Wireshark Project-1

Wireshark Analysis of DHCP messages

In this project, I intend to observe messages exchanged between client and DHCP server while requesting and assigning of dynamic Ip address using Wireshark analysis tool.

Wireshark:

Wireshark is a popular and powerful open-source network protocol analyzer (packet sniffer) used for network troubleshooting, analysis, software development, and education. It allows you to capture and inspect the data packets traveling over a network in real-time or from saved capture files. Here are some key features and uses of Wireshark:

- Packet Capture
- Packet Analysis
- Exporting Data
- Statistics
- Scripting and Customization
- Cross-platform

DHCP:

In networking, DHCP (Dynamic Host Configuration Protocol) is a protocol that allows devices on a network to obtain IP addresses and other network configuration parameters dynamically from a DHCP server. The DHCP process involves a series of messages known as DORA, which stands for Discover, Offer, Request, and Acknowledge. These messages are part of the DHCP handshake process between a client and a DHCP server. Here's a brief overview of each message:

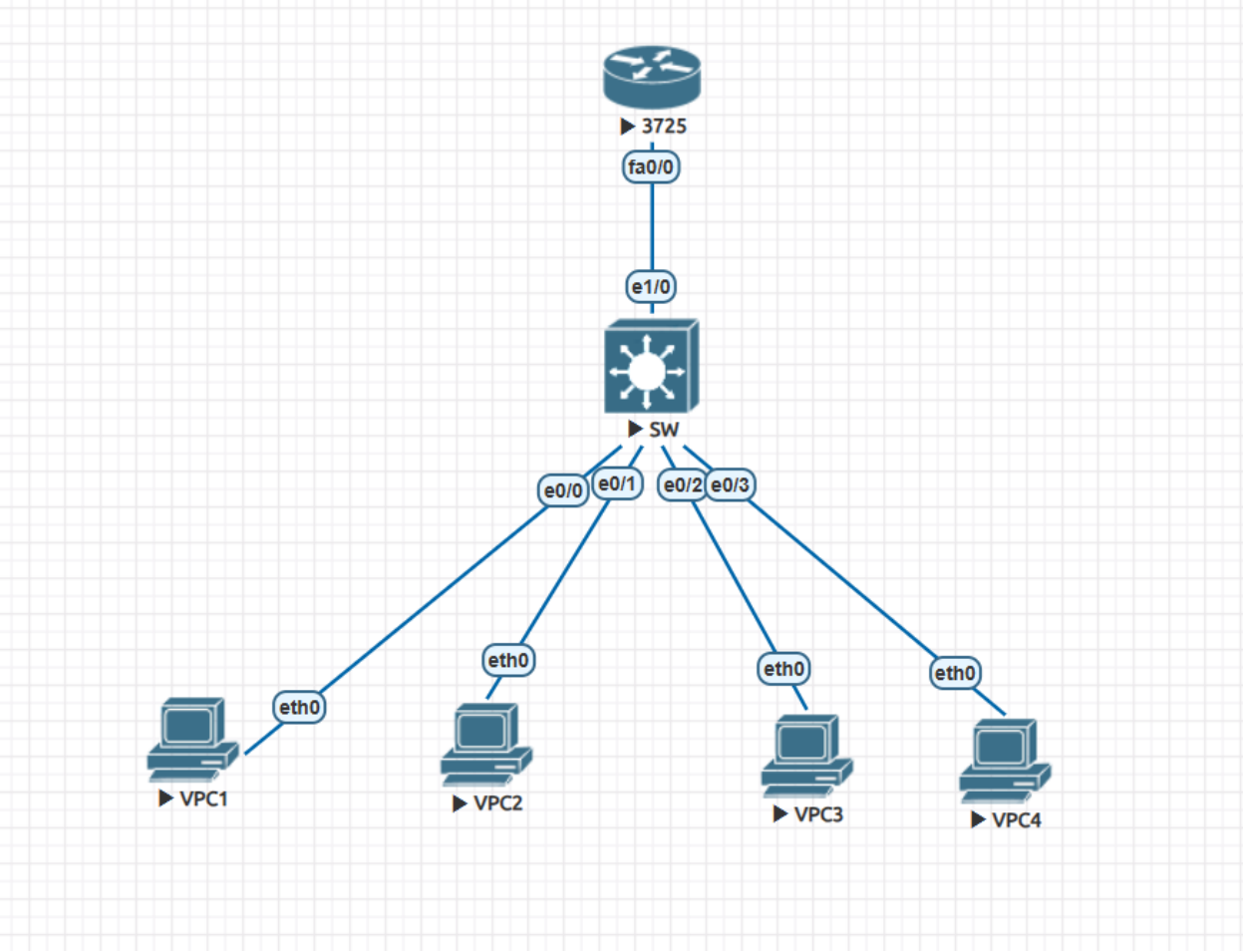
1. **Discover (D):**
    - When a client device connects to a network, it sends out a DHCP Discover message as a broadcast to discover available DHCP servers.

