Name: Avis Oh Xin Wan **SIM ID:10255528** **UOW ID:8465678**

1.4 DHCP messages that I can see are DHCP Discover, DHCP Offer, DHCP Request and DHCP ACK.

a. The device uses the addresses **0.0.0.0** and **255.255.255.255** in the source and destination address fields, respectively. The **0.0.0.0** and **255.255.255.255** are two special addresses. Any device, whether it has a valid IP configuration or not, can use these addresses to send local broadcast messages. From these addresses, the **0.0.0.0** is used as the source address. If a device does not have the source address, it can use this address to send broadcast messages. **255.255.255.255** is the local broadcast address. Any message sent to this address is received by all hosts of the local network.

b. The DHCP Discovery message sent over UDP.

c. The source port number is 68, destination port number is 67.

Prove:

```
User Datagram Protocol, Src Port: 68, Dst Port: 67
    Source Port: 68
    Destination Port: 67
    Length: 308
    Checksum: 0xe97b [unverified]
    [Checksum Status: Unverified]
    [Stream index: 1]
  > [Timestamps]
    UDP payload (300 bytes)
> Dynamic Host Configuration Protocol (Discover)
```

d. The client's MAC address is Dell_4f:36:23 (00:08:74:4f:36:23)
The name of the client is Noho.
IP address the client is requesting is 192.168.1.101
The other parameters that the client request are Subnet Mask, Domian Name, Router, Domain Name Server, NetBIOS over TCP/IP Name Server, NetBIOS over TCP/IP Node Type, NetBIOS over TCP/IP Scope, Perform Router Discover, Static Route, Private/Classless Static Route (Microsoft), and Vendor-specific Information
Prove:

```
> Option: (12) Host Name
> Option: (60) Vendor class identifier
v Option: (55) Parameter Request List
    Length: 11
    Parameter Request List Item: (1) Subnet Mask
    Parameter Request List Item: (15) Domain Name
    Parameter Request List Item: (3) Router
    Parameter Request List Item: (6) Domain Name Server
    Parameter Request List Item: (44) NetBIOS over TCP/IP Name Server
    Parameter Request List Item: (46) NetBIOS over TCP/IP Node Type
    Parameter Request List Item: (47) NetBIOS over TCP/IP Scope
    Parameter Request List Item: (31) Perform Router Discover
    Parameter Request List Item: (33) Static Route
    Parameter Request List Item: (249) Private/Classless Static Route (Microsoft)
    Parameter Request List Item: (43) Vendor-Specific Information
> Option: (255) End
  Padding: 00000000000000000
```

4. The IP address of the DHCP server is 192.168.1.1
Prove:

```
Requested IP Address: 192.168.1.101
Option: (54) DHCP Server Identifier (192.168.1.1)
   Length: 4
   DHCP Server Identifier: 192.168.1.1
```

5. a.
   i. Subnet mask:255.255.255.0
   ii. Router:192.168.1.1
   iii. Domain Name Server Address: 63.240.76.19 and 204.127.198.19
   iv. IP address lease time: (86400s) 1 day
   v. IP address : 192.168.1.101

6. When DHCP sends configuration information to a client, the information is sent with a lease time. This is the length of time that the client can use the IP address it has been assigned.

During the lease time, the DHCP server cannot assign that IP address to any other clients. The purpose of a lease is to limit the length of time that a client can use an IP address. A lease prevents unused clients from taking up IP addresses when there are more clients than addresses. It also enables the administrator to make configuration changes to all of the clients on the network in a limited amount of time. When the lease expires, the client will request a new lease from DHCP. If the configuration data has changed, the new data will be sent to the client at that time.

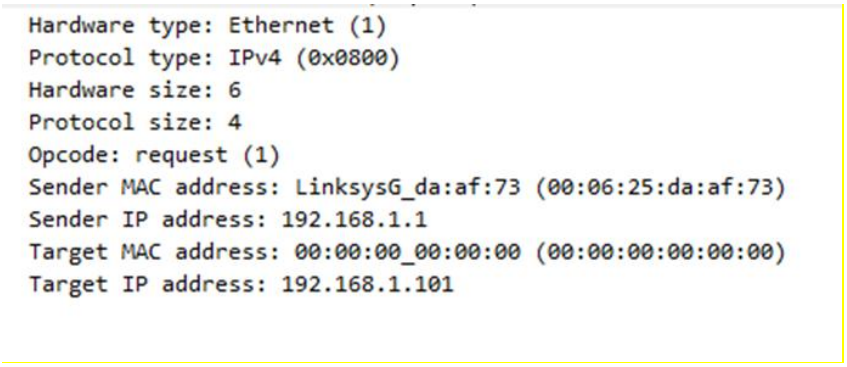
7. IP address did the client request is 192.168.1.101
Prove:

```
Option: (50) Requested IP Address (192.168.1.101)
   Length: 4
   Requested IP Address: 192.168.1.101
```

