is connected to. So DHCP Client (R1) is configured to get IP Addresses from DHCP Pool "PHOOL_1" and the DHCP client (R3) is configured to get IP Address from DHCP Phool "PHOOL_2"

Configuration changes made on R2 for the above setup to work:

Step1: Manually configure a static IP Address for the int fa0/1 on R2 to transmit/receive or to route packets between DHCP Server (R2) and DHCP Client (R3)

R2

```
R2(config)#int fa1/0
R2(config-if)#ip address 10.0.0.1 /24
                           ^
% Invalid input detected at '^' marker.

R2(config-if)#ip address 10.0.0.1 255.255.255.0
R2(config-if)#no shut
R2(config-if)#exit
R2(config)#
*Nov  7 04:26:33.185: %LINK-3-UPDOWN: Interface FastEthernet1/0, changed state to up
*Nov  7 04:26:34.185: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/0, changed state to up
R2(config)#exit
R2#show
*Nov  7 04:26:38.769: %SYS-5-CONFIG_I: Configured from console by console
R2#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.100.1   YES manual up                    up
FastEthernet1/0        10.0.0.1        YES manual up                    up
```

Step 2:: Configure the DHCP Pool on R2 ::

```
                302f.30
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ip dhcp pool POOL2
R2(dhcp-config)#network 10.0.0.0 255.255.255.0
R2(dhcp-config)#default-router 10.0.0.1
R2(dhcp-config)#dns-server 10.0.0.1
R2(dhcp-config)#exit
R2(config)#exit
R2#
*Nov  7 04:28:10.009: %SYS-5-CONFIG_I: Configured from console by console
R2#sh ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.100.1   YES manual up                    up
FastEthernet1/0        10.0.0.1        YES manual up                    up
```

Step 3 : Check the DHCP binding on R2 using “ip dhcp binding”

R2

```
R2#
R2#
R2#sh ip dhcp binding
Bindings from all pools not associated with VRF:
IP address      Client-ID/              Lease expiration        Type       State      Interface
                Hardware address/
                User name
10.0.0.2        0063.6973.636f.2d63.    Nov 08 2022 04:28 AM    Automatic  Active     FastEthernet1/0
                6130.332e.3065.3634.
                2e30.3030.302d.4661.
                302f.30
192.168.100.2   0063.6973.636f.2d63.    Nov 18 2022 09:26 PM    Automatic  Active     FastEthernet0/0
                6130.312e.3265.3034.
                2e30.3030.302d.4661.
                302f.30
R2#sh cdp neighbours
                ^
% Invalid input detected at '^' marker.

R2#sh cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay
```

Step 4 :: Check the client mapping using "cdp neighbors" command . Here you could see R3, R1 dhcp binding to R2.

```
R2#sh cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
R3               Fas 1/0           131                    7206VXR   Fas 0/0
R1               Fas 0/0           127                    7206VXR   Fas 0/0
R2#
```

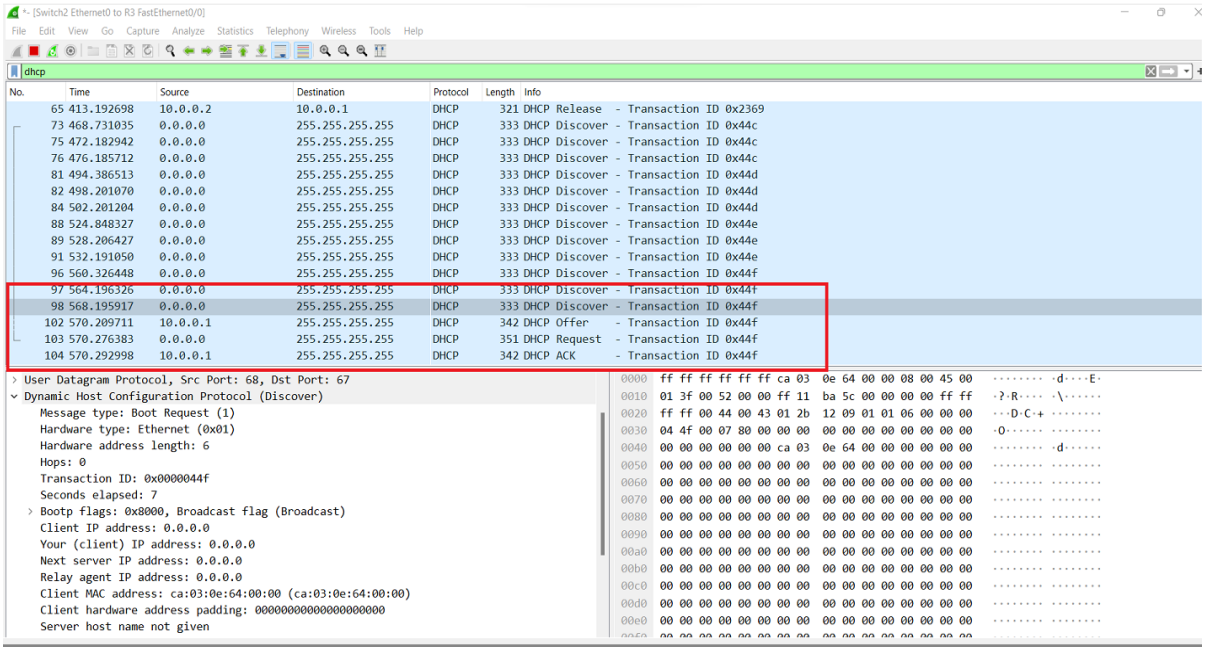
Step 5:: Configure the "ip address dhcp" on DHCP Client (R3) . You could find R3 (Client) is configured with dynamic IP Address from DHCP Server Pool_2.

```
R3
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#no ip routing
R3(config)#int fa0/0
R3(config-if)#ip address dhcp
R3(config-if)#no shut
R3(config-if)#exit
R3(config)#exit
*Nov  7 14:07:14.007: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state t
o up
*Nov  7 14:07:15.007: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/0, changed state to up
R3(config)#exit
R3#sh
*Nov  7 14:07:15.467: %SYS-5-CONFIG_I: Configured from console by console
R3#sh ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        unassigned      YES DHCP   up                    up
R3#
*Nov  7 14:31:29.131: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned
 DHCP address 10.0.0.2, mask 255.255.255.0, hostname R3

R3#sh ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        10.0.0.2        YES DHCP   up                    up
```

2. Now configure R3 as a DHCP client and R2 configured to also be the DHCP server for R3. Paste screenshots of DHCP messages exchanged and R3 getting the IP via DHCP.
[10 points]

Configuring R2 as Server for R3	Configuring R3 as Client
R2(config)#ip dhcp pool POOL2	R3(config)#no ip routing
R2(dhcp-config)#network 10.0.0.0 255.255.255.0	R3(config)#int fa0/0
R2(dhcp-config)#default-router 10.0.0.1	R3(config-if)#ip address dhcp
R2(dhcp-config)#dns-server 10.0.0.1	R3(config-if)#no shut
R2(dhcp-config)#exit	R3(config-if)#exit

```
                302f.30
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ip dhcp pool POOL2
R2(dhcp-config)#network 10.0.0.0 255.255.255.0
R2(dhcp-config)#default-router 10.0.0.1
R2(dhcp-config)#dns-server 10.0.0.1
R2(dhcp-config)#exit
R2(config)#exit
R2#
*Nov  7 04:28:10.009: %SYS-5-CONFIG_I: Configured from console by console
R2#sh ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.100.1   YES manual up                    up
FastEthernet1/0        10.0.0.1        YES manual up                    up
R3#
*Nov  7 15:14:20.323: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned
 DHCP address 10.0.0.2, mask 255.255.255.0, hostname R3

R3#sh ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        10.0.0.2        YES DHCP   up                    up
R3#
```



3. When R1 and R3 sent DHCP DISCOVER packets, how did R2 choose which IP to assign? How does R2 know which DHCP pool to use to loan IPs, if there are multiple pools configured on R2. **[10 points]**

Based on the port/Interface configured with the same network subnet.

Upon receiving a DHCP Discover, the DHCP server finds a user class matching the client and selects an IP address in the address range of the user class for the client. A user class can include multiple matching rules, and a client matches the user class as long as it matches any of the rules. In address pool view, you can specify different address ranges for different user classes.

The DHCP server selects an IP address for a client by performing the following steps:

1. **DHCP server compares the client against DHCP user classes in the order they are configured.**
2. **If the client matches a user class, the DHCP server selects an IP address from the address range of the user class.**

For example, two address pools 10.10.1.0/24 and 10.1.1.0/25 are configured but not applied to any DHCP server's.

- **If the IP address of the receiving interface is 10.10.1.0/25, the DHCP server selects the address pool 10.1.1.0/25. If the address pool has no available IP addresses, the DHCP server will not select the other pool and the address allocation will fail.**
- **If the IP address of the receiving interface is 10.10.1.130/25, the DHCP server selects IP addresses for clients from the address pool 10.1.1.0/24.**

If only a one seconday subnet is matched, the DHCP server does not select any IP address from other secondary subnets when the matching subnet has no assignable addresses.