- The message does not contain an IP address because the client does not yet have one.

2. **Offer (O):**
    - DHCP servers on the network that receive the Discover message may respond with a DHCP Offer message.
    - The Offer message includes an available IP address, lease duration, subnet mask, and other network configuration options.
    - Multiple DHCP servers may respond with offers, but the client typically accepts the first one it receives.
3. **Request (R):**
    - After receiving one or more Offer messages, the client sends a DHCP Request message to the chosen DHCP server.
    - The Request message indicates the client's acceptance of the offered IP address.
    - If the client received multiple offers, it selects one and sends a Request for that specific offer.
4. **Acknowledge (A):**
    - The DHCP server that received the Request message confirms the allocation of the IP address to the client.
    - It sends a DHCP Acknowledgment (Ack) message to the client, finalizing the lease and providing any additional configuration parameters.
    - If there is an issue with the Request or the DHCP server cannot fulfill the request, it may send a DHCP NACK (Negative Acknowledgment) instead.

After the client receives the DHCP Ack message, it configures its network interface with the provided IP address and other settings. The client and server will periodically renew the lease as it nears expiration, ensuring that the client continues to have a valid IP address while on the network.

Let’s Build a small lab in EVE NG platform



Nodes:

- Virtual PC's
- Router
- Multilayer Switch

Step 1: Configure the Switch with Vlan 10, 20 and allocate unused ports with two PC's to each Vlan.

```
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#name Sales
Switch(config-vlan)#ex
Switch(config)#vlan 20
Switch(config-vlan)#name Operations
Switch(config-vlan)#ex
```

- Assign Ethernet Ports e0/0 and e0/1 to Vlan 10 and simultaneously to Vlan 20

```
Switch(config)#int range e0/0-1
Switch(config-if-range)#sw
Switch(config-if-range)#switchport mode ac
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#sw
Switch(config-if-range)#switchport access vlan 10
Switch(config-if-range)#
```

```
Switch(config)#int range e0/2-3
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 20
Switch(config-if-range)#
Switch(config-if-range)#
```

- Check the assigned vlans using “show vlan brief” command

```
Switch#sh vlan bri

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Et1/0, Et1/1, Et1/2, Et1/3
10   Sales                            active    Et0/0, Et0/1
20   Operations                       active    Et0/2, Et0/3
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
Switch#
Switch#
```

Step 2: Create EthernetPort0/0 as Trunk port which allows tagged or untagged frames within respective vlan.

```
Switch(config)#int e0/0
Switch(config-if)#sw
Switch(config-if)#switchport mode trunk
Command rejected: An interface whose trunk encapsulation is "Auto" can not be co
nfigured to "trunk" mode.
Switch(config-if)#switchport trunk encapsulation dot1q
Switch(config-if)#switchport mode trunk
Switch(config-if)#
Switch(config-if)#
```

Step 3: Create DHCP allocation for individual Vlan's. Make sure to exclude the static address used by the Router as Gateway.

```
Router(config)#ip dhcp excluded-address 192.168.10.1
Router(config)#ip dhcp pool IP_Add_VLAN10
Router(dhcp-config)#network 192.168.10.0 255.255.255.0
Router(dhcp-config)#default-
Router(dhcp-config)#default-router 192.168.10.1
Router(dhcp-config)#exit
Router(config)#
```

Create another DHCP pool for vlan 20 nodes

```
Router(config)#ip dhcp excluded-address 192.168.20.1
Router(config)#ip dhcp pool IP_ADD_VLAN20
Router(dhcp-config)#network 192.168.20.0 255.255.255.0
Router(dhcp-config)#de
Router(dhcp-config)#default-router 192.168.20.1
Router(dhcp-config)#
Router(dhcp-config)#ex
```

