# CS 6043 Computer Networking

# PROJECT 2

By Mitra Sasanka Darbha, M#13523468

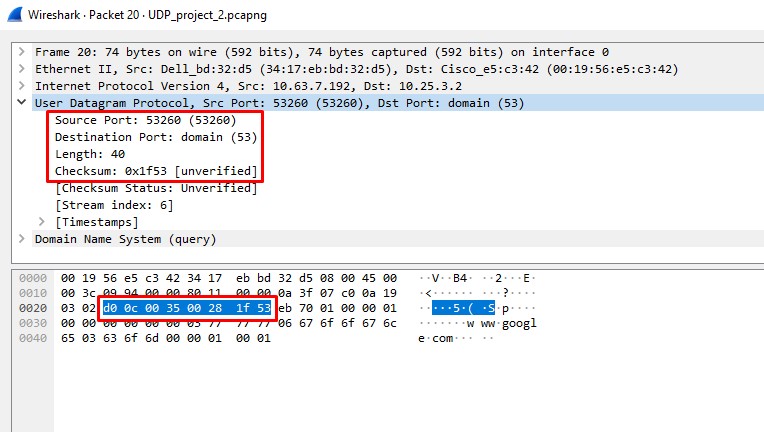
# Part 1: UDP

## Question 1:

UDP header contains 4 fields:

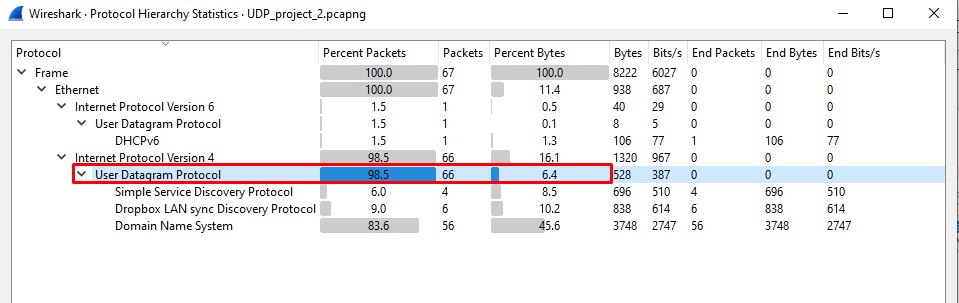
1. Source port
2. Destination port
3. Length
4. Checksum

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



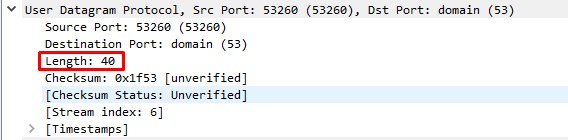
## Question 2:

The percentage of IPv4 UDP packets is 98.5%



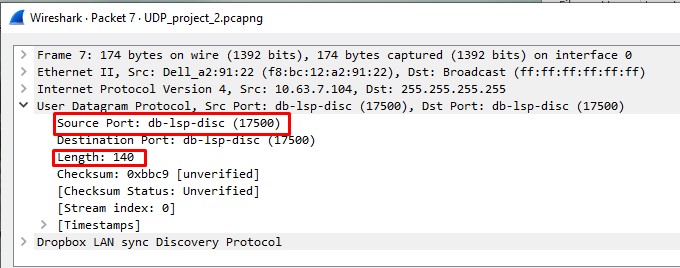
## Question 3:

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 if UDP payload for selected packet is 32 bytes. (40 bytes – 8bytes = 32 bytes)



## Question 4:

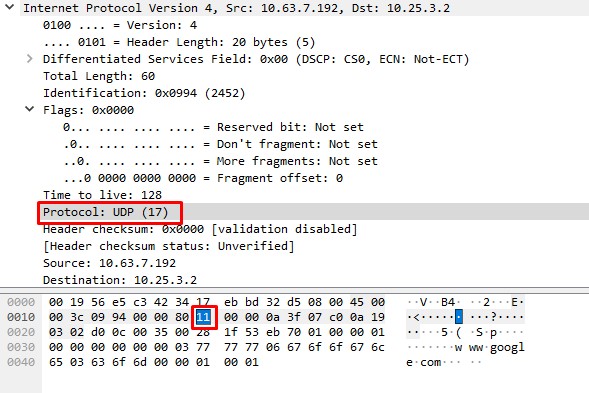
The source port is 17500 and length is 140 bytes. This is a Dropbox LAN sync discovery protocol packet.