4. Explain excluded DHCP addresses are and why you would use them. Did you configure this on R2? If so, what are some of the DHCP excluded addresses on R2 in your topology? **[3 points]**

Excluded DHCP addresses are nothing but "Reserved IP Addresses". Some network devices need to use statically assigned IP Addresses rather than dynamically assigned IP Addresses some network devices may/may not support DHCP, such as printers etc
For the devices that need static IP assignments, we create exclusion range from each IP address range. Creating one or more exclusion ranges prevents the DHCP server from assigning a client lease with any address in the exclusion range, thereby protecting it for use as a static IP address and preventing address conflicts between statically configured devices and dynamically configured devices.

Some of the excluded address on my R2 topoplogy are :
R2(config)# IP dhcp excluded-address 192.168.100.1 192.168.100.19
R2(config)# IP dhcp excluded-address 10.0.0.1 10.0.0.20

5. Can R1's f0/0 interface communicate with R3's f0/1 interface? If yes, how? Make this work without adding any static routes on any of the routers. Paste screenshots of what you did to make it work and the successful ping. **[3 points]**

R1's f0/0 interface communicate with R3's f0/1 interface as they are connected over a Layer-3 default-gateway configured with the appropriate static routes to each network with same subnets. The screenshot of the successful ping is attached

```
R2#
R2#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.100.1   YES manual up                    up
FastEthernet1/0        10.0.0.1        YES manual up                    up
R2#sh run
Building configuration...

Current configuration : 1087 bytes
!
! Last configuration change at 05:11:40 UTC Mon Nov 7 2022
```

```
R1#sh ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.100.2   YES DHCP   up                    up
R1#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 = 52/59/64 ms
R1#
```

6. Explain four differences between TCP and UDP. Mention two advantages and disadvantages of both. **[5 points]**

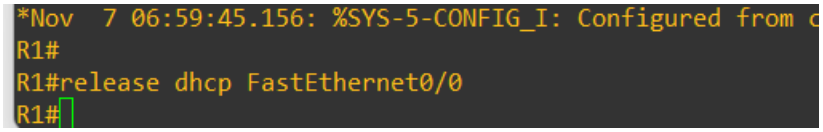
TCP (Transmission Control Protocol)	UCP (User Datagram Protocol)
Connection-oriented Protocol	Connection-less protocol
Sends data in particular sequential order	No fixed order
Leverages Flow-control	UDP doesn't use any flow-control techniques
supports only point-to-point transmissions only	Supports both multicast and broadcast
Variable length header upto 60 bytes	Fixed-length header with only 8 bytes
Congestional-control for packet-integrity	No Copngestion-control, drops packets if network is congested
Uses Checksum, timeout, acknowledgment to prevent and correct error	uses only checkdum to prevent errors and cannot correct error

Advantages of TCP	Advantages of UDP
It is a scalable, client-server architecture. This allows networks to be added without disrupting the current services	Connection-less so latency is low and ensures faster packet transmission suitable for time-sensitive applications such as VoIP, Streaming
Three-way handshake for data-reliability and ensures data-security	Suitable for broadcasts, UDP Broadcasts can be received by large number of clients without server-side overhead

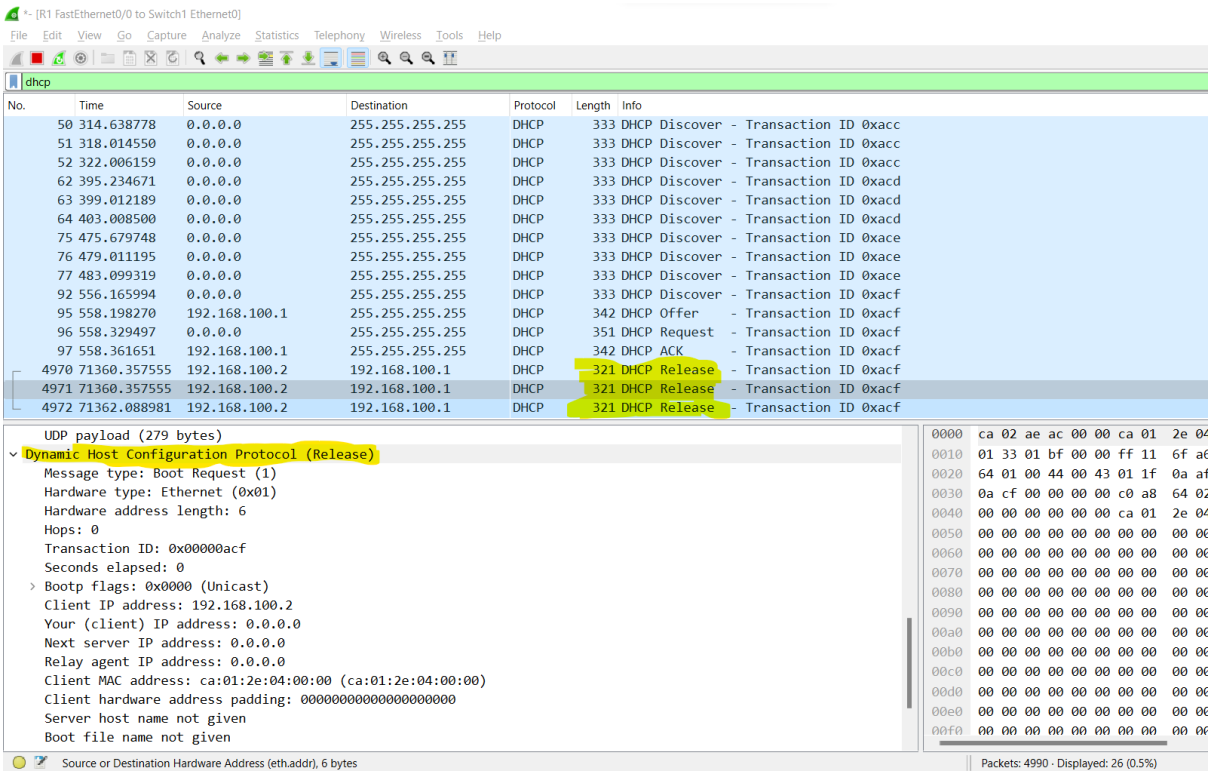
It operates independently of the operating system	UDP's speed makes it useful for query-response protocols such as DNS, in which data packets are small and transactional.
Disadvanatges of TCP	Disadvantages of UDP
TCP is comparatively slower than UDP making it ineffcient for streaming applications	There is no flow control and no acknowledgement for received data or lost data
designed for a wide area network (WAN). It has not been optimized for LAN (local area network) and personal area network (PAN)	UDP has no windowing and no function to ensure data is received in the same order as it was transmitted

7. Release the DHCP IP from the client R1. What command did you use? Paste the screenshot of the packet capture on Wireshark where these DHCP messages are captured. **[5 points]**

   Ans :: Release DHCP <Interface>



   Paste the screenshot of the packet capture on Wireshark where these DHCP messages are captured. **[5 points]**



8. Can the server also retrieve the DHCP IP back from the client before the lease time is over? If yes, what command can you use on the server to do this? **[5 points]**

Ans :: Yes. Renew DHCP <Int type>

```
R1#release dhcp FastEthernet0/0
R1#renew dhcp FastEthernet0/0
R1#sh ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        unassigned      YES DHCP   up                    up
R1#
*Nov  7 07:04:13.120: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned
 DHCP address 192.168.100.3, mask 255.255.255.0, hostname R1

R1#sh ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.100.3   YES DHCP   up                    up
R1#
```

9. Now turn on DHCP debugging on R1 and R3. What commands did you use? **[3 points]**

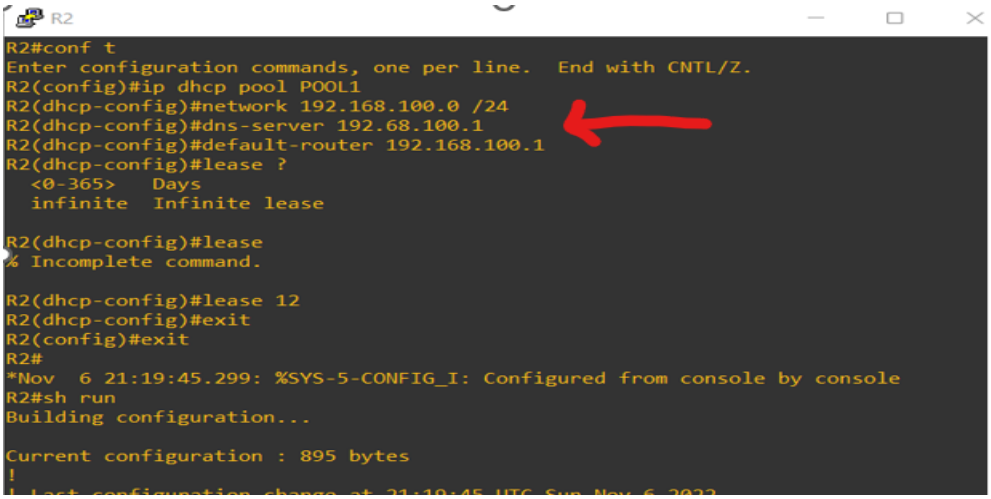
Ans :: DEBUG IP DHCP SERVER PACKET DEBUG IP DHCP PACKET