Step 4: Create Sub interfaces of Router f0/0 interface to create Inter-Vlan routing and assign the default gateway address to each sub interface.

```
Router(config-if)#int f0/0.10
Router(config-subif)#enca
Router(config-subif)#encapsulation dot1q 10
Router(config-subif)#ip add
*Mar  1 00:56:34.371: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
% Incomplete command.

Router(config-subif)#ip add 192.168.10.1 255.255.255.0
Router(config-subif)#
Router(config-subif)#int f0/0.20
Router(config-subif)#en
Router(config-subif)#encapsulation dot1q 20
Router(config-subif)#ip add 192.168.20.1 255.255.255.0
Router(config-subif)#
```

Step 5: Begin Wireshark to capture packet's from PC1.

DORA Messages:

Discover: PC1 is not given an IP address, so once it requests for an IP, the message will be forwarded to DHCP server if it is local LAN. We use DHCP relay agents to communicate with DHCP server present in remote location.

- Source IP of PC is 0.0.0.0 and Destination: 255.255.255.255. PC is broadcasting frame to all available DHCP servers.

Offer: DHCP Offer message is sent from client to server.

- Source IP: Server IP and Destination IP: New IP address offered by server. Offer message can be unicast or broadcast message depending on what is assigned to Boot Protocol. Here PC1 is assigned in Vlan 10 broadcast domain, so Vlan will be registered under 192.168.10.0 range.
- Server offeres 192.168.10.2 IP address

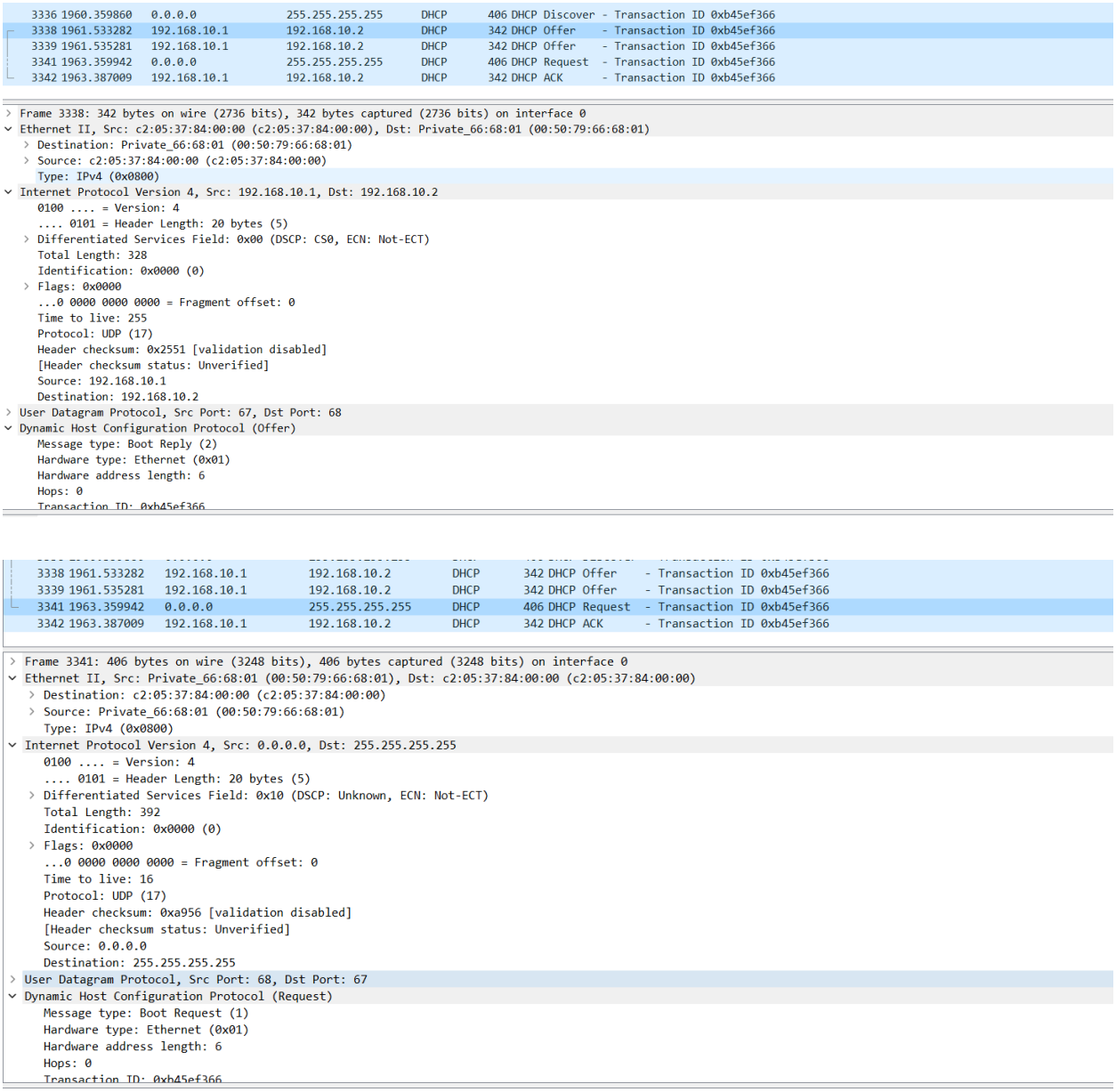
Request: Request frame is sent from client to server.