The largest possible source port number is 216-1 = 65535

## Question 5:

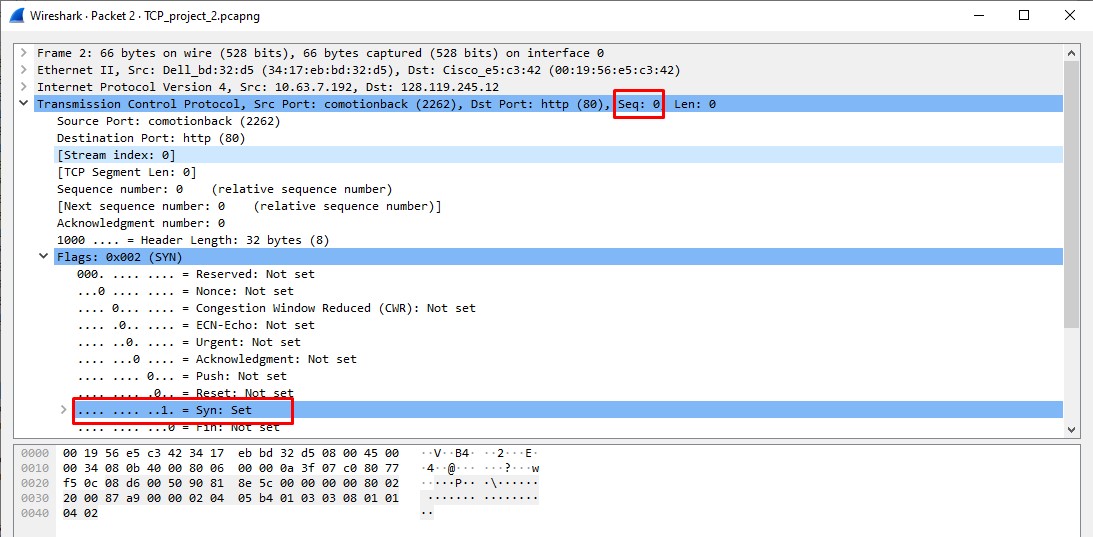
The IP protocol number for UDP is 0x11 hex, which is 17 in decimal value.



# Part 2: TCP

## Question 1:

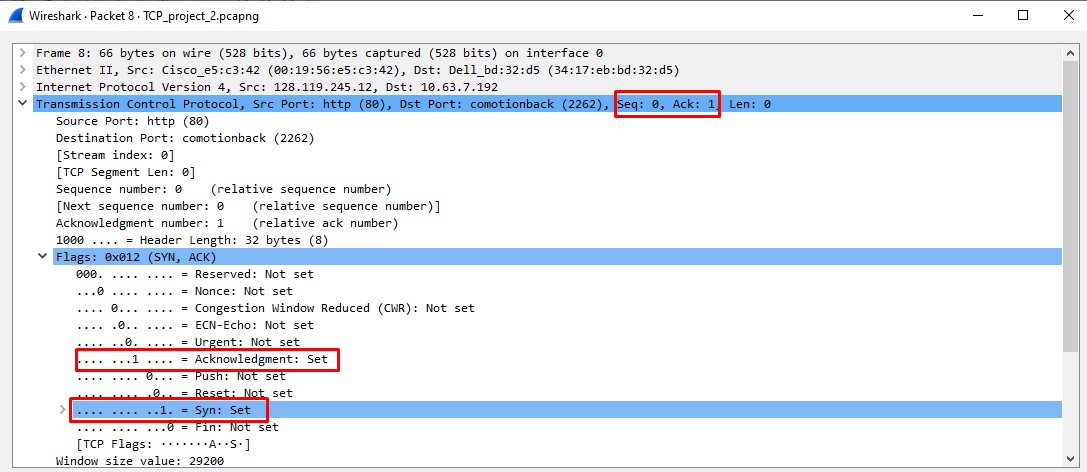
The sequence number of the TCP SYN segment is 0 since it is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu



According to screenshot, in the Flags section, the SYN flag is set to 1 indicating that this segment is a SYN segment.

## Question 2:

Sequence number of the SYNACK segment from gaia.cs.umass.edu to the client computer in reply to the SYN has the value of 0 in this trace. The value of the ACK in SYNACK segment is 1.