ARP

1. ARP is necessary because the software address (IP address) of the host or computer connected to the network needs to be translated to a hardware address (MAC address). Without ARP, a host would not be able to figure out the hardware address of another host. The LAN keeps a table or directory that maps IP addresses to MAC addresses of the different devices, including both endpoints and routers on that network.
   This table or directory is not maintained by users or even by IT administrators. Instead, the ARP protocol creates entries on the fly. If a user's device does not know the hardware address of the destination host, the device will send a message to every host on the network asking for this address. When the proper destination host learns of the request, it will reply back with its hardware address, which will then be stored in the ARP directory or table.
   If ARP is not supported, manual entries can be made to this directory.

2. Identify the MAC address corresponding to the IP address 192.168.1.101.



HTTP

HTTP (Hypertext Transfer Protocol) is the set of rules for transferring files -- such as text, images, sound, video and other multimedia files -- over the web. As soon as a user opens their web browser, they are indirectly using HTTP. HTTP is an application protocol that runs on top of the TCP/IP suite of protocols, which forms the foundation of the internet. When a client wants to access a website using its domain name, it first needs to obtain the IP address of the server hosting that website. This is where DNS comes into play.

When a client makes an HTTP request to a domain name (e.g., www.example.com), the client's device sends a DNS query to a DNS resolver (typically provided by the Internet Service Provider or configured by the user). The DNS resolver looks up the IP address associated with the domain name in the DNS hierarchy and returns it to the client.

Once the client receives the IP address, it can establish a connection with the server using that IP address and initiate the HTTP communication. The client sends the HTTP request to the server's IP address, and the server responds accordingly.

When you enter a URL in your web browser and hit Enter, the browser sends an HTTP request to the web server hosting the requested website. The server processes the request, generates an HTML page or other content, and sends it back to the browser as an HTTP response. HTTP operates over TCP (Transmission Control Protocol) on port 80.

DNS

The Domain Name System (DNS) is a crucial protocol that helps translate human-readable domain names (like www.example.com) into IP addresses (such as 192.168.1.1) that computers use to identify one another on the internet.

Recursive DNS Server: If the IP address isn't found in the local cache, your computer sends a request to a recursive DNS server provided by your Internet Service Provider (ISP). The recursive DNS server may have the answer in its cache, in which case it responds with the IP address. If not, it will initiate a DNS query on your behalf.

Authoritative DNS Server: The recursive DNS server contacts the authoritative DNS server responsible for the domain (e.g., example.com). This server holds the definitive record for the IP address associated with the requested domain.

Connection to the Website: With the IP address in hand, your computer establishes a connection to the web server associated with the domain name, allowing you to access the desired website.

In summary, DNS plays a crucial role in HTTP communication by resolving domain names to IP addresses, enabling clients to connect to the appropriate servers and access the requested resources. Without DNS, clients would have to rely on IP addresses directly, which would be inco inconvenient and less user-friendly.

HTTP

1. The client is gaia.cs.umass.edu and server's IP address is 128.119.245.12
2. The client connected using TCP.
3. The source port is 4127.
   The destination port is 80.
   Prove:

```
Transmission Control Protocol, Src Port: 4127, Dst Port: 80,
   Source Port: 4127
   Destination Port: 80
   [Stream index: 0]
   [Conversation completeness: Incomplete, DATA (15)]
   [TCP Segment Len: 501]
   Sequence Number: 1    (relative sequence number)
   Sequence Number (raw): 4113720498
   [Next Sequence Number: 502    (relative sequence number)]
   Acknowledgment Number: 1    (relative ack number)
```

4.

a. The client is trying to get /ethereal-labs/lab2-1.html

```
Hypertext Transfer Protocol
 GET /ethereal-labs/lab2-1.html HTTP/1.1\r\n
   [Expert Info (Chat/Sequence): GET /ethereal-labs/lab2-1.html HTTP/1.1\r\n]
    Request Method: GET
    Request URI: /ethereal-labs/lab2-1.html
    Request Version: HTTP/1.1
  Host: gaia.cs.umass.edu\r\n
  User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.0.2) Gecko/20021120 Netscape/7.01\r\n
```

b. The version of HTTP is the client running is HTTP/1.1

c. The client is using Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.0.2) Gecko/20021120 Netscape /7.01\r\n

d. (i)Host:
Purpose: The "Host" field specifies the domain name or IP address of the target server to which the HTTP request is being sent. It's used when a single web server hosts multiple websites on the same IP address. The "Host" field helps the server route the request to the correct website.

