SEHH2238

Computer Networking

Group 13

Assignment 2

Member:

Tsoi Yiu Chik 20195601A(Leader)

Lau Siu Paak 20001245A

Lee Wai Leuk 20031030A

Mak Kin Ho 20042880A

Sin Pui Lam 20036097A

Cheung Man Hei 20093215A

**Part A**

Q1

1. N=1+5+7+5+0+0 = 18 mod 10 +17

=25

Therefore, the address is 10.100.111.216/25

The binary address is 00001010.01100100.01101111.11011000/25

The mask in dotted binary notation is 11111111.11111111.11111111.10000000

The mask in dotted decimal notation is 255.255.255.128

1. The starting range is 10.100.111.128

The end range is 10.100.111.255

which have a total of 128 addresses

1. The network address(binary) is 00001010.01100100.01101111.10000000

The network address(decimal) is 10.100.111.128

1. The broadcast address(binary) is 00001010.01100100.01101111.11111111

The broadcast address(decimal) is 10.100.111.255

Q2

1. 1st first address: 10101100 00011111 00000000 00000000/22

|  |  |  |
| --- | --- | --- |
|  | First address | Last address |
| 1st company | 172.31.0.0/22 | 172.31.3.255/22 |
| 2nd company | 172.31.4.0/22 | 172.31.7.255/22 |
| 3rd company | 172.31.8.0/22 | 172.31.11.255/22 |
| 4th company | 172.31.12.0/22 | 172.31.15.255/22 |

1. 1st first address: 10101100 00011111 00010000 00000000/25

|  |  |  |
| --- | --- | --- |
|  | First address | Last address |
| 1st company | 172.31.16.0/25 | 172.31.16.127/25 |
| 2nd company | 172.31.16.128/25 | 172.31.16.255/25 |
| 3rd company | 172.31.17.0/25 | 172.31.17.127/25 |
| 4th company | 172.31.17.128/25 | 172.31.17.255/25 |
| 4 last company | 172.31.30.0/25 | 172.31.30.127/25 |
| 3 last company | 172.31.30.128/25 | 172.31.30.255/25 |
| 2 last company | 172.31.31.0/25 | 172.31.31.127/25 |
| Last company | 172.31.31.128/25 | 172.31.31.255/25 |

1. 1st first address: 10101100 00011111 00100000 00000000/27

|  |  |  |
| --- | --- | --- |
|  | First address | Last address |
| 1st company | 172.31.32.0/27 | 172.31.32.31/27 |
| 2nd company | 172.31.32.32/27 | 172.31.32.63/27 |
| 3rd company | 172.31.32.64/27 | 172.31.32.95/27 |
| 4th company | 172.31.32.96/27 | 172.31.32.127/27 |
| 4 last company | 172.31.47.128/27 | 172.31.47.159/27 |
| 3 last company | 172.31.47.160/27 | 172.31.47.191/27 |
| 2 last company | 172.31.47.192/27 | 172.31.47.223/27 |
| Last company | 172.31.47.224/27 | 172.31.47.255/27 |

**Part B**

Q1

|  |  |
| --- | --- |
| Physical Address | 2C-F0-5D-38-0C-6B |
| IPv4 Address | 192.168.0.159 |
| Default Gateway | 192.168.0.1 |
| DNS Server | 1.1.1.1 |

Q2

ip.src == 192.168.0.159 and tcp

|  |  |
| --- | --- |
|  | **Information(in hex)** |
| Destination Ethernet Address | b0:4e: 26:1c:f9:44 |
| Source Ethernet Address | 2c:f0:5d:38:0c:6b |
| Type | 0x0800 |

1. 192.168.0.1
2. No
3. Router

Q3

|  |  |  |
| --- | --- | --- |
| **Header Field** | **Value (hex)** | **Meaning** |
| Version | 4 | Version:4 |
| Header Length | 5 | 20 bytes |
| Service type | 00 | DSCP: CS0, ECN: NOT-ECT |
| Total Length | 00 7e | 126 |
| Identification | 34 1b | 13339 |
| Flags | 40 | Don’t fragment |
| Fragment Offset | 00 | 0 |
| Time to Live | 80 | 128 |
| Protocol | 06 | TCP |
| Header checksum | 00 00 | validation disabled |
| Source IP address | C0 a8 00 9f | 192.168.0.159 |
| Destination address | A2 9f 81 eb | 162.159.129.235 |