- Source IP: 0.0.0.0 (because the IP is not yet allocated) Destination IP: 255.255.255.255.

Ack: Ack message frame is sent from Server to Client.

- Source IP: 192.168.10.1 Destination IP: 192.168.10.2

The final step to assign the IP address. DHCP protocol has options like Lease time, Renewal Time and Subnet Mask.



```
3338 1961.533282   192.168.10.1      192.168.10.2      DHCP    342 DHCP Offer    - Transaction ID 0xb45ef366
3339 1961.535281   192.168.10.1      192.168.10.2      DHCP    342 DHCP Offer    - Transaction ID 0xb45ef366
3341 1963.359942   0.0.0.0           255.255.255.255   DHCP    406 DHCP Request  - Transaction ID 0xb45ef366
3342 1963.387009   192.168.10.1      192.168.10.2      DHCP    342 DHCP ACK      - Transaction ID 0xb45ef366
```

```
Dynamic Host Configuration Protocol (Request)
   Message type: Boot Request (1)
   Hardware type: Ethernet (0x01)
   Hardware address length: 6
   Hops: 0
   Transaction ID: 0xb45ef366
   Seconds elapsed: 0
 > Bootp flags: 0x0000 (Unicast)
   Client IP address: 192.168.10.2
   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: Private_66:68:01 (00:50:79:66:68:01)
   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: (54) DHCP Server Identifier (192.168.10.1)
 > Option: (50) Requested IP Address (192.168.10.2)
 > Option: (61) Client identifier
 > Option: (12) Host Name
 > Option: (55) Parameter Request List
 > Option: (255) End
   Padding: 000000000000000000000000000000000000000000000000…
```

```
3338 1961.533282   192.168.10.1      192.168.10.2      DHCP    342 DHCP Offer    - Transaction ID 0xb45ef366
3339 1961.535281   192.168.10.1      192.168.10.2      DHCP    342 DHCP Offer    - Transaction ID 0xb45ef366
3341 1963.359942   0.0.0.0           255.255.255.255   DHCP    406 DHCP Request  - Transaction ID 0xb45ef366
3342 1963.387009   192.168.10.1      192.168.10.2      DHCP    342 DHCP ACK      - Transaction ID 0xb45ef366
```

```
   Message type: Boot Reply (2)
   Hardware type: Ethernet (0x01)
   Hardware address length: 6
   Hops: 0
   Transaction ID: 0xb45ef366
   Seconds elapsed: 0
 > Bootp flags: 0x0000 (Unicast)
   Client IP address: 192.168.10.2
   Your (client) IP address: 192.168.10.2
   Next server IP address: 0.0.0.0
   Relay agent IP address: 0.0.0.0
   Client MAC address: Private_66:68:01 (00:50:79:66:68:01)
   Client hardware address padding: 00000000000000000000
   Server host name not given
   Boot file name not given
   Magic cookie: DHCP
 > Option: (53) DHCP Message Type (ACK)
 > Option: (54) DHCP Server Identifier (192.168.10.1)
 > Option: (51) IP Address Lease Time
 > Option: (58) Renewal Time Value
 > Option: (59) Rebinding Time Value
 > Option: (1) Subnet Mask (255.255.255.0)
 > Option: (3) Router
 > Option: (255) End
   Padding: 00000000000000000000000000000000000000000000
```