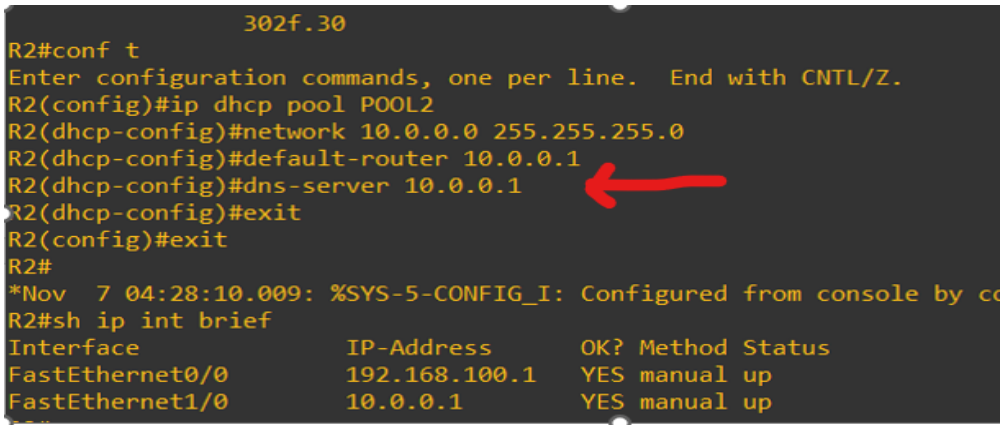
```
R1#debug ip packet
IP packet debugging is on
R1#debug ip dhcp server packet
DHCP server packet debugging is on.
R1#
```

```
R3#debug ip dhcp server packet
DHCP server packet debugging is on.
R3#debug ip packet
IP packet debugging is on
R3#
```

10. Update the configuration on R2 to provide extra DHCP option for DNS. The DNS server you are using should be R2 itself. Include the appropriate IP address(es) to use in the DHCP configuration. Paste a screenshot of updated DHCP configuration on R2.
[10 points]

DNS config for R1(client)





11. After DHCP is successful, paste screenshots of debug messages you captured on R1 and R3 indicating the success. **[10 points]**

R3 Capture

R3

```
*Nov  7 17:57:11.395: DHCP: Try 1 to acquire address for FastEthernet0/0
*Nov  7 17:57:11.491: DHCP: allocate request
*Nov  7 17:57:11.495: DHCP: new entry. add to queue
*Nov  7 17:57:11.499: DHCP: SDiscover attempt # 1 for entry:
*Nov  7 17:57:11.499: DHCP: SDiscover: sending 291 byte length DHCP packet
*Nov  7 17:57:11.499: DHCP: SDiscover 291 bytes
*Nov  7 17:57:11.503: IP: s=0.0.0.0 (local), d=255.255.255.255 (FastEthernet0/0)
, len 319, sending broad/multicast
*Nov  7 17:57:11.507: IP: s=0.0.0.0 (local), d=255.255.255.255 (FastEthernet0/0)
R3#, len 319, sending full packet
*Nov  7 17:57:11.511:              B'cast on FastEthernet0/0 interface from 0.0.0
.0
*Nov  7 17:57:11.567: IP: s=10.0.0.1 (FastEthernet0/0), d=10.0.0.2, len 56, inpu
t feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FA
LSE
R3#
*Nov  7 17:57:13.059: IP: s=10.0.0.1 (FastEthernet0/0), d=10.0.0.2, len 56, inpu
t feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FA
LSE
*Nov  7 17:57:13.555: IP: s=10.0.0.1 (FastEthernet0/0), d=255.255.255.255, len 3
28, input feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu 0, f
wdchk FALSE
*Nov  7 17:57:13.555: IP: s=10.0.0.1 (FastEthernet0/0), d=255.255.255.255, len 3
28, rcvd 1
*Nov  7 17:57:13.555: DHCP: Received a BOOTREP pkt
*Nov  7 17:57:13.555: DHCP: offer received from 10.0.0.1
*Nov  7 17:57:13.559: DHCP: SRequest attempt # 1 for entry:
*Nov  7 17:57:13.559: DHCP: SRequest- Server ID option: 10.0.0.1
*Nov  7 17:57:13.559: DHCP: SRequest- Requested IP addr option: 10.0.0.2
*Nov  7 17:57:13.563: DHCP: SRequest placed lease len option: 86400
*Nov  7 17:57:13.563: DHCP: SRequest: 309 bytes
*Nov  7 17:57:13.563: DHCP: SRequest: 309 bytes
*Nov  7 17:57:13.567: IP: s=0.0.0.0 (local), d=255.255.255.255 (FastEthernet0/0)
, len 337
R3#, sending broad/multicast
*Nov  7 17:57:13.571: IP: s=0.0.0.0 (local), d=255.255.255.255 (FastEthernet0/0)
, len 337, sending full packet
*Nov  7 17:57:13.571:              B'cast on FastEthernet0/0 interface from 0.0.0
.0
*Nov  7 17:57:13.603: IP: s=10.0.0.1 (FastEthernet0/0), d=255.255.255.255, len 3
28, input feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu 0, f
wdchk FALSE
*Nov  7 17:57:13.603: IP: s=10.0.0.1 (FastEthernet0/0), d=255.255.255.255, len 3
28, rcvd 1
*Nov  7 17:57:13.603: DHCP: Received a BOOTREP pkt
R3#
*Nov  7 17:57:17.671: DHCP Client Pooling: ***Allocated IP address: 10.0.0.2
*Nov  7 17:57:17.715: Allocated IP address = 10.0.0.2  255.255.255.0
```

R1 Capture:

R1

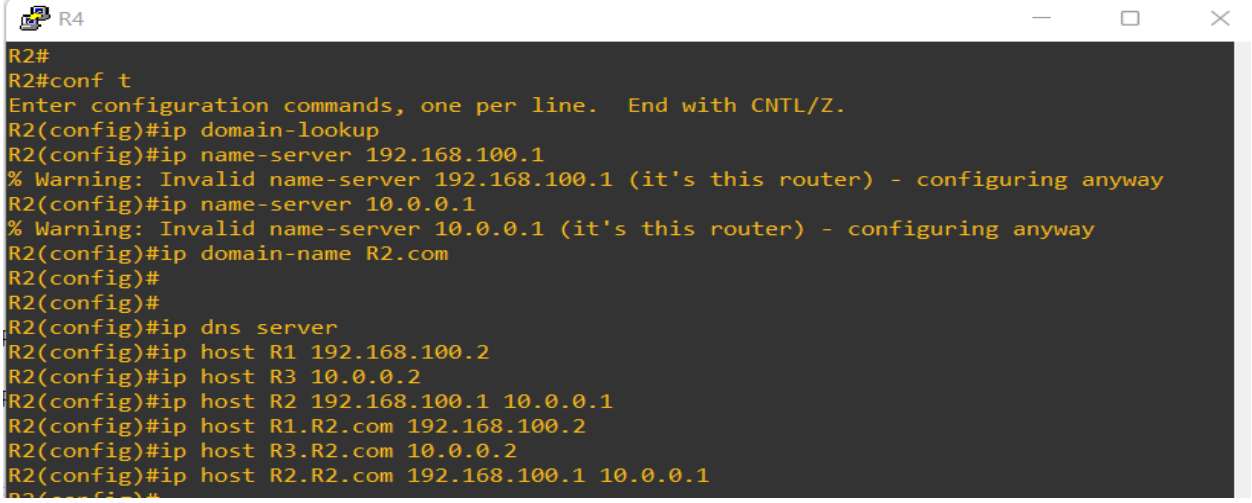
```
R1#
*Nov  7 07:57:21.216: DHCP: SDiscover attempt # 1 for entry:
*Nov  7 07:57:21.216: DHCP: SDiscover: sending 291 byte length DHCP packet
*Nov  7 07:57:21.220: DHCP: SDiscover 291 bytes
*Nov  7 07:57:21.220: IP: s=0.0.0.0 (local), d=255.255.255.255 (FastEthernet0/0)
, len 319, sending broad/multicast
*Nov  7 07:57:21.224: IP: s=0.0.0.0 (local), d=255.255.255.255 (FastEthernet0/0)
, len 319, sending full packet
*Nov  7 07:57:21.224:              B'cast on FastEthernet0/0 interface from 0.0.0
.0
R1#
*Nov  7 07:57:23.244: IP: s=192.168.100.1 (FastEthernet0/0), d=255.255.255.255,
len 328, input feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu
 0, fwdchk FALSE
*Nov  7 07:57:23.248: IP: s=192.168.100.1 (FastEthernet0/0), d=255.255.255.255,
len 328, rcvd 1
*Nov  7 07:57:23.256: DHCP: Received a BOOTREP pkt
*Nov  7 07:57:23.256: DHCP: offer received from 192.168.100.1
*Nov  7 07:57:23.260: DHCP: SRequest attempt # 1 for entry:
*Nov  7 07:57:23.260: DHCP: SRequest- Server ID option: 192.168.100.1
*Nov  7 07:57:23.260: DHCP: SRequest- Requested IP addr option: 192.168.100.4
*Nov  7 07:57:23.264: DHCP: SRequest placed lease len option: 1036800
*Nov  7 07:57:23.264: DHCP: SRequest: 309 bytes
*Nov  7 07:57:23.264: DHCP: SRequest: 309 bytes
*Nov  7 07:57:23.268: IP: s=0.0.0.0 (local), d=255.255.255.255 (FastEthernet0/0)
, len 337, sending broad/multicast
*Nov  7 07:57:23.276: IP: s=0.0.0.0 (local), d=255.255.255.255 (FastEthernet0/0)
R1#en 337, sending full packet
*Nov  7 07:57:23.276:              B'cast on FastEthernet0/0 interface from 0.0.0
.0
*Nov  7 07:57:23.292: IP: s=192.168.100.1 (FastEthernet0/0), d=255.255.255.255,
len 328, input feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu
 0, fwdchk FALSE
*Nov  7 07:57:23.292: IP: s=192.168.100.1 (FastEthernet0/0), d=255.255.255.255,
len 328, rcvd 1
*Nov  7 07:57:23.296: DHCP: Received a BOOTREP pkt
*Nov  7 07:57:24.092: DHCPD: interface FastEthernet0/0 coming up
R1#
*Nov  7 07:57:26.324: DHCPD: IP address change on interface FastEthernet0/0
*Nov  7 07:57:26.356: DHCPD: IP address change on interface FastEthernet0/0
R1#
*Nov  7 07:57:27.364: DHCP Client Pooling: ***Allocated IP address: 192.168.100.4
*Nov  7 07:57:27.408: Allocated IP address = 192.168.100.4  255.255.255.0

R1#
*Nov  7 07:57:27.408: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned DHCP address 192.1

R1#
*Nov  7 07:57:29.300: DHCPD: interface FastEthernet0/0 coming up
```

