

CS315 : Computer Networks Lab

Assignment 6

Sourabh Bhosale (200010004)

February 7, 2023

1 Part 1: Wireshark UDP

```
~ $ nslookup www.nyu.edu
Server:      10.250.200.3
Address:     10.250.200.3#53

Non-authoritative answer:
www.nyu.edu canonical name = d1q5ku5vnwkd2k.cloudfront.net.
Name:   d1q5ku5vnwkd2k.cloudfront.net
Address: 108.159.28.104
Name:   d1q5ku5vnwkd2k.cloudfront.net
Address: 108.159.28.89
Name:   d1q5ku5vnwkd2k.cloudfront.net
Address: 108.159.28.92
Name:   d1q5ku5vnwkd2k.cloudfront.net
Address: 108.159.28.39

~ $
```

1. Select the first UDP segment in your trace. What is the packet number of this segment in the trace file? What type of application-layer protocol message is being carried in this UDP segment? Look at the details of this packet in Wireshark. How many fields are there in the UDP header? What are the names of these fields?

Packet number : 1

Application-layer protocol message : DNS UDP header contains 6 fields: Source port, Destination port, Length, Checksum, Timestamps, UDP payload

2. By consulting the displayed information in Wireshark's packet content field for this packet, what is the length (in bytes) of each of the UDP header fields?

The UDP header has a fixed length of 8 bytes. Each of these 4 header fields is 2 bytes long

The image shows a Wireshark interface with a packet list on the left and a packet details pane on the right. The packet list shows various protocols including MDNS, DHCP, and NBNs. The selected packet is a User Datagram Protocol (UDP) packet with source port 54915 and destination port 54915. The packet details pane shows the following information:

- Source Port: 54915
- Destination Port: 54915
- Length: 271
- Checksum: 0x2a77 [unverified]
- [Checksum Status: Unverified]
- [Stream index: 0]
- [Timestamps]
- [Time since first frame: 0.000000000 seconds]
- [Time since previous frame: 0.000000000 seconds]
- UDP payload (263 bytes)
- Data (263 bytes)

```

0 AAAA https.local, "QM" question
0 A https.local, "QM" question
se 0x0000 TXT, cache flush PTR _mi-connect._udp.local PTR {"nm":"siva","as":
3 PTR _37f83649._sub._googlecast._tcp.local, "QM" question PTR _googlecast._tcp
se 0x0000 TXT, cache flush PTR _mi-connect._udp.local PTR {"nm":"siva","as":
se 0x0000 PTR, cache flush Android-37.local PTR, cache flush Android-37.local
se 0x0000 PTR, cache flush Android-37.local PTR, cache flush Android-37.local

```

```

0010 01 23 fa 76 00 00 80 11 1f dc 0a c4 09 f0 0a c4 # v . . . . .
0020 ff ff d6 83 d6 83 01 0f 2a 77 00 50 72 44 a4 00 . . . . . *w PrDJ .
0030 00 00 00 bb 03 e3 0c 02 00 00 e0 b5 4f aa 4b 00 . . . . . 0 K .
0040 00 00 00 00 00 00 00 00 00 00 33 27 00 00 00 00 . . . . . 3 . . . .
0050 00 00 f0 ba 03 e3 0c 02 00 00 40 96 e1 e3 0c 02 . . . . . @ . . . .
0060 00 00 00 41 e2 e3 0c 02 00 00 20 00 00 00 00 00 . . . . . A . . . .
0070 00 00 7c 6a 54 54 00 00 00 00 20 a4 22 55 00 00 . jTT . . . "U .
0080 00 00 19 ba 4f aa 4b 00 00 00 00 00 00 00 00 00 . . . . . 0 K . . . .
0090 00 00 80 96 e1 e3 0c 02 00 00 64 b6 4f aa 4b 00 . . . . . d 0 K .
00a0 00 00 80 b6 4f aa 4b 00 00 00 68 33 3b 7b 37 62 . . . . . 0 K . h3 ; 7b
00b0 30 32 32 39 32 65 2d 33 30 63 64 2d 34 31 33 34 02292e-3 0cd-4134
00c0 2d 38 33 35 38 2d 38 65 37 36 38 61 64 63 34 66 -8358-8e 768adc4f
00d0 35 39 7d 00 00 00 00 00 00 00 01 00 00 00 00 00 59} . . . . .

```

3. The value in the Length field is the length of what? Verify your claim with your captured UDP packet.

The length field specifies the number of bytes in the UDP segment (header plus data). An explicit length value is needed since the size of the data field may differ from one UDP segment to the next. The length of UDP payload for selected packet is 263 bytes. 271 bytes - 8 bytes = 263 bytes.

4. What is the maximum number of bytes that can be included in a UDP payload? (Hint: the answer to this question can be determined by your answer to 2. above)

The maximum number of bytes that can be included in a UDP payload is $(2^{16} - 1)$ bytes plus the header bytes. This gives 65535 bytes – 8 bytes = 65527 bytes.

5. What is the largest possible source port number? (Hint: see the hint in 4.)

The largest possible source port number is $(2^{16} - 1) = 65535$.

6. What is the protocol number for UDP? Give your answer in decimal notation. To answer this question, you'll need to look into the Protocol field of the IP datagram containing this UDP segment.

