EECS 489 - FA 24
Discussion 6

TCP

Logistics

Assignment 2

Due: Friday, Oct. 11 (right before Fall Break)

- Groups of 1-3 people
- 3 AG submits per day
- 3 late days across all 3 remaining projects
- **Order of magnitude longer and more difficult than Assignment 1!**

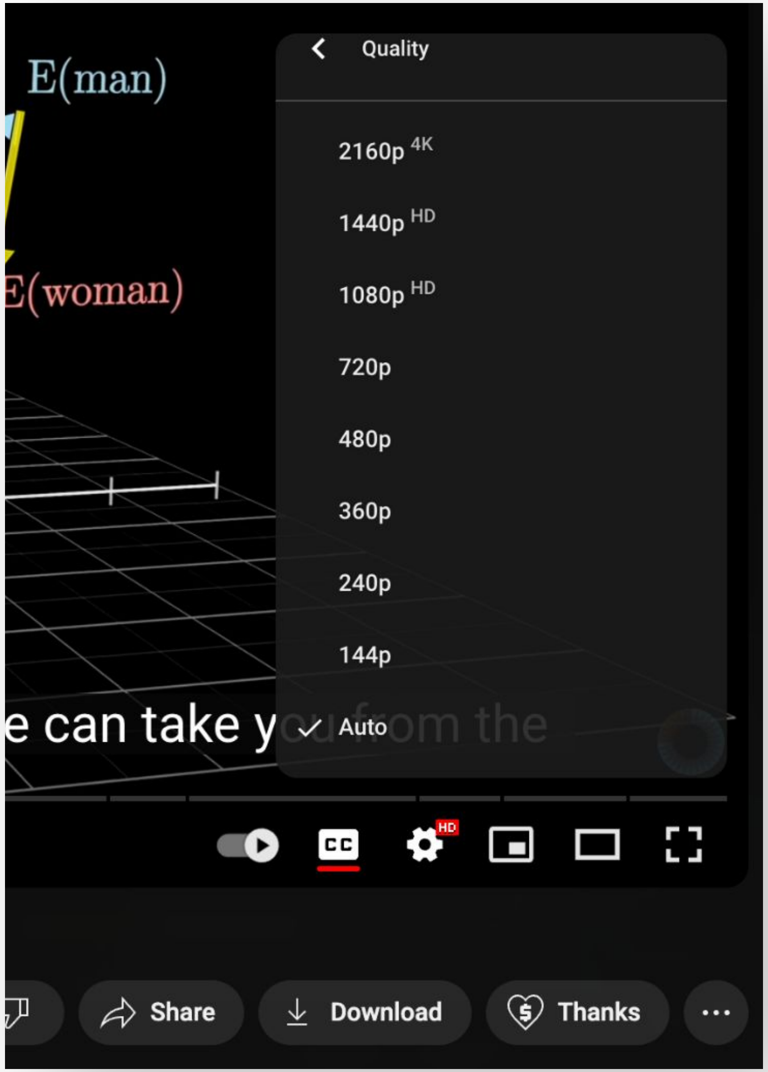
How to Develop:

- Develop on AWS like Project 1; you will have to set up a VNC Client on your computer, as we need a GUI for this project.

Assignment 2

Part 1: HTTP Proxy for video streaming

- Sits between a client (web browser) and a standard video server.
- Modifies client requests to the video server to ask for the correct bitrate, based on constant measurement; echoes replies from the video server back to the client.
- Should be able to deal with multiple concurrent client connections!



Assignment 2

Part 2: DNS Server with Load-Balancing

- Serves requests only for the URL of the video server
- Load-balances to different IPs using two strategies (configurable when the executable is called)
  - **Strategy 1:** Round robin
  - **Strategy 2:** Geographical proximity
- No hierarchical DNS lookup – all queries go directly to this load-balancing server, which responds with an A-record