Objective-3: Getting started with DNS

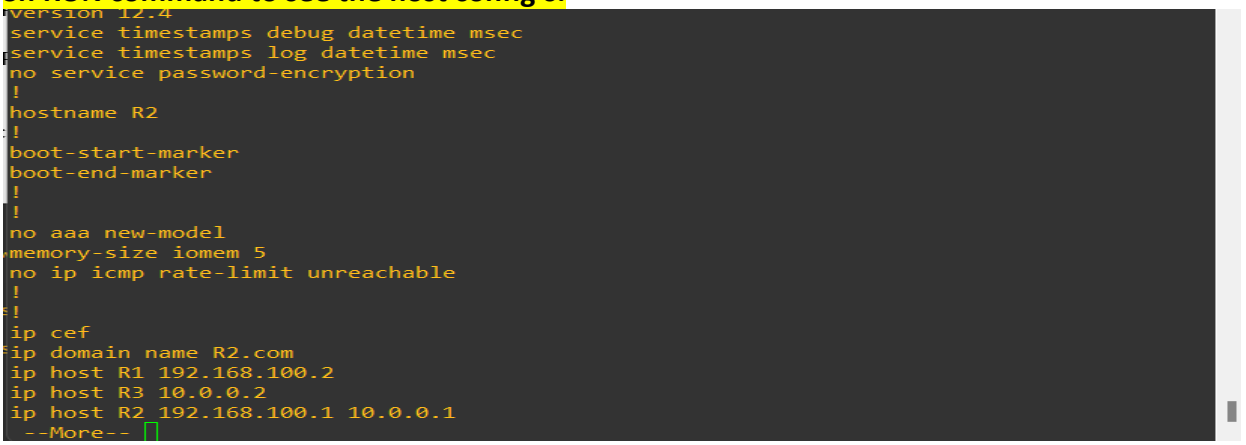
1. Now configure R2 as the DNS server. Below are the mappings you will add on the DNS server:

Hostname	IP address
R1	R1's interface IP
R2	R2's interface IP

Paste a screenshot of the configuration on R2 indicating the hostname configurations. **[10 points]**



Sh RUN command to see the host config's.



2. To implement DNS, do you need any additional configuration on R1 and R3? If yes, explain and paste screenshots. If not, explain. **[5 points]**

YES, we need to configure DNS-Client R1, R3 with "ip domain-lookup", "name-server", "domain-name" of DNS-SERVER in this case, R2.

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip domain-lookup
R1(config)#ip name-server 192.168.100.1
R1(config)#ip domain-name R2.com
R1(config)#end
R1#sh
*Nov  7 09:28:10.164: %SYS-5-CONFIG_I: Configured from console by console
R1#sh run
```

```
R3(config)#ip domain-lookup
R3(config)#ip name-server 10.0.0.1
R3(config)#ip domain-name R2.com
R3(config)#
R3(config)#end
R3#sh
*Nov  7 19:31:37.106: %SYS-5-CONFIG_I: Configured from console by console
```

3. If DNS is successfully configured, from R1 you should be able to issue the command "ping R2" and on R2 use the command "ping R1". Show screenshots of the ping working. **[10 points]**

From "R1" pinging "R2"

```
R1#ping R2
Translating "R2"...domain server (192.168.100.1) [OK]

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.100.1, timeout is 2 seconds:
!!!!
*Nov  7 09:44:29.520: IP: tableid=0, s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), routed via RIB
*Nov  7 09:44:29.520: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 55, sending
*Nov  7 09:44:29.520: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 55, sending full packet
*Nov  7 09:44:29.588: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100.2, len 87, input feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk
*Nov  7 09:44:29.592: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100.2, len 87, rcvd 1
*Nov  7 09:44:29.600: IP: tableid=0, s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), routed via RIB
*Nov  7 09:44:29.600: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 100, sending
*Nov  7 09:44:29.600: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 100, sending full packet
*Nov  7 09:44:29.620: IP: s=192.168.100.1 (FastEthernet0/0), d=1!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/33/52 ms
R1#92.168.100.2, len 100, input feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Nov  7 09:44:29.624: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100.2, len 100, rcvd 1
*Nov  7 09:44:29.628: IP: tableid=0, s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), routed via RIB
*Nov  7 09:44:29.628: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 100, sending
*Nov  7 09:44:29.628: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 100, sending full packet
*Nov  7 09:44:29.660: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100.2, len 100, input feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdch
*Nov  7 09:44:29.660: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100.2, len 100, rcvd 1
*Nov  7 09:44:29.664: IP: tableid=0, s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), routed via RIB
*Nov  7 09:44:29.664: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEth
R1#ernet0/0), len 100, sending
*Nov  7 09:44:29.672: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 100, sending full packet
*Nov  7 09:44:29.712: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100.2, len 100, input feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdch
*Nov  7 09:44:29.712: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100.2, len 100, rcvd 1
*Nov  7 09:44:29.712: IP: tableid=0, s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), routed via RIB
*Nov  7 09:44:29.712: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 100, sending
*Nov  7 09:44:29.712: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 100, sending full packet
*Nov  7 09:44:29.728: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100.2, len 100, input feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdch
*Nov  7 09:44:29.732: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100
R1#.2, len 100, rcvd 1
*Nov  7 09:44:29.740: IP: tableid=0, s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), routed via RIB
*Nov  7 09:44:29.740: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 100, sending
*Nov  7 09:44:29.740: IP: s=192.168.100.2 (local), d=192.168.100.1 (FastEthernet0/0), len 100, sending full packet
*Nov  7 09:44:29.768: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100.2, len 100, input feature, MCI Check(92), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdch
*Nov  7 09:44:29.772: IP: s=192.168.100.1 (FastEthernet0/0), d=192.168.100.2, len 100, rcvd 1
R1#
```

From "R2" pinging "R1"

```
R2#
R2#
R2#
R2#ping R1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.100.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/32/44 ms
R2#ping R1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.100.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/30/48 ms
R2#
```

4. Initiate a Wireshark capture in your topology. Where would you initiate the capture? Paste a screenshot of the Wireshark capture of the DNS messages that are exchanged when you issue either “ping R1” or “ping R2” command. **[10 points]**

DNS capture when issued “Ping R2” from R1.

R1 source Address : 192.164.100.2 pininging destination R2 192.164.100.1

```
R1#00, sending full packet
R1#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.100.2   YES DHCP   up                    up
R1#
```

```
R2#sh ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.100.1   YES manual up                    up
FastEthernet0/1        10.0.0.1        YES manual up                    up
R2#
```

*- [Switch1 Ethernet1 to R4 FastEthernet0/0]

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

dns

No.	Time	Source	Destination	Protocol	Length	Info
20	58.480015	192.168.100.2	192.168.100.1	DNS	69	Standard query 0x0002 A R2.R2.com
21	58.495792	192.168.100.1	192.168.100.2	DNS	101	Standard query response 0x0002 A R2.R2.com A 192.168.100.1 A 10.0.0.1
45	75.767322	192.168.100.2	192.168.100.1	DNS	69	Standard query 0x0003 A R3.R2.com
46	75.781722	192.168.100.1	192.168.100.2	DNS	85	Standard query response 0x0003 A R3.R2.com A 10.0.0.2
171	769.246371	192.168.100.2	192.168.100.1	DNS	69	Standard query 0x0004 A R2.R2.com
172	769.306532	192.168.100.1	192.168.100.2	DNS	101	Standard query response 0x0004 A R2.R2.com A 192.168.100.1 A 10.0.0.1

```
> Ethernet II, Src: ca:01:2e:04:00:00 (ca:01:2e:04:00:00), Dst: cc:04:16:a8:00:00 (cc:04:1
v Internet Protocol Version 4, Src: 192.168.100.2, Dst: 192.168.100.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 55
    Identification: 0x0000 (0)
  > 000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 255
    Protocol: UDP (17)
    Header Checksum: 0x7261 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.100.2
    Destination Address: 192.168.100.1
v User Datagram Protocol, Src Port: 59877, Dst Port: 53
```

