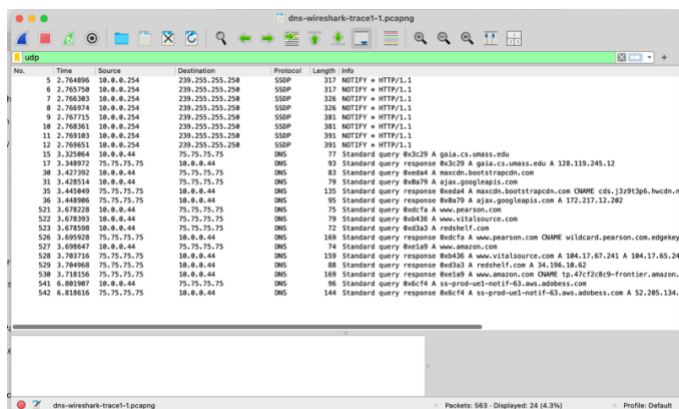


Catalina Cisneros

## Lab: Wireshark UDP v9.0

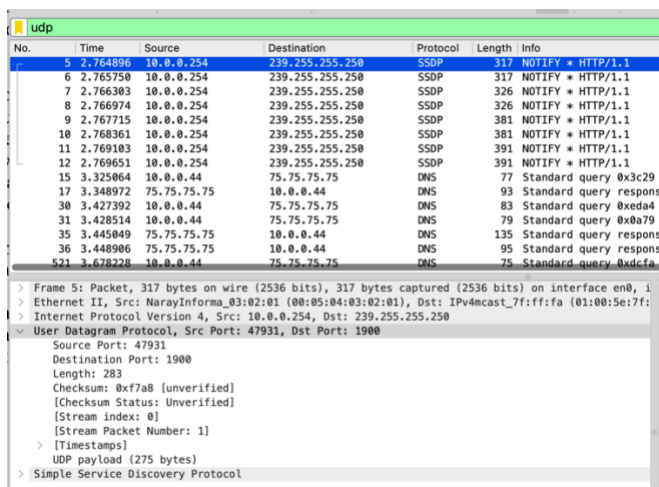
Downloaded the packet files from <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces-9e.zip>, since I couldn't do a real time packet capture.



## Questions:

1. Select the first UDP segment in your trace. What is the packet number4 of this segment in the trace file? What type of application-layer payload or protocol message is being carried in this UDP segment? Look at the details of this packet in Wireshark. How many fields there are in the UDP header? (You shouldn't look in the textbook! Answer these questions directly from what you observe in the packet trace.) What are the names of these fields?

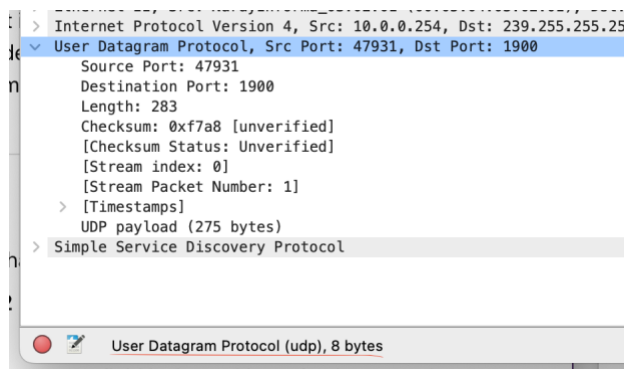
→ The first UDP segment is packet 5, which carries an SSDP (HTTP/1.1) message. The UDP header includes four fields: Source Port (1900), Destination Port (1900), Length (317), and Checksum



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

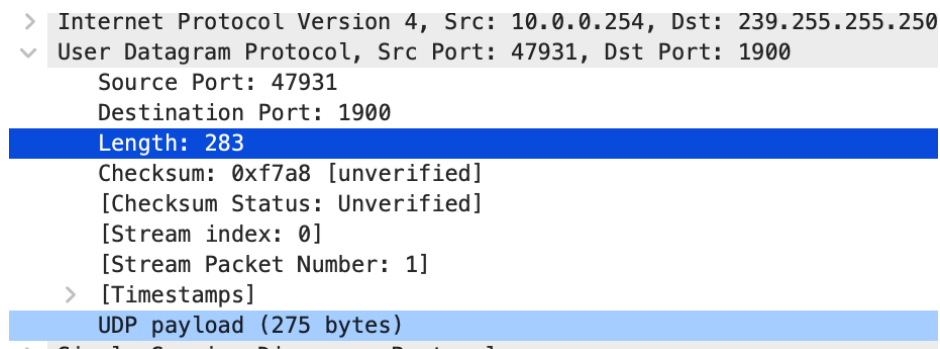
→ Each field (Source Port, Destination Port, Length, and Checksum) is 2 bytes. The total header length is 8 bytes

Destination Port (udp.dstport), 2 bytes



3. The value in the Length field is the length of what? (You can consult the text for this answer). Verify your claim with your captured UDP packet.