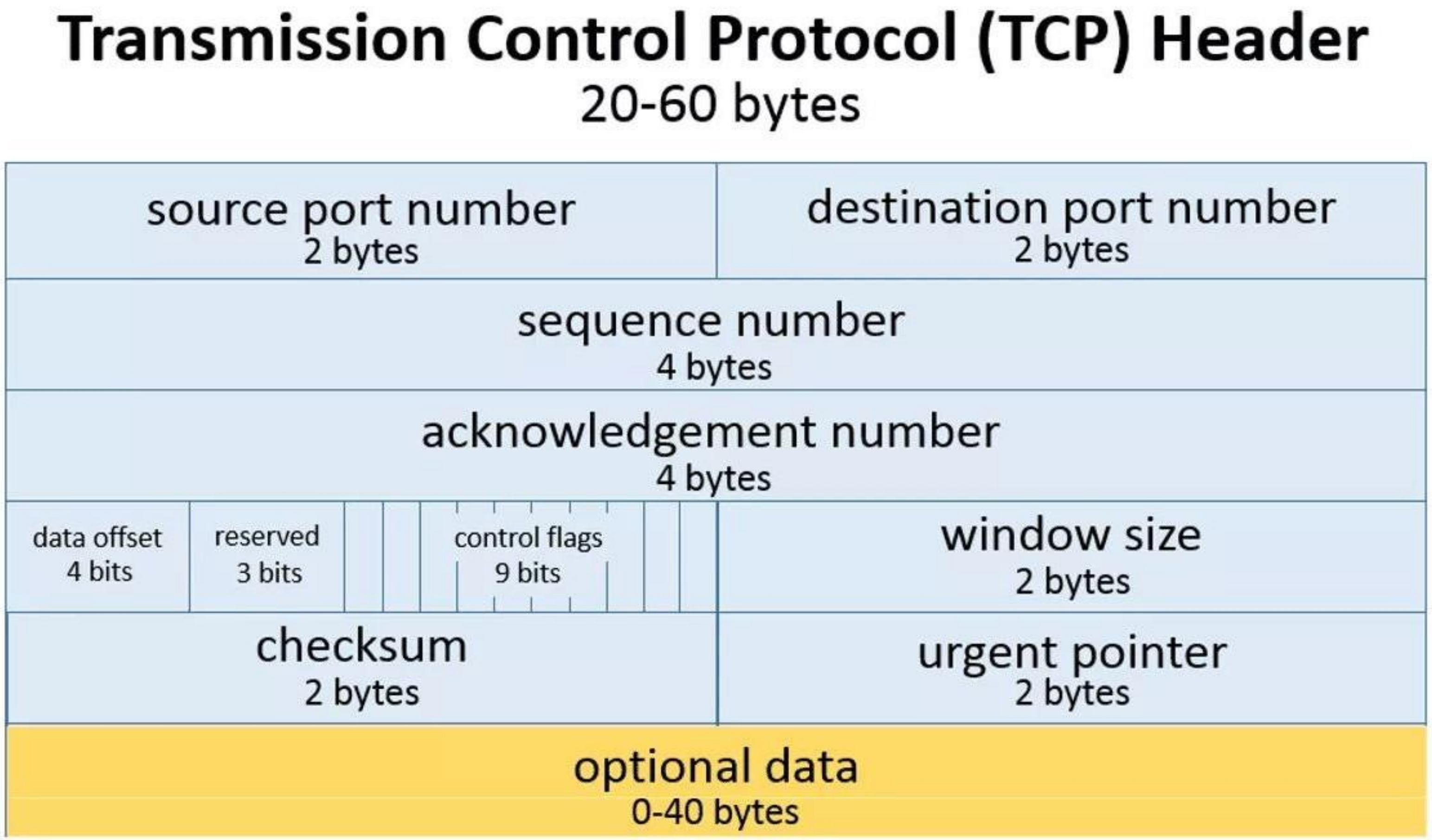
Assignment 2

Integrating Part 1 and Part 2

- Have the HTTP proxy call the DNS server when a new client connects to figure out which server to route that client's traffic to.
- You can work on Part 1, Part 2 independently and then integrate at the end – both are testable on their own.

TCP

TCP: Transmission Control Protocol



TCP: Transmission Control Protocol

- Provides a layer of **reliable transport.**
- Modeled as a **bytestream** rather than a stream of packets (even though this bytestream is divided into packets in practice).
- Provides **congestion control** mechanisms to dynamically discover the optimal speed for sending data.