```
0000  cc 04 16 a8 00 00 ca 01  2e 04 00 00 08 00 45 00   ........ .....E·
0010  00 37 00 00 00 00 ff 11  72 61 c0 a8 64 02 c0 a8   ·7...... ra··d···
0020  64 01 e9 e5 00 35 00 23  ce dd 00 03 01 00 00 01   d····5·# ........
0030  00 00 00 00 00 00 02 52  33 02 52 32 03 63 6f 6d   .......R 3·R2·com
0040  00 00 01 00 01                                     .....
```

Domain Name System: Protocol

Packets: 254 · Displayed: 6 (2.4%)

Profile: Vo

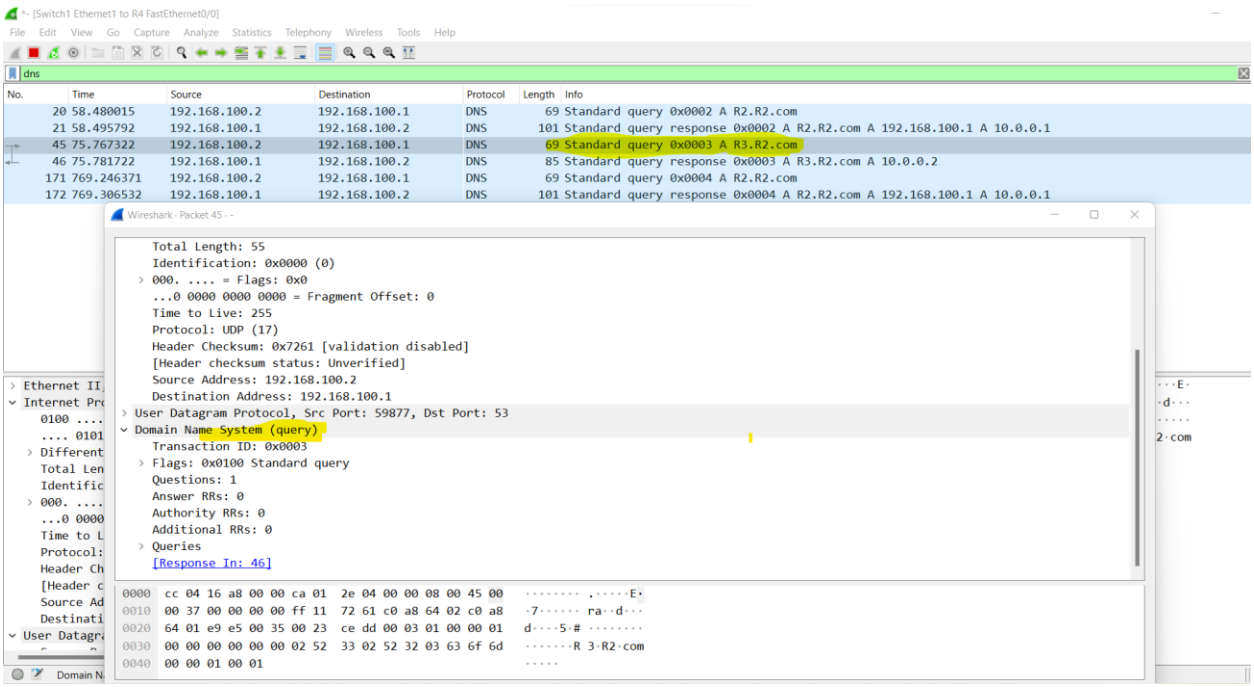
5. Explain in detail the sequence of DNS messages that are exchanged. **[8 points]**

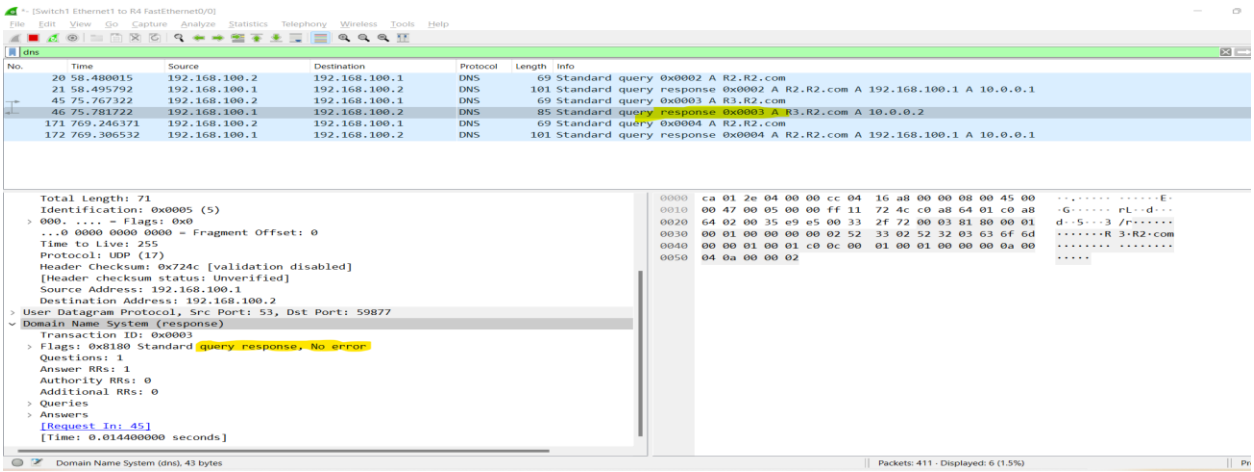
DNS resolves the IP Address into Domain Name by exchanging information between client and server using two type of messages :

Query/ Response Message :: The 69 Byte query frame constitutes of Header, Question – 16 bit field to specify the count of questions in the section. Answer RRs – 16 bit field has "0" value in query and its only valued in Response. Authoratative RRs – gives info about domain names and Additional Records is only available in Response messages.

The first host sends the DNS query message to its local DNS name-server server, with "Type" and "Class Field". The DNS server performs "IP DOMAIN-LOOK UP" Domain-resolution by first checking the local dns cache, if can't be resolved the query is sent to "Authorative Server" and response is sent back in "Response" with fields "Type", "Class IN", "addr 10.0.02"

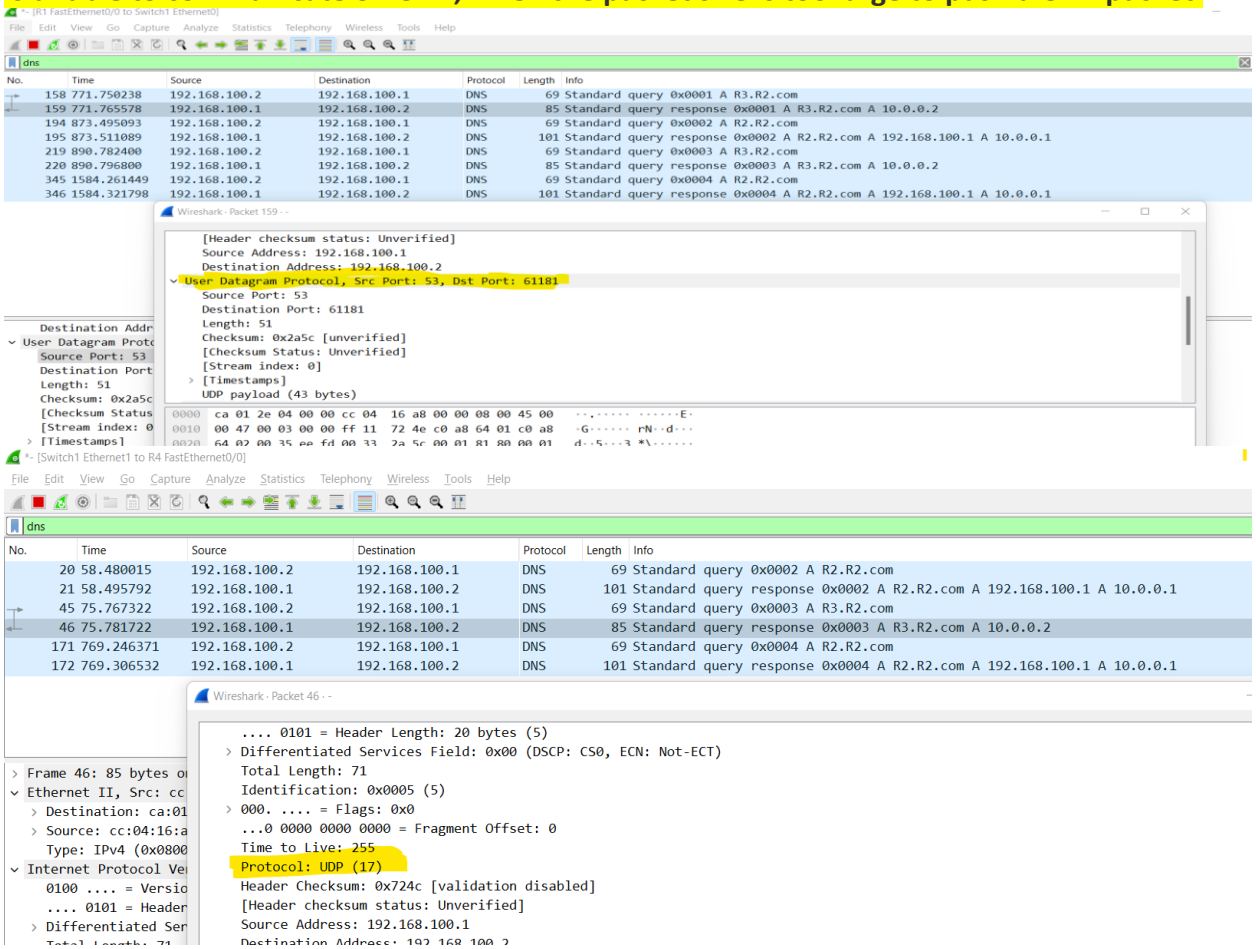
```
Additional RRs: 0
Queries
  R3.R2.com: type A, class IN
Answers
  R3.R2.com: type A, class IN, addr 10.0.0.2
[Request In: 45]
[Time: 0.014400000 seconds]
```





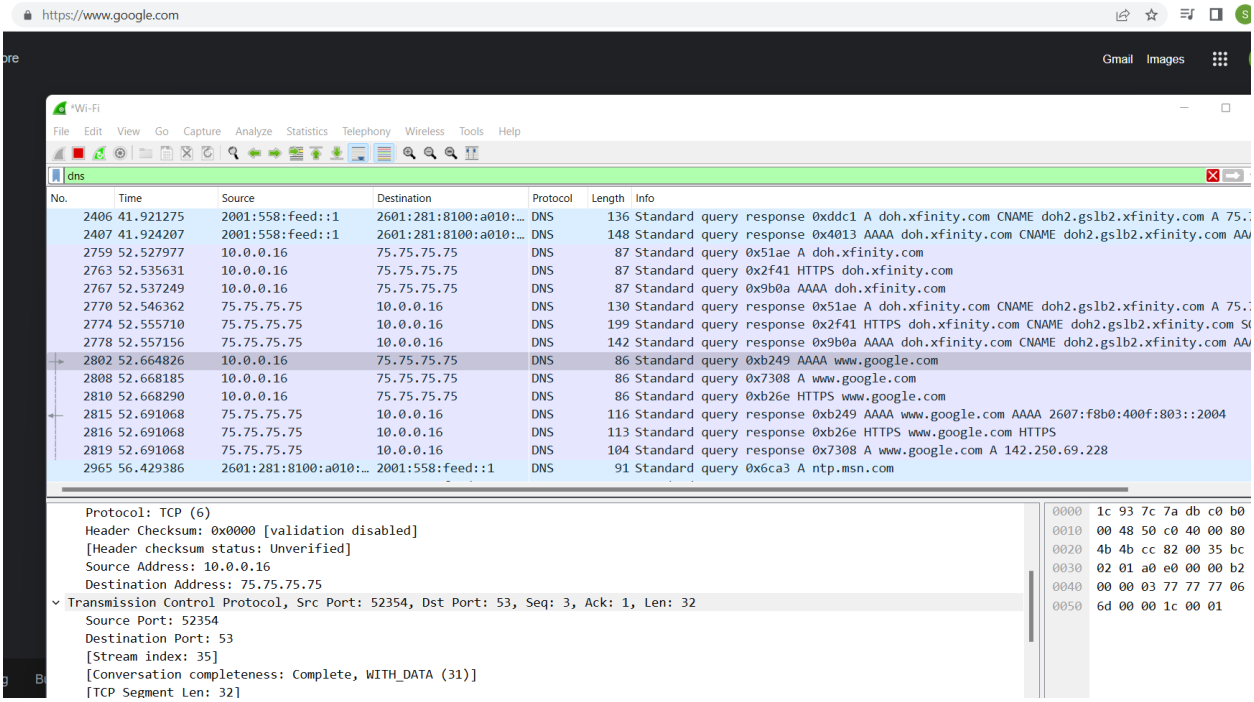
6. Did DNS use UDP or TCP as the transport layer protocol in this case? Will it ever use the other protocol? If yes, when? **[3 points]**

Ans: DNS has used UDP (User Datagram Protocol) in both Query / Response messages. And yes, while DNS is mostly UDP Port 53, DNS will also rely on TCP as a fall back when it is unable to communicate on UDP, when the packet size is too large to push a UDP packet.



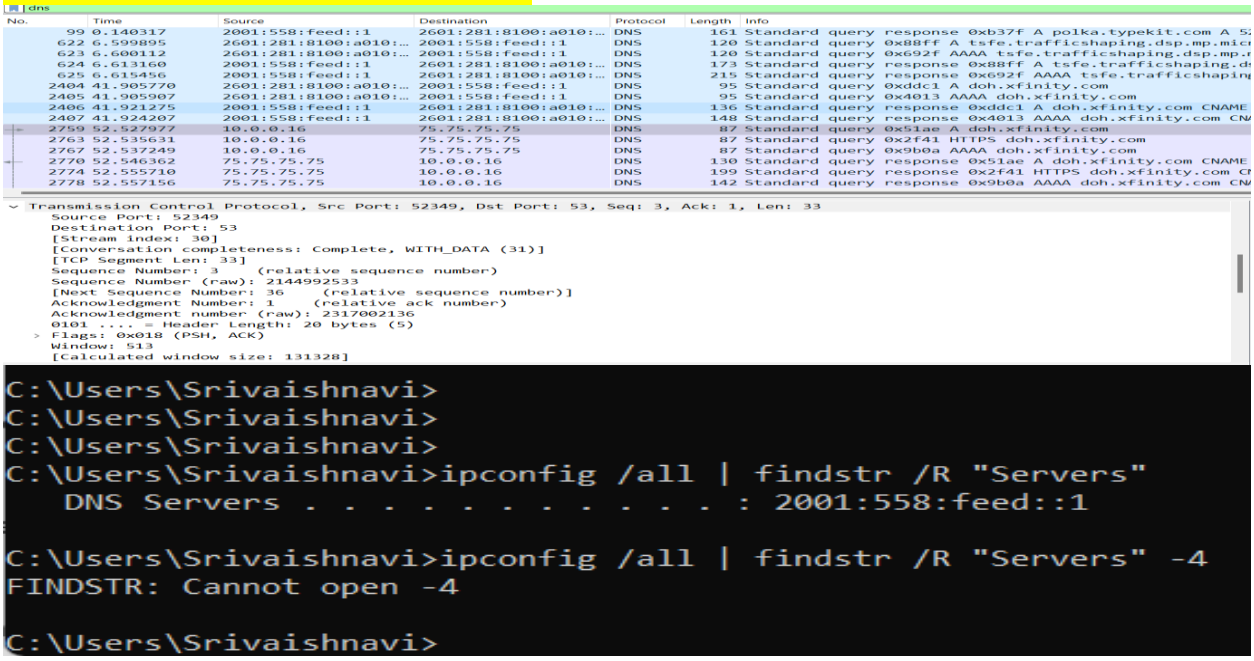
Objective-4: Report Questions

1. Run Wireshark on your laptop and start the capture on the interface going to the Internet. Ping www.google.com



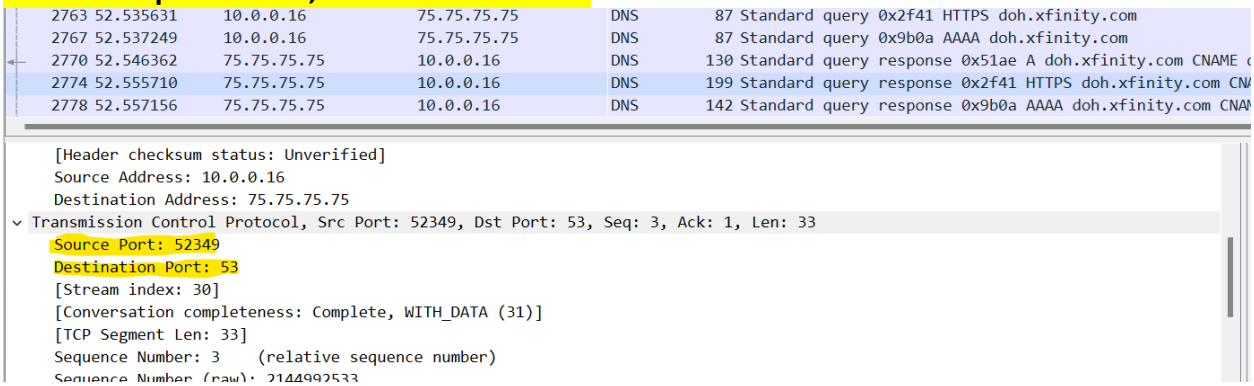
2. What IP is your laptop using as the DNS server? How do you know this? **[2 points]**

The DNS Source IP is :: 2001:558:feed::1



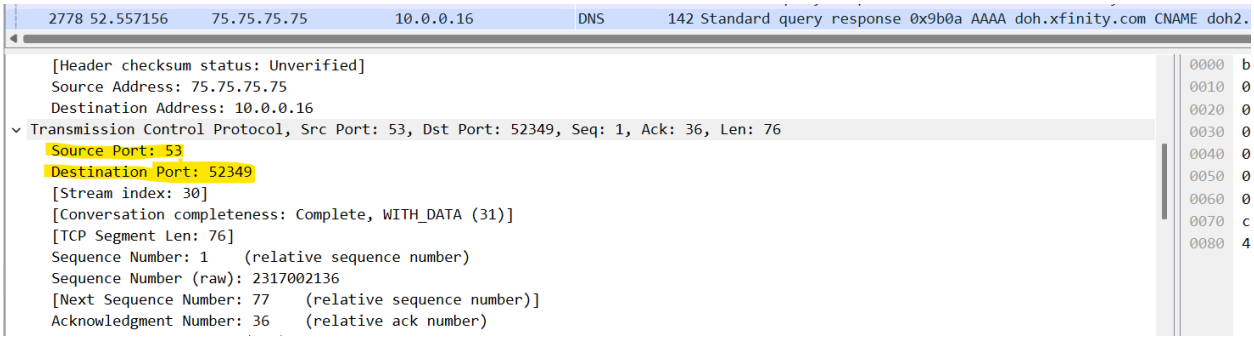
3. For a DNS query, what is the source and destination port numbers? **[2 points]**

The source port: 52349, Destination Port: 53

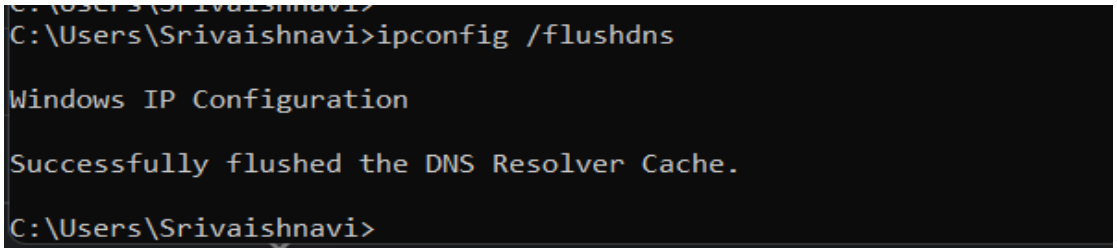


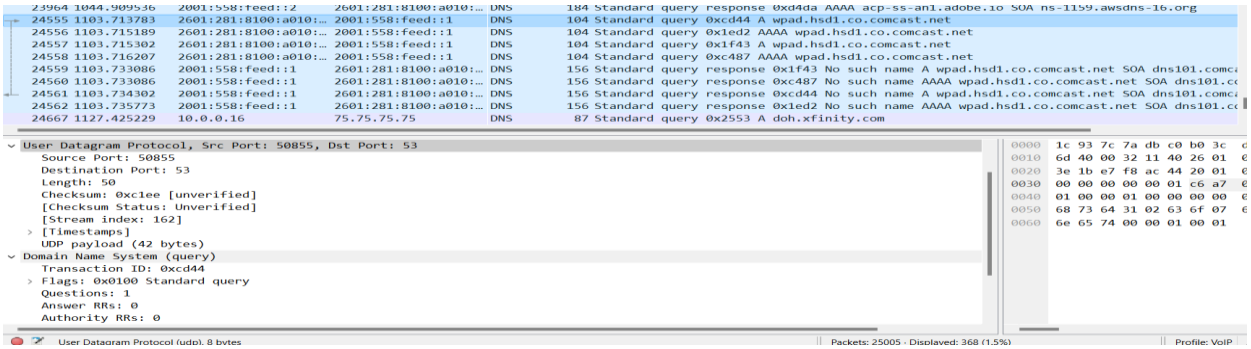
4. For a DNS response what is the source and destination port numbers? **[2 points]**

The source port :: 53, Destination Port:: 52349



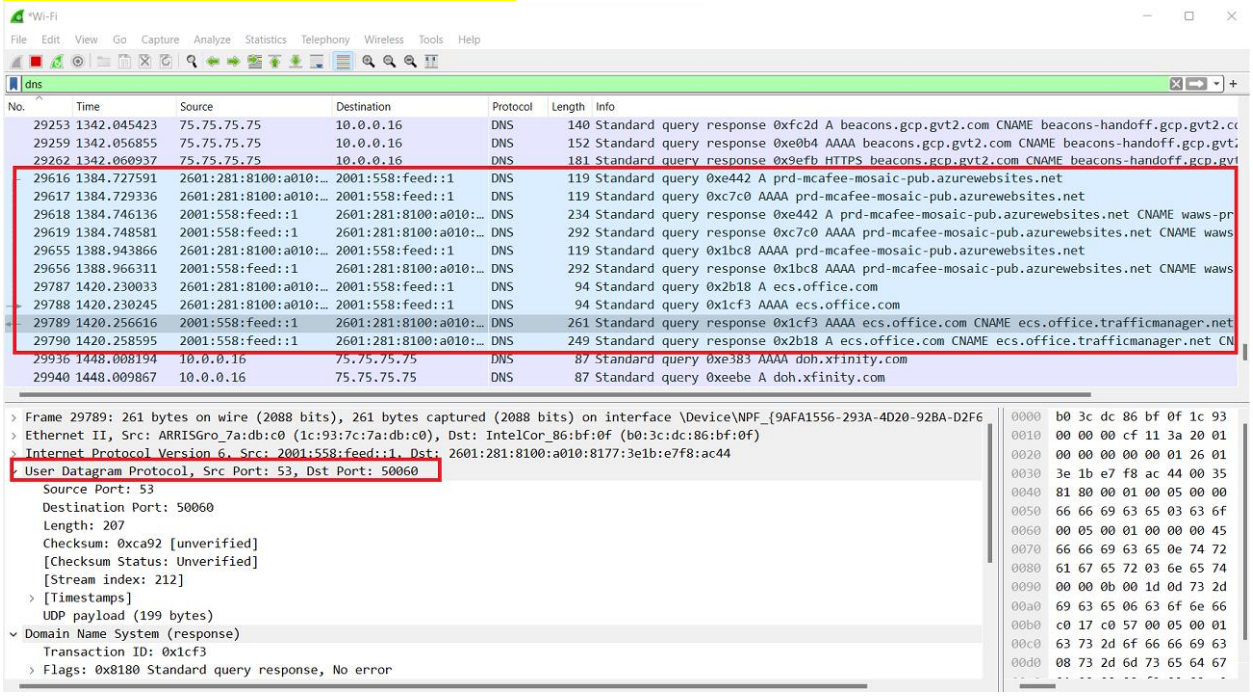
5. Clear any DNS cache on your laptop. How did you do this? Paste screenshot. **[2 points]**





6. Paste a screenshot of the DNS messages exchanged in this case. Did DNS use UDP or TCP in this case? **[2 points]**

It used UDP. (User Data Gram protocol)



7. You had your DNS cache cleared. Assume all DNS nameservers in the world have their DNS cache cleared. Now explain in theory how your DNS query is resolved. Assuming no caches exist, what levels of the hierarchy does the query need to propagate through to get resolved? Explain the sequence of events and the flow. **[10 points]**