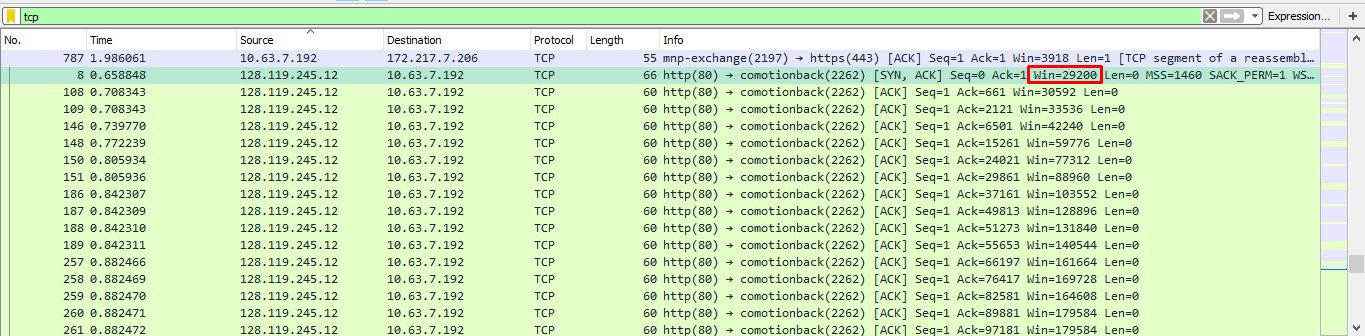


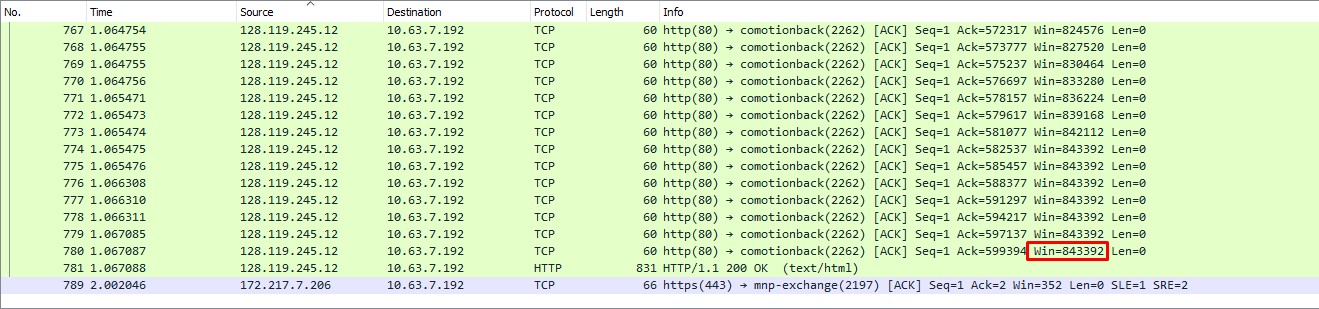
The value of the ACK field in the SYNACK segment is determined by gaia.cs.umass.edu by adding 1 to the initial sequence number of SYN segment from the client computer (i.e. the sequence number of the SYN segment initiated by the client computer is 0).

The SYN flag and Acknowledgement flag in the segment are set to 1 and they indicate that this segment is a SYNACK segment.