(ii) Accept:
Purpose: The "Accept" field specifies the types of media or data formats that the client (usually a web browser) can handle or prefers. It informs the server about the types of responses (e.g., HTML, JSON, XML) that the client can accept.

(iii) Keep-Alive:
Purpose: The "Keep-Alive" field, often referred to as the "Connection: keep-alive" header, is used to maintain a persistent connection between the client and the server. This allows multiple HTTP requests and responses to be sent over the same connection, reducing the overhead of establishing a new connection for each request, thus improving performance.

(iv) Connection:
Purpose: The "Connection" field is used to control the behaviour of the connection between the client and server. It can specify whether the connection should be kept alive for future requests or closed after a single request-response cycle. Common values include "keep-alive" (to maintain the connection) and "close" (to close the connection after a single request).

6. a.
The version of the web server is Apache/2.0.40 (Ret Hat Linux)\r\n
Prove:

```
Hypertext Transfer Protocol
  HTTP/1.1 200 OK\r\n
    [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
    Response Version: HTTP/1.1
    Status Code: 200
    [Status Code Description: OK]
    Response Phrase: OK
  Date: Tue, 23 Sep 2003 05:29:50 GMT\r\n
  Server: Apache/2.0.40 (Red Hat Linux)\r\n
```

b. The server is running on a Linux machine.

DNS

1. Source IP address is 128.238.38.160
Destination IP address is 128.238.29.23

```
Internet Protocol Version 4, Src: 128.238.38.160, Dst: 128.238.29.23
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 58
  Identification: 0x229e (8862)
  000. .... = Flags: 0x0
  ...0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 128
  Protocol: UDP (17)
  Header Checksum: 0xd281 [validation disabled]
```

2. DNS messages are sent over UDP.

3. Source port number is 3163.
Destination port number is 53.

```
User Datagram Protocol, Src Port: 3163, Dst Port: 53
  Source Port: 3163
  Destination Port: 53
  Length: 38
  Checksum: 0x8acb [unverified]
  [Checksum Status: Unverified]
  [Stream index: 1]
  [Timestamps]
  UDP payload (30 bytes)
```

4. a.
The client is trying to figure out what IP address the webserver that hosts www.ietf.org has. As a result, it looks up the DNS for that IP address.

```
Domain Name System (response)
   Transaction ID: 0x006e
 ▾ Flags: 0x8180 Standard query response, No error
      1... .... .... .... = Response: Message is a response
      .000 0... .... .... = Opcode: Standard query (0)
      .... .0.. .... .... = Authoritative: Server is not an authority for domain
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... 1... .... = Recursion available: Server can do recursive queries
      .... .... .0.. .... = Z: reserved (0)
      .... .... ..0. .... = Answer authenticated: Answer/authority portion was not authenticated by the server
      .... .... ...0 .... = Non-authenticated data: Unacceptable
      .... .... .... 0000 = Reply code: No error (0)
   Questions: 1
   Answer RRs: 2
   Authority RRs: 0
   Additional RRs: 0
 ▾ Queries
   ▸ www.ietf.org: type A, class IN
 ▾ Answers
   ▸ www.ietf.org: type A, class IN, addr 132.151.6.75
   ▸ www.ietf.org: type A, class IN, addr 65.246.255.51
   [Request In: 8]
   [Time: 0.000844000 seconds]
```

5. a.

The server give the client 2 answers. They are www.ietf.org: type A, class IN, addr 132.151.6.75 and www.ietf.org: type A, class IN, addr 65.246.255.51.

```
  Questions: 1
  Answer RRs: 2
  Authority RRs: 0
  Additional RRs: 0
▾ Queries
  ▸ www.ietf.org: type A, class IN
▾ Answers
  ▸ www.ietf.org: type A, class IN, addr 132.151.6.75
  ▸ www.ietf.org: type A, class IN, addr 65.246.255.51
```

b. The client is allowed to keep the answer for 1678 (27 minutes, 58 seconds)

Prove:

```
▾ Answers
  ▾ www.ietf.org: type A, class IN, addr 132.151.6.75
      Name: www.ietf.org
      Type: A (Host Address) (1)
      Class: IN (0x0001)
      Time to live: 1678 (27 minutes, 58 seconds)
      Data length: 4
      Address: 132.151.6.75
  ▾ www.ietf.org: type A, class IN, addr 65.246.255.51
      Name: www.ietf.org
      Type: A (Host Address) (1)
      Class: IN (0x0001)
      Time to live: 1678 (27 minutes, 58 seconds)
      Data length: 4
      Address: 65.246.255.51
```