if the DNS Cache is cleared, and the OS has no record of the website we are trying to resolce, the request will be sent to the "**DNS RecursiveResolver**", before sending the request to the rest of the DNS hierarchy, the resolver tries to resolve the query by checking its local cache, it it doesn't have cached entry then it forwards the packet to "**DNS ROOT SERVERS**"

Root servers are on the top DNS hierarchy in resolving uncached DNS queries. The root server will look at the top-level domain portion of the DNS query and provide the address of the appropriate **top-level domain server**. For example, if its google.com, the root server would look at the top-level domain ".com," and it would provide the address to the .com top-level domain name server.

TLD servers are the next in the DNS hierarchy. These contain the information for the domain's authoritative nameserver. In google.com, the .com TLD server would have the address of the authoritative name server for the domain "google".

Authoritative name servers are the last level of the DNS hierarchy and contain the actual DNS records for a group of domains. These servers give authoritative responses to DNS queries. And for "google.com" authoritative name server for would return the address to the recursive resolver server. The recursive resolver server would then return the address to the client machine after completing the request.

8. Explain briefly the different type of DNS records. **[5 points]**
   DNS record types provide important information about a hostname or domain including the current IP address for a domain.
   The 5 major DNS Record types are :

- **A record – is for "address" and shows the IP address for a specific hostname/domain**
- **AAAA record – points for a "IPV6 address"**
- **CNAME record – is "canonical name", and points a domain name (an alias) to another domain**
- **NS Record -- specifies the authoritative DNS server for a domain**
- **Mail exchange (MX) record -- shows where emails for a domain should be routed to. Basically, an MX record makes it possible to direct emails to a mail server**

9. What are the different types of DNS nameservers? Could you configure your laptop to be a DNS server too? If yes, explain what type of DNS nameserver or nameservers it can be. **[5 points]**
   Types include :: Root Name Servers, TLD Name Servers, Authoritative Name Servers. And also Primary servers, Secondary Servers and Caching servers.
   Yes, its quite possible to configure laptop to be a DNS server, incase of windows is as easy as editing a file. But only works on that computer. It can be either a Primary or Secondary Root Name Server/ TLD server.

10. Explain briefly the HTTP Error Messages with their status code. **[2 points]**
    There is two types of HTTP Error messages 1) Client Errors, 2)Server Errors

400 – Bad Request	**500 – Internal Server Error**
403 – Forbidden	**502 – Bad Gateway**
404 – Page Not Found	**503 – Service Unavailable**
408 – Request Time Out	**504 – Gateway Timeout**
407 – Proxy Authentication Required	**505 – HTTP Version not supported**

11. Explain briefly the different types of HTTP requests. **[2 points]**
    The most used HTTP Request methods are
    1. **GET, 2. POST, 3. PUT, 4. PATCH, 5. DELETE**

12. What is a proxy web server? Mention any four advantages of using a proxy webserver. **[3 points]**

