Computer Networks: Chapter 3 Summary (MIDTERM VERSION)

Author: Shun (@shun4midx)

Transport-Layer Services

Transport services provide logical communication between application processes running on different hosts. Two transport protocols available to internet applications are **TCP and UDP**.

Overview

In transport protocols, the **sender breaks application messages into segments** and passes it to the **nextwork layer**, whereas the **receiver reassembles segments into messages**, and passes it into the **application layer**.

Transport Layer vs Network Layer

The **transport layer** is about the **communication between processes**, whereas the **network layer** is about the **communication between hosts**.

Multiplexing and Demultiplexing

To allow multiple applications to use the network simultaneously, the transport layer performs both **multiplexing** and **demultiplexing**.

Definitions

- **Multiplexing (Sender side):** Gathering data from multiple sockets, adding transport headers (with port numbers and addresses), and passing the resulting segments to the network layer. **Small to large transport.**
- **Demultiplexing (Receiver side):** Using header information (source/destination IP and port numbers) to deliver received segments to the correct socket/application process. **Large to small transport.**

Header Information

Each transport segment includes:

- Source **IP address**
- Destination **IP address**
- Source **port number**
- Destination **port number**

These fields are used by the receiver to identify the **appropriate receiving socket**.

Connectionless vs Connection-Oriented Demultiplexing

- **UDP (Connectionless):** Each segment is directed to a socket based only on its **destination port number**. Multiple senders sending to the same port reach the same receiving socket.
- **TCP (Connection-Oriented):** Each connection is identified by a unique **4-tuple**:

  (Source IP, Source Port, Destination IP, Destination Port)

  A server can distinguish multiple TCP connections on the same port (e.g., port 80 for multiple clients).

[Safe for Exam Criteria] UDP Checksum (Safety Addition)

The **UDP checksum** provides simple error detection. Its purpose is to detect bit errors, not correct them.

Overview

It is computed by treating the UDP segment as a sequence of 16-bit integers, summing them using **one's-complement arithmetic**, and taking the **one's complement** of the result.

At the receiver, all words (including the checksum) are summed again—if the result is all 1s, no error is detected.