The IP protocol number for UDP is 0x11 hex, which is 17 in decimal value.

```
31 0.418000 10.196.0.111 224.0.0.251 MDNS 71 Standard query 0x0000 AAAA https.local, "QM" question
32 0.418088 10.196.0.111 224.0.0.251 MDNS 91 Standard query 0x0000 A https.local, "QM" question
33 0.418589 fe80::61df:6756:a9... ff02::fb MDNS 412 Standard query response 0x0000 TXT, cache flush PTR_mi-connect_udp.local PTR ("nm":"siva","as":")
34 0.412841 fe80::343c:5dff:fe... ff02::fb MDNS 183 Standard query 0x0003 PTR_37f83649_sub_googlecast_tcp.local, "QM" question PTR_googlecast_tcp
35 0.414544 10.196.4.94 224.0.0.251 MDNS 392 Standard query response 0x0000 TXT, cache flush PTR_mi-connect_udp.local PTR ("nm":"siva","as":")
36 0.417163 10.196.4.94 224.0.0.251 MDNS 312 Standard query response 0x0000 PTR, cache flush Android-37.local PTR, cache flush Android-37.local
39 0.514491 fe80::6a66:433a:a9... ff02::fb MDNS 292 Standard query response 0x0000 PTR, cache flush Android-37.local PTR, cache flush Android-37.local
40 0.514924 10.196.12.232 224.0.0.251 MDNS

> Ethernet II, Src: LiteonTc_1c:15:83 (88:30:49:1c:15:83), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 10.196.9.240, Dst: 10.196.255.255
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 291
    Identification: 0xfa76 (64118)
  > 0000 .... = Flags: 0x0
  ...0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 128
  Protocol: UDP (17)
  Header Checksum: 0x1fdc [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 10.196.9.240
  Destination Address: 10.196.255.255

0010 01 23 fa 76 00 00 00 11 1f dc 0a c4 09 f0 0a c4 # v .....
0020 ff ff d6 83 06 83 01 0f 2a 77 00 50 72 44 4a 00 ..... wv PrDJ
0030 00 00 00 00 03 c3 0c 02 00 00 e0 5f aa 4b 00 ..... 0 K
0040 00 00 00 00 00 00 00 00 00 00 33 27 00 00 00 00 ..... 3'....
0050 00 00 f0 ba 03 e3 0c 02 00 00 40 96 e1 e3 0c 02 ..... @.....
0060 00 00 00 41 e2 e3 0c 02 00 00 00 00 00 00 00 00 ..... A.....
0070 00 00 7c 6a 54 54 00 00 00 00 20 e4 22 55 00 00 ..... [TT..... "U...
0080 00 00 19 ba 4f aa 4b 00 00 00 00 00 00 00 00 00 ..... 0 K.....
0090 00 00 80 96 e1 e3 0c 02 00 00 64 b6 4f aa 4b 00 ..... d 0 K.....
00a0 00 00 80 b6 4f aa 4b 00 00 68 33 3b 7b 37 62 ..... 0 K..... h3;7b
00b0 30 32 32 39 32 65 2d 33 30 63 64 2d 34 31 33 34 02292e-3 bcd-4134
00c0 2d 38 33 35 38 2d 38 65 37 36 38 61 64 63 34 66 -8358-8e 768adc4f
00d0 35 39 7d 00 00 00 00 00 00 00 01 00 00 00 00 00 59).....
00e0 00 00 60 b6 4f aa 4b 00 00 00 00 00 00 00 00 00 ..... 0 K.....
00f0 00 00 00 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 00 00 00 .....
0110 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0120 00 00 00 00 00 00 00 00 00 00 00 00 00 c5 25 b3 ..... %
```

7. Examine the pair of UDP packets in which your host sends the first UDP packet and the second UDP packet is a reply to this first UDP packet. (Hint: for a second packet to be sent in response to a first packet, the sender of the first packet should be the destination of the second packet). What is the packet number of the first of these two UDP segments in the trace file? What is the packet number of the second of these two UDP segments in the trace file? Describe the relationship between the port numbers in the two packets.

Packet number of the first of these two UDP segments : 415

Packet number of the second of these two UDP segments : 594

The source port of the UDP packet sent by the host is the same as the destination port of the reply packet, and conversely the destination port of the UDP packet sent by the host is the same as the source port of the reply packet.

```
> Frame 415: 71 bytes on wire (568 bits), 71 bytes captured (568 bits) on interface en0, id 0
> Ethernet II, Src: Apple_ad:cf:45 (14:7d:da:ad:cf:45), Dst: ExtremeNetworks_9a:82:e8 (02:04:96:9a:82:e8)
> Internet Protocol Version 4, Src: 10.196.7.179, Dst: 10.250.200.3
√ User Datagram Protocol, Src Port: 55670, Dst Port: 53
    Source Port: 55670
    Destination Port: 53
    Length: 37
    Checksum: 0x44f8 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 49]
    √ [Timestamps]
        [Time since first frame: 0.000000000 seconds]
        [Time since previous frame: 0.000000000 seconds]
    UDP payload (29 bytes)
```

```
> Frame 594: 178 bytes on wire (1424 bits), 178 bytes captured (1424 bits) on interface en0, id 0
> Ethernet II, Src: ExtremeNetworks_9a:82:e8 (02:04:96:9a:82:e8), Dst: Apple_ad:cf:45 (14:7d:da:ad:cf:45)
> Internet Protocol Version 4, Src: 10.250.200.3, Dst: 10.196.7.179
√ User Datagram Protocol, Src Port: 53, Dst Port: 55670
    Source Port: 53
    Destination Port: 55670
    Length: 144
    Checksum: 0xca12 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 49]
    √ [Timestamps]
        [Time since first frame: 0.864810000 seconds]
        [Time since previous frame: 0.864810000 seconds]
    UDP payload (136 bytes)
```