Proxy web server is a gateway between users and internet acting as a filter/firewall providing enhanced security and also cache data to speed up the HTTP common requests, and
1.Anonymity – masks the IP address there by hides the real ip address for security
2. Protection – Protects the users from accessing the malicious software on internet
3. Perfromance – Increases the performance by caching the data

13. You are trying to connect to www.bbc.com but the page does not load and keeps buffering. So now you try to connect to www.cnn.com and the page loads relatively faster. Brainstorm four possible reasons that could have led to this scenario. Mention the steps you will follow to troubleshoot this. **[10 points]**

Possible reasons :

1. **Network Connection between the end user and your server. If the server is in London, but the user is in USA- the performance would be slow.**
2. **Because the www.bbc.com is buffering there is a possibility that the website may be hosted on shared configured with UDP , sharing resources with other websites resulting in slower response time or the huge network traffic is causing packet drop**
3. **May be the website is not using caching. Caching can help to improve website performance by storing frequently accessed data in memory, so that it can be quickly accessed when needed.**
4. **May be the DNS server of the website is down so the domain-name lookup is hindered causing the page to not load**
5. **May be the website is using a content management system (CMS) that is not optimised for performance which lead to possibly slower response times as the CMS has to process and load each page every time it is accessed**

Steps to troubleshoot :

1. **I would clear the browser cache/ cookies that may be causing the problem**
2. **Try checking the working of the same url in the different browser/system**
3. **I would perform the "ping" / "tracert" to check the network connectivity with the website, the "tracert" command gives a clear indication of the packet hop between routers and TTL to find the point of packet drop/latency**
4. **Then I would perform DNS ns-Lookup , and if the Dns-Look up fails I would check with the ISP as to if the network is configuration supports the website functionality or not.**

What is a major disadvantage in using HTTP? What is another protocol that would solve this issue? **[2 points]**

Major disadvantage of HTTP is it is less-secure. And that's why HTTPS is used, it is a protocol which uses an encrypted HTTP connection by transport-layer security by sending information via HTTPs is done via the Secure Socket Layer (SSL),

Objective-5: DHCP Relay- Extra Credit [+20]

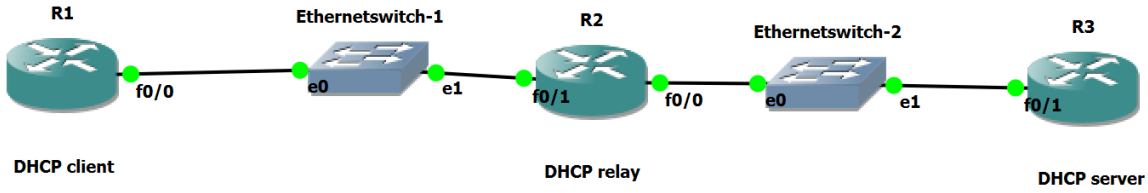


Fig.3

1. What is a DHCP relay? When would you use one? **[2 points]**
   A DHCP relay agent is a host or router that forwards DHCP packets between DHCP clients and DHCP servers when the server is present in a different network/ or to provide DHCP services across different broadcast domains. Relay agents, regenerate the DHCP messages to send out on other interfaces.

2. Clear any previous configurations on your topology. Setup the topology shown in Fig3. Initiate a Wireshark capture on Switch-1 and a simultaneous Wireshark capture on Switch-2.

3. In this case, R1 should be configured as a DHCP client to get its IP from R3 which is the DHCP server. R2 is the DHCP relay.

4. Paste screenshot of DHCP and interface configurations on R1, R2 and R3 that will work. (Hint: show run | begin ip dhcp and sh ip int br) **[5 points]**

R1 :

