HTTP/3 LAB

Pre-Lab Report

Bar Binyamin Varsulker,

Ori Meir Kushnir,

Answers to Preparation Questions

a) The main difference between TCP and UDP is that TCP is connection-oriented and ensures reliable, ordered delivery of data, while UDP is connectionless and doesn’t guarantee delivery or order. TCP handles retransmissions and flow control, whereas UDP is much lighter and faster, but with less reliability.

b) TCP is great when you need accuracy and reliability, like with web pages or file transfers. The downside is that it's slower and more resource-intensive because of its handshake and error-checking mechanisms. UDP, on the other hand, is faster and better for real-time applications like video streaming or online gaming, but it doesn’t care if some packets get lost along the way.

c) QUIC (Quick UDP Internet Connections) is a modern transport protocol developed by Google, designed to improve the performance of connection-oriented web traffic. It’s built on top of UDP, not TCP, and includes features like multiplexing and encryption by default. Basically, it brings TCP-like reliability and TLS-level security over a faster, simpler UDP foundation.

d) HTTP/3, which is based on QUIC, has several advantages over HTTP/1.1 and HTTP/2. It reduces latency, avoids Head-of-Line Blocking (and TCP-level HOL), supports faster connection establishment, and maintains better performance on unstable networks—like when switching from Wi-Fi to mobile data. It also simplifies encryption by integrating TLS directly into the transport layer.

e) HTTP/2 solves several key problems that exist in HTTP/1.1, especially performance-wise. One of the main issues in HTTP/1.1 was Head-of-Line Blocking, where one delayed request could block all subsequent requests on the same connection, causing significant delays in page load times. HTTP/2 resolves this by allowing multiple requests and responses to be sent in parallel over a single connection, known as multiplexing, which helps avoid this blocking.

Another issue with HTTP/1.1 was the overhead caused by having multiple TCP connections open for multiple requests. HTTP/2 eliminates the need for this by reusing a single connection for all requests, reducing latency and improving overall efficiency.

Additionally, HTTP/2 compresses the headers of requests and responses, which reduces the amount of redundant data being sent and further boosts performance. It also introduces stream prioritization, which lets browsers prioritize important resources and load them faster.

In addition to the performance improvements, HTTP/2 also addresses some of the security issues present in HTTP/1.1. One of the significant security concerns in HTTP/1.1 was that the protocol didn't mandate encryption, meaning that data could be transmitted in plain text, making it vulnerable to interception, eavesdropping, and man-in-the-middle attacks. With HTTP/2, while encryption is not strictly required by the protocol itself, it is typically used with TLS in practice, especially in modern implementations, which ensures that data is encrypted and secure during transmission.

This shift towards encryption helps protect the privacy and integrity of data, making it much harder for attackers to access sensitive information. Moreover, HTTP/2's multiplexing also improves security in certain scenarios, such as preventing certain types of attacks that could exploit multiple open connections in HTTP/1.1. By using a single connection for multiple streams, HTTP/2 reduces the attack surface for potential threats.

f) Head-of-Line Blocking is when one packet delay or loss prevents others behind it from being processed, even if they arrived correctly. In TCP, this happens because all data has to arrive in order, so if one packet is missing, everything stops and waits. It can really slow things down, especially with multiple streams over one connection like in HTTP/2.

g) HTTP/3 solves Head-of-Line Blocking by using QUIC, which allows each stream of data to be delivered independently. So if one stream gets delayed, it doesn't hold up the others—everything can keep moving smoothly.

h) SCTP (Stream Control Transmission Protocol) is a transport-layer protocol used for message-oriented communication. It supports multi-homing and multi-streaming and is designed to transport PSTN signaling messages over IP networks, among other uses.

The fields in an SCTP header:

- **Source Port (16-bit)**
- **Destination Port (16-bit)**
- **Verification Tag (32-bit)** - Uniquely identifies an SCTP association. Prevents spoofing by ensuring that SCTP packets are accepted only if they belong to a valid, established association. Acts similarly to a connection ID in other transport protocols like QUIC.
- **Checksum**

Each chunk in the SCTP packet also has its own header, which includes fields like Chunk Type, Chunk Flags, and Chunk Length.

i) Multi-homing in SCTP refers to the ability of an endpoint to support multiple IP addresses. This provides redundancy and increased reliability, as if one network path fails, SCTP can switch to an alternate IP address.
Multi-streaming allows data to be divided into multiple independent streams within a single SCTP association. This helps prevent head-of-line blocking because if one stream is delayed, it does not block the others.

j) SCTP chunks are data structures used in SCTP packets. Each chunk contains a specific type of control or user data. A single SCTP packet may contain multiple chunks.

- DATA chunk — carries user data
- INIT chunk — initiates an association between SCTP endpoints
- INIT ACK chunk — acknowledges association establishment
- SACK chunk — acknowledges received DATA chunks and informs the peer endpoint of gaps in the received subsequences of DATA chunks
- HEARTBEAT chunk — tests the reachability of an SCTP endpoint
- HEARTBEAT ACK chunk — acknowledges reception of a HEARTBEAT chunk

- ABORT chunk — forces an immediate close of an association
- SHUTDOWN chunk — initiates a graceful close of an association
- SHUTDOWN ACK chunk — acknowledges reception of a SHUTDOWN chunk
- ERROR chunk — reports various error conditions
- COOKIE ECHO chunk — used during the association establishment process
- COOKIE ACK chunk — acknowledges reception of a COOKIE ECHO chunk
- SHUTDOWN COMPLETE chunk — completes a graceful association close

k) **Advantages of SCTP:**

- It is a full- duplex connection (i.e. users can send and receive data simultaneously).
- It has properties of both TCP and UDP protocol.

Disadvantages of SCTP:

- One of key challenges is that it requires changes in transport stack on node.
- Applications need to be modified to use SCTP instead of TCP/UDP.

l) A Docker container is a lightweight, isolated environment that includes everything needed to run a specific application: code, libraries, and settings. Unlike a virtual machine, it doesn’t include a full operating system, so it’s much faster and uses fewer resources. Containers share the host OS kernel, which makes them efficient and portable.

m) Docker is very useful for running an HTTP server in a lab because it lets you spin up a server quickly, with no setup headaches. Everyone can run the exact same environment, regardless of their host system. It keeps the lab clean, consistent, and easy to debug.