→ The UDP Length field shows 283 bytes, meaning the total segment size includes 8 bytes of header and 275 bytes of SSDP data



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)

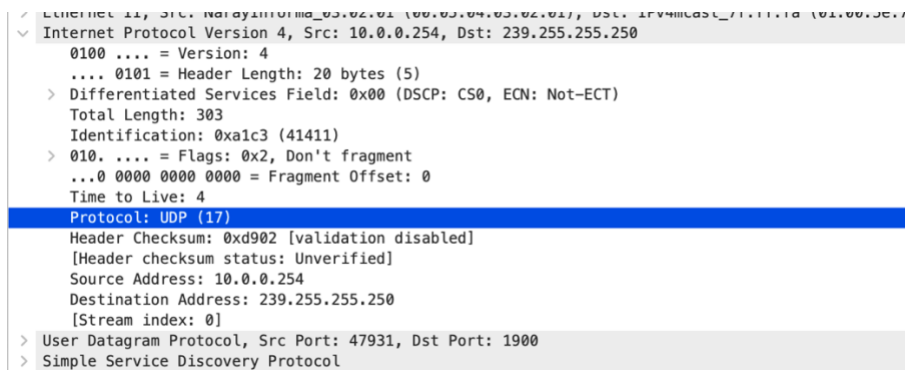
→ Since the UDP Length field is 16 bits, it can represent up to 65,535 bytes in total. Subtracting the 8-byte UDP header leaves 65,527 bytes for the payload

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

→ The Source Port field is 16 bits, so the largest possible value is 65,535

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 (see Figure 4.13 in the text, and the discussion of IP header fields).

→ The protocol number for UDP is 17



```
Internet Protocol Version 4, Src: 10.0.0.254, Dst: 239.255.255.250
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 303
    Identification: 0xalc3 (41411)
  > 010. .... = Flags: 0x2, Don't fragment
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 4
  Protocol: UDP (17)
  Header Checksum: 0xd902 [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 10.0.0.254
  Destination Address: 239.255.255.250
  [Stream index: 0]
  > User Datagram Protocol, Src Port: 47931, Dst Port: 1900
  > Simple Service Discovery Protocol
```

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<sup>5</sup> of the first of these two UDP segments in the trace file? What is the value in the source port field in this UDP segment? What is the value in the destination port field in this UDP segment? What is the packet number<sup>6</sup> of the second of these two UDP segments in the trace file? What is the value in the source port field in this second UDP segment? What is the value in the destination port field in this second UDP segment? Describe the relationship between the port numbers in the two packets.

→ First UDP segment is packet 30.

- Source port = 53905
- Destination port = 53

30	3.427392	10.0.0.44	75.75.75.75	DNS	
31	3.428514	10.0.0.44	75.75.75.75	DNS	
35	3.445049	75.75.75.75	10.0.0.44	DNS	1
36	3.448906	75.75.75.75	10.0.0.44	DNS	
521	3.678228	10.0.0.44	75.75.75.75	DNS	
522	3.678393	10.0.0.44	75.75.75.75	DNS	
523	3.678598	10.0.0.44	75.75.75.75	DNS	

> Frame 30: Packet, 83 bytes on wire (664 bits), 83 bytes captured (664 bit

> Ethernet II, Src: Apple\_98:d9:27 (78:4f:43:98:d9:27), Dst: Maxlinear\_80:0

> Internet Protocol Version 4, Src: 10.0.0.44, Dst: 75.75.75.75

> User Datagram Protocol, Src Port: 53905, Dst Port: 53

Source Port: 53905

Destination Port: 53

Length: 49

→ Second UDP segment is packet 35.

- Source port = 53
- Destination port = 53905

31	3.428514	10.0.0.44	75.75.75.75	DNS	1
35	3.445049	75.75.75.75	10.0.0.44	DNS	13
36	3.448906	75.75.75.75	10.0.0.44	DNS	9
521	3.678228	10.0.0.44	75.75.75.75	DNS	7
522	3.678393	10.0.0.44	75.75.75.75	DNS	7
523	3.678598	10.0.0.44	75.75.75.75	DNS	7

> Frame 35: Packet, 135 bytes on wire (1080 bits), 135 bytes captured (1080

> Ethernet II, Src: Maxlinear\_80:00:00 (00:50:f1:80:00:00), Dst: Apple\_98:d

> Internet Protocol Version 4, Src: 75.75.75.75, Dst: 10.0.0.44

> User Datagram Protocol, Src Port: 53, Dst Port: 53905

Source Port: 53

Destination Port: 53905

Length: 101

Checksum: 0xad07 [unverified]

→ The reply swaps the ports: server uses 53, client uses the same ephemeral port as in the request.