Q4

Used filter: ip.addr == 14.136.239.59 and tcp.port == 443 and tcp.port == 54016

1. 14.126.239.59
2. Server: 443 client: 54016

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Source IP address** | **Destination IP**  **address** | **Source port no.** | **Destination port no.** | **Flags**  **(0x)** | **What happened** |
| 1903 | 192.168.0.159 | 14.136.239.59 | 54016 | 443 | 40 | SYN |
| 1921 | 14.136.239.59 | 192.168.0.159 | 443 | 54016 | 40 | SYN, ACK |
| 1922 | 192.168.0.159 | 14.136.239.59 | 54016 | 443 | 40 | ACK |

|  |  |
| --- | --- |
|  | **Length in bytes** |
| Total length | 1500 bytes |
| Ethernet header length | 14 bytes |
| IP header length | 20 bytes |
| TCP header length | 20 bytes |
| TCP Segment length | 1460 bytes |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Source IP address** | **Destination IP**  **address** | **Source port no.** | **Destination port no.** | **Flags**  **(0x)** | **What happened** |
| 5777 | 192.168.0.159 | 14.136.239.59 | 54016 | 443 | 011 | FIN, ACK |
| 5842 | 14.136.239.59 | 192.168.0.159 | 443 | 54016 | 011 | FIN, ACK |
| 5845 | 192.168.0.159 | 14.136.239.59 | 54016 | 443 | 014 | RST, ACK |

1. TCP Keep-Alive packet is to check two links between the two link is connecting and to prevent the link disconnected. After the server issues a FIN request, the client sends TCP Keep-Alive packets to ensure that the server still running and does not disconnect.
2. Selected frame 2126

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Frame no.** | **payload** | **Payload size(bytes)** | **Sequence no.** | **Acknowledge number** |
| 2121 | 407-1866 | 1460 | 32240 | 3202 |
| 2122 | 1867-3326 | 1460 | 33700 | 3202 |
| 2123 | 3327-4786 | 1460 | 35160 | 3202 |
| 2124 | 4787-6246 | 1460 | 36620 | 3202 |
| 2125 | 6247-7706 | 1460 | 38080 | 3202 |

1. Frame no.: 2126

TCP segment length: 8221

Graphical user interface, text, application, email

Description automatically generated

Q5

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Frame no.** | **Source IP address** | **Destination IP address** | **Transport layer protocol** | **Source port number** | **Destination port number** | **Domain name/ IP address** |
| 24 | 192.168.0.159 | 1.1.1.1 | UDP | 58666 | 53 | clients2.google |
| 33 | 1.1.1.1 | 192.168.0.159 | UDP | 53 | 58666 | clients2.google |
| 68 | 192.168.0.159 | 1.1.1.1 | UDP | 63790 | 53 | mtalk.google |
| 70 | 1.1.1.1 | 192.168.0.159 | UDP | 53 | 63790 | mtalk.google |
| 106 | 192.168.0.159 | 1.1.1.1 | UDP | 52690 | 53 | update.googleapis |
| 114 | 1.1.1.1 | 192.168.0.159 | UDP | 53 | 52690 | update.googleapis |

Q6

Text

Description automatically generated(A)

Graphical user interface, text, application, email

Description automatically generated(B)Used filter: dns or icmp

(C)

(D)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Frame no** | **Source IP address** | **Destination IP address** | **Protocol** | **Source port number** | **Destination port number** | **Domain name/ IP address** |
| 557 | 192.168.0.159 | 192.168.0.1 | DNS | 49628 | 53 | microsoft |
| 558 | 192.168.0.1 | 192.168.0.159 | DNS | 53 | 49628 | microsoft |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Frame no.** | **Time** | **Source IP address** | **Destination IP address** | **Protocol** | **Time to Live** | **Type** | **Response Time** | **Sequence Number (BE)** | **Sequence Number (LE)** |
| 560 | 2.691 | 192.168.0.159 | 184.51.241.179 | ICMP | 128 | 8 | n.a. | 508 | 64513 |
| 561 | 2.696 | 184.51.241.179 | 192.168.0.159 | ICMP | 59 | 0 | 5.738ms | 508 | 64513 |
| 756 | 3.693 | 192.168.0.159 | 184.51.241.179 | ICMP | 128 | 8 | n.a. | 509 | 64769 |
| 758 | 3.698 | 184.51.241.179 | 192.168.0.159 | ICMP | 59 | 0 | 5.781ms | 509 | 64769 |
| 1070 | 4.695 | 192.168.0.159 | 184.51.241.179 | ICMP | 59 | 8 | n.a. | 510 | 65025 |
| 1072 | 4.700 | 184.51.241.179 | 192.168.0.159 | ICMP | 128 | 0 | 5.792ms | 510 | 65025 |
| 1276 | 5.697 | 192.168.0.159 | 184.51.241.179 | ICMP | 59 | 8 | n.a. | 511 | 65281 |
| 1278 | 5.702 | 184.51.241.179 | 192.168.0.159 | ICMP | 128 | 0 | 5.508ms | 511 | 65281 |

Graphical user interface, text, application, email

Description automatically generatedE) We can find timestamp in the packet headers. While packets are being captured, each packet is time stamped as it comes in. While capturing packets, Wireshark gets the time stamps from the Npcap library and the Npcap library in turn gets them from the operating system kernel. Finally, Wireshark gets the response time. capture data is loaded from a capture file, Wireshark obviously gets the data from that file.

F)

32 bytes

Hex:

61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69

ASCII:

Graphical user interface, text, application

Description automatically generateda b c d e f g h I j k l m n o p q r s t u v w a b c d e f g h i