TCP: Transmission Control Protocol

- State variables maintained at sender:
  - **Congestion Window Size (CWND):** The actual window size is the minimum of CWND and the window size advertised by the receiver (RWND).
  - **Slow-start threshold (ssthresh):** The threshold at which the CWND update rate changes.
  - **Timer:** Standard sender-side timer for detecting timeout.
  - **Duplicate ACK Count:** How many duplicate ACKs have been received (0, 1, 2, 3). This indicates a one-off packet drop.

TCP: Transmission Control Protocol

- Events to update sender state:
  - **ACK** for new data
  - **Timeout** on sender-side
  - **Duplicate ACKs**

TCP: Transmission Control Protocol

- **ACK** for new data
  - Good! Increase CWND.
- **Timeout** on sender-side
  - Bad! Decrease CWND and recalibrate ssthresh.
- **Duplicate ACKs**
  - OK (implies one-off loss). Retransmit missing packet and decrease CWND slightly.

Q1: TCP File Transfer 1

Consider transferring an enormous file of **L** bytes from host A to host B.

What is the maximum value of **L** such that we don't run out of TCP sequence numbers?

Note: TCP sequence number is 4 bytes in the header.

Q1: TCP File Transfer 1

Consider transferring an enormous file of **L** bytes from host A to host B.

What is the maximum value of **L** such that we don't run out of TCP sequence numbers?

Note: TCP sequence number is 4 bytes in the header.

2^32 Bytes

Q1: TCP File Transfer 1

Consider transferring an enormous file of **L** bytes from host A to host B.

Given **L** = 2^32 bytes, find how long it takes to transmit the file Assume:

- MSS (max segment size) = 1460 Bytes
- MTU (max transmission unit) = 1500 Bytes
- 128 Mbps link from A to B
- Ignore flow and congestion control, assume A sends as fast as possible contiguously.

Q1: TCP File Transfer 1

Consider transferring an enormous file of **L** bytes from host A to host B.

Given **L** = 2^32 bytes, find how long it takes to transmit the file Assume:

- MSS (max segment size) = 1460 Bytes
- MTU (max transmission unit) = 1500 Bytes
- 128 Mbps link from A to B
- Ignore flow and congestion control, assume A sends as fast as possible contiguously.

(2^32 / 1460 * 1500) / (128 *10^6 / 8) = **276 s**
amount of data / bandwidth

Q2 TCP Segment Metadata

Host A (sender) and B (receiver) are communicating over a TCP connection.
Assume the following events happen in order:

- B has received the first 127 bytes of the flow from A, this consumes seq num 0-126
- A then sends two segments, S1 (80 bytes of data), S2 (40 bytes of data)
- S1 has sequence num 127, source port 30302, destination port 80
- B sends ACK1 and ACK2 to A when it receives the first / second segment respectively

Assume S1 and S2 arrive in order.

For S2, what are the sequence num, source port and destination port?

Q2 TCP Segment Metadata

Host A (sender) and B (receiver) are communicating over a TCP connection.
Assume the following events happen in order:

- B has received the first 127 bytes of the flow from A, this consumes seq num 0-126
- A then sends two segments, S1 (80 bytes of data), S2 (40 bytes of data)
- S1 has sequence num 127, source port 30302, destination port 80
- B sends ACK1 and ACK2 to A when it receives the first / second segment respectively

Assume S1 and S2 arrive in order.

For S2, what are the sequence num, source port and destination port?

Sequence Num: 207

Source Port: 30302

Destination Port: 80

Q2 TCP Segment Metadata

Host A (sender) and B (receiver) are communicating over a TCP connection.
Assume the following events happen in order:

- B has received the first 127 bytes of the flow from A, this consumes seq num 0-126
- A then sends two segments, S1 (80 bytes of data), S2 (40 bytes of data)
- S1 has sequence num 127, source port 30302, destination port 80
- B sends ACK1 and ACK2 to A when it receives the first / second segment respectively

Assume S1 and S2 arrive in order.

For ACK1, what are the ack num, source port and destination port?

Q2 TCP Segment Metadata

Host A (sender) and B (receiver) are communicating over a TCP connection.
Assume the following events happen in order:

- B has received the first 127 bytes of the flow from A, this consumes seq num 0-126
- A then sends two segments, S1 (80 bytes of data), S2 (40 bytes of data)
- S1 has sequence num 127, source port 30302, destination port 80
- B sends ACK1 and ACK2 to A when it receives the first / second segment respectively

Assume S1 and S2 arrive in order.