```
R1#
*Nov  7 07:04:13.120: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned
 DHCP address 192.168.100.3, mask 255.255.255.0, hostname R1

R1#sh ip int brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.100.3   YES DHCP   up                    up
```

R2 :

R2

```
!
!
!
interface FastEthernet0/0
 ip address 192.168.10.30 255.255.255.0
 ip helper-address 192.168.100.1
 duplex auto
 speed auto
!
interface FastEthernet0/1
 ip address 192.168.100.20 255.255.255.0
 duplex auto
 speed auto
!
!
no ip http server
no ip http secure-server
ip forward-protocol nd
```

R3 :

```
#conf t
ter configuration commands, one per line.  End with CNTL/Z.
(config)#ip dhcp pool POOL1
(dhcp-config)#network 192.168.100.0 /24
(dhcp-config)#dns-server 192.68.100.1
(dhcp-config)#default-router 192.168.100.1
(dhcp-config)#lease ?
<0-365>   Days
infinite  Infinite lease

(dhcp-config)#lease
Incomplete command.

(dhcp-config)#lease 12
(dhcp-config)#exit
(config)#exit
#
ov  6 21:19:45.299: %SYS-5-CONFIG_I: Configured from console by console
#sh run
ilding configuration...

rent configuration : 895 bytes
```

5. Is the configuration on DHCP server and DHCP client same as before? Or did you have to do anything extra in this case? If yes, mention the extra configuration you had to do. **[3 points]**

   I had to add helper-address on relay agent.

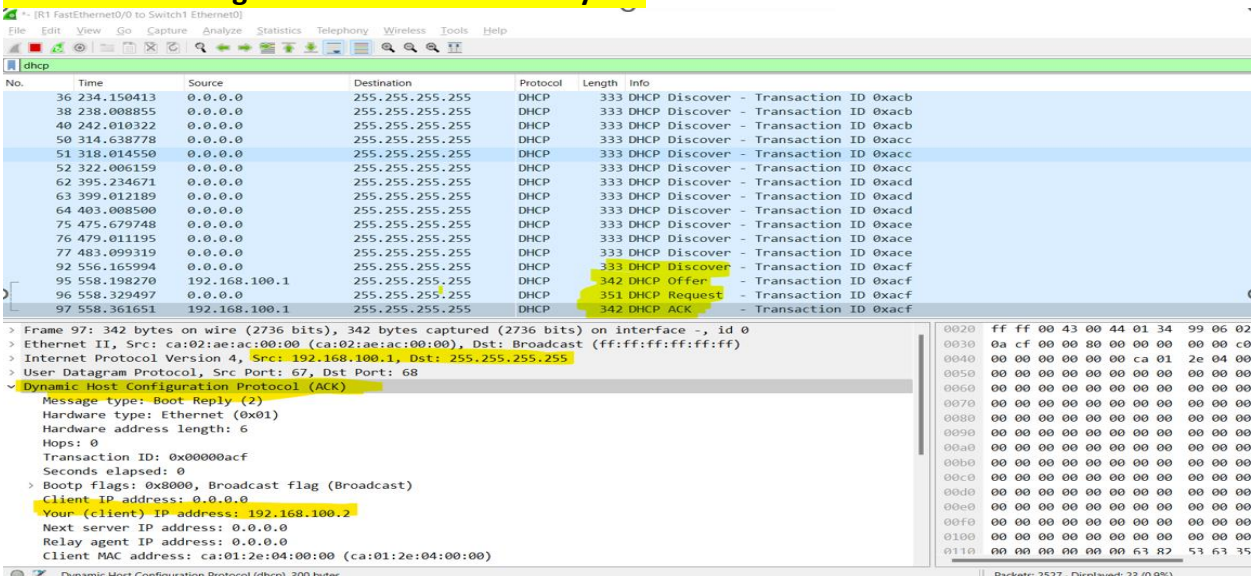
   Enter configuration commands, one per line. End with CNTL/Z.
   R2(config)#int fa0/0
   R2(config-if)#ip helper
   R2(config-if)#ip helper-address 192.168.100.1
   R2(config-if)#no shut
   R2(config-if)#end

6. After successful DHCP, examine the Wireshark capture.

Mention Source IP, Dest IP, Source MAC and Dest MAC of all 4 DHCP messages for the capture on the Ethernet-1 switch interface. Also note if each individual message is a broadcast or unicast message at Layer-2 and Layer-3. **[5 points]**

Source IP : 192.168.100.1 Dest IP: 255.255.255.255 Source Mac ::ca:02:ae:ac:00:00 Dest Mac :: ff:ff:ff:ff:ff:ff

DHCP Discover is a broadcast at Layer2, Layer3 , DHCP Offer is a broadcast at layer 3 as the server doesn't know client's IP Address, DHCP Request is a broadcast at layer3 DHCP Acknowledgment is a broadcast at Layer3.



7. Mention Source IP, Dest IP, Source MAC and Dest MAC of all 4 DHCP messages for the capture on the Ethernet-2 switch interface. Also note if each individual message is a broadcast or unicast message at Layer-2 and Layer-3. **[5 points]**
   Source IP : 192.168.10.10 Dest IP: 255.255.255.255
   Source Mac:: cc:01:f3:44:00:00 Dest Mac :: ff:ff:ff:ff:ff:ff
   DHCP Discover is a broadcast at Layer2, Layer3 , DHCP Offer is a Unicast layer 3 as the, DHCP Request is a Unicast at layer3
   DHCP Acknowledgment is a broadcast at Layer3.

Format:

	Src IP	Dest IP	Src MAC	Dest MAC	L2	L3
DHCP Discover	XXXX	YYYY	abcd	efgh	uni/broadcast	uni/broadcast

Score: _____ / 200 points [+20 points]