## Question 3:

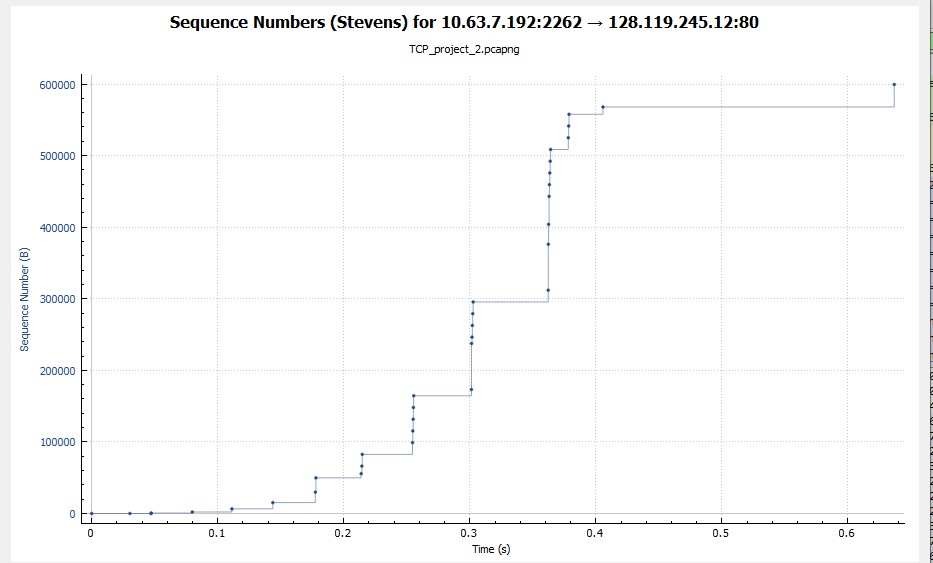
The minimum amount of buffer space (receiver window) advertised at gaia.cs.umass.edu for the entire trace is 29200 bytes, which is shown in the first acknowledgment from the server. This receiver window grows steadily until a maximum receiver buffer size of 843392 bytes.





## Question 4:

There are no retransmitted segments in the trace file. We can verify this by checking the sequence numbers of the TCP segments in the trace file. In the *Time-Sequence-Graph (Stevens)* of this trace, all the sequence numbers from source (10.63.7.192) to the destination (128.119.245.12) are increasing with respect to time. If there is a retransmitted segment, the sequence number of this retransmitted segment should be less than those of its neighbouring segments.

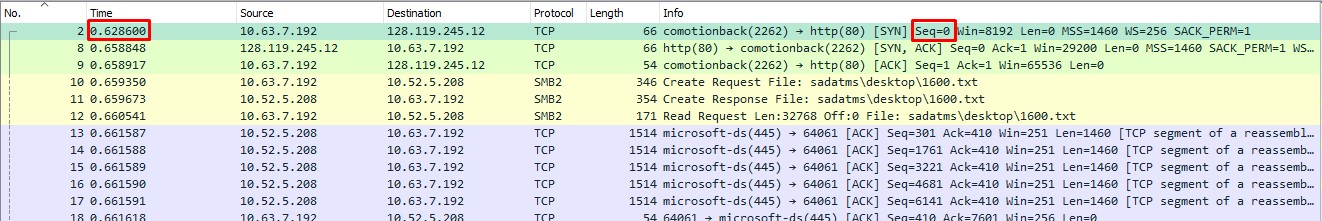


## Question 5:

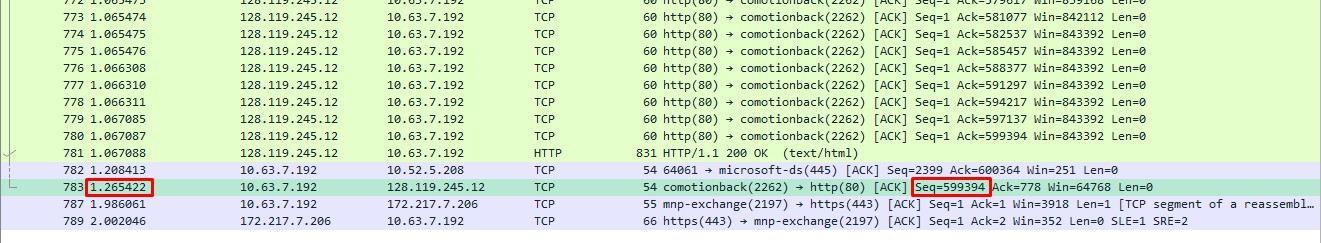
Throughput = Total amount of data/Total transmission time.

Using Wireshark, the total amount of data transmitted can be computed by the difference between the sequence number of the first TCP packet (M for frame #X) and the acknowledged sequence number of the last ACK (N for frame #Y). So, total data = (N - M) bytes

The total transmission time is the difference between the time instances of frame #X (t1) and frame #Y (t2). So, total time = (t2 – t1) seconds.

In this packet trace, M = 0 bytes for frame #2 and t1 = 0.628600 seconds 

N = 599394 bytes for frame #783 and t2 = 1.265422 seconds



Throughput = (N-M)/(t2-t1)

= (599394 – 0)/(1.265422 – 0.6286)

= 599394/0.636822 bytes per second

= 941,226.9 bytes per second

= 941.2 K bytes per second.