For ACK1, what are the ack num, source port and destination port?

ACK Num: 207
Source Port: 80
Destination Port: 30302

Q2 TCP Segment Metadata

Host A (sender) and B (receiver) are communicating over a TCP connection.
Assume the following events happen in order:

- B has received the first 127 bytes of the flow from A, this consumes seq num 0-126
- A then sends two segments, S1 (80 bytes of data), S2 (40 bytes of data)
- S1 has sequence num 127, source port 30302, destination port 80
- B sends ACK1 and ACK2 to A when it receives the first / second segment respectively

Now assume S1 and S2 come out of order
For ACK1, what are the ack num, source port and destination port?

Q2 TCP Segment Metadata

Host A (sender) and B (receiver) are communicating over a TCP connection.
Assume the following events happen in order:

- B has received the first 127 bytes of the flow from A, this consumes seq num 0-126
- A then sends two segments, S1 (80 bytes of data), S2 (40 bytes of data)
- S1 has sequence num 127, source port 30302, destination port 80
- B sends ACK1 and ACK2 to A when it receives the first / second segment respectively

Now assume S1 and S2 come out of order
For ACK1, what are the ack num, source port and destination port?

ACK Num: 127
Source Port: 80
Destination Port: 30302

Q2 TCP Segment Metadata

Host A (sender) and B (receiver) are communicating over a TCP connection.
Assume the following events happen in order:

- B has received the first 127 bytes of the flow from A, this consumes seq num 0-126
- A then sends two segments, S1 (80 bytes of data), S2 (40 bytes of data)
- S1 has sequence num 127, source port 30302, destination port 80
- B sends ACK1 and ACK2 to A when it receives the first / second segment respectively

What is the likely application-layer protocol being used in this communication?

Q2 TCP Segment Metadata

Host A (sender) and B (receiver) are communicating over a TCP connection.
Assume the following events happen in order:

- B has received the first 127 bytes of the flow from A, this consumes seq num 0-126
- A then sends two segments, S1 (80 bytes of data), S2 (40 bytes of data)
- S1 has sequence num 127, source port 30302, destination port 80
- B sends ACK1 and ACK2 to A when it receives the first / second segment respectively

What is the likely application-layer protocol being used in this communication?

HTTP (Port 80).

Q3 TCP CWND

Consider sending a large file over a lossless TCP connection. Assume:

- TCP uses AIMD for congestion control with slow start
- **ssthres** = 16 MSS (Maximum Segment Size)
- Approximately constant RTT
- CWND starts at 1 MSS
- On ACK for new data:
  - CWND += 1 when CWND < ssthresh
  - CWND += 1/CWND otherwise

How long does it take for CWND to increase from 1 MSS to 20 MSS?

Q3 TCP CWND

Consider sending a large file over a lossless TCP connection. Assume:

- TCP uses AIMD for congestion control with slow start
- **ssthres** = 16 MSS (Maximum Segment Size)
- Approximately constant RTT
- CWND starts at 1 MSS
- On ACK for new data:
  - CWND += 1 when CWND < ssthresh
  - CWND += 1/CWND otherwise

How long does it take for CWND to increase from 1 MSS to 20 MSS?

CWND: 1 2 4 8 16 17 18 19 20 → **8 RTT**

Q3 TCP CWND

Consider sending a large file over a lossless TCP connection. Assume:

- TCP uses AIMD for congestion control with slow start
- **ssthres** = 16 MSS (Maximum Segment Size)
- Approximately constant RTT
- CWND starts at 1 MSS
- On ACK for new data:
  - CWND += 1 when CWND < ssthresh
  - CWND += 1/CWND otherwise

What is the average throughput (in terms of MSS and RTT) of the above process?

Q3 TCP CWND

Consider sending a large file over a lossless TCP connection. Assume:

- TCP uses AIMD for congestion control with slow start
- **ssthres** = 16 MSS (Maximum Segment Size)
- Approximately constant RTT
- CWND starts at 1 MSS
- On ACK for new data:
  - CWND += 1 when CWND < ssthresh
  - CWND += 1/CWND otherwise

What is the average throughput (in terms of MSS and RTT) of the above process?

(1+2+4+8+16+17+18+19) / 8 = **85 MSS / 8 RTT**

Thanks

Have a good one!