

Midterm Prep

Non-exhaustive summary for the first half of computer networks course.

Introduction

- **Computer network:** collection of computing devices that are connected in various ways to communicate and share resources.
 - **Types of networks (ordered by scale):** PAN, LAN, MAN, WAN.
 - **Network topologies:** point-to-point, bus, ring, star, tree, mesh, hybrid.
- **Internet:** the largest computer network of interconnected ISPs.
 - **Internet Service Providers (ISP):** provide services including Internet access and web hosting.
- **Network structure:**
 - **Network edge:** network-connected hosts/end-systems including PCs and smart phones.
 - **Network core:** networking hardware including routers, switches, bridges, and hubs.
 - **Communication links:** physical wired/wireless media that connect nodes (hosts/hardware) through the medium. Mediums include:
 - **Twisted-pair cables** (DSL, telephone lines).
 - **Coaxial cables** (DOCSIS, TV lines).
 - **Optical fiber cables** (Ethernet, dedicated Internet lines).
 - **Radio waves** (IEEE 802.11, wireless communications).
- **Network protocol:** a set of rules to support communication between network applications, including how to format, transmit, and receive exchanged data.
 - **Persistent protocol** establishes a persistent connection before sending data.
 - **Stateless protocol** in which the receiver (usually a server) must not retain session state from previous sender requests.
- **Internet protocol suite:** a theoretical model abstracting the networking functionality by organizing network protocols into several layers.
 - **OSI model:** the standard 7-layer model including the application, presentation, session, transport, network, data link, and the physical layer.
 - **TCP/IP model:** the practical and more popular 5-layer model that merges the functionality of the application, presentation, and the session layers from the OSI model into one (application) layer.
- **Network delays:** end-to-end delay between two hosts is the summation of transmission, propagation, queuing, and processing delays.

Application Layer

- **Application architecture**
 - **Client/Server architecture:**
 - **Server:** an always-on host machine with a fixed name or address providing services to clients.
 - **Client:** a host machine that initiates the communication with a server to use its services.
 - **Peer-to-Peer architecture:** a decentralized architecture where every node can act as a client and a server to collaborate and share resources.
- **Some application-layer protocols**
 - **HTTP:** a stateless protocol used for retrieving Internet resources such as HTML web pages, images, and other media.
 - **URI:** defines a standard syntax for identifying web resources.
 - **HTTP request methods:** include GET, POST, PUT, and DELETE.
 - **HTTP response classes:** informational, success, redirection, client error, server error.
 - **HTTP cookies:** key-value pairs set by servers and resent by the browser for session management, personalization, and tracking.
 - **FTP:** a stateful protocol that allows clients to upload/download files to/from a server. It uses two separate TCP connections for control and data and may operate in active or passive modes.
 - **DNS protocol:** a stateless protocol for querying the distributed DNS database that stores resource records used for IP/hostname translation, host aliasing, and load balancing.
 - **Mail Agent (software):**
 - MUA (user), MSA (submission), MTA (transfer), MDA (delivery), MRA (retrieval).
 - **Email protocols:**
 - **SMTP:** used for sending mail from MUA to MSA and from MSA to MTA.
 - **IMAP:** used by a MRA/MUA to view and manipulate mail stored on a server.
 - **POP3:** used by a MDA to download arrived email from server to client.

Transport Layer

- **Transport layer** is responsible for directing network traffic (divided into segments) to the destination process running on a host and listening on a certain port.
 - **Port numbers:** 16-bit number representing an entry point (address) of a process
 - **System ports (0-1023):** used by well-known services such as HTTP and FTP. Require superuser privileges to be bound to processes.
 - **Registered ports (1024-49151):** assigned by IANA to services that requested them.
 - **Dynamic ports (49152-65535):** available for temporary and private usages.
 - **UNIX/POSIX/Berkley Sockets:** a software structure identifying a connection between two process using the tuple (src IP, dst IP, src port, dst port, protocol).
- **User Datagram Protocol (UDP):** a simple transport-layer protocol for sending arbitrary messages over an IP network.
 - UDP is connectionless, unreliable, and does not implement error recovery.
 - UDP segments are known as datagrams.
 - UDP is suitable for time-sensitive applications that prefer speed over reliability. Examples include media streaming, video conferencing, and online gaming.
 - UDP uses checksums to verify data integrity.
 - Checksum is the one's complement sum of the packet data + metadata.
 - Sender calculates checksum and sends it along with the data for the receiver to verify the data integrity and discard invalid packets.
- **Transmission Control Protocol (TCP):** transports logically-related segments between processes as a stream of data.
 - TCP guarantees data to arrive in-order, without errors, and without duplicates.
 - TCP is connection-oriented, reliable, and implements error detection and correction.
 - TCP connection is established using a 3-way handshake (SYN, SYN/ACK, ACK).
 - TCP is suitable for applications that require reliability. Examples include file transfer, database transactions, and email delivery
 - TCP flow control ensures that a fast sender does not overwhelm a slow receiver
 - TCP congestion control ensures that the network will not be overwhelmed by data
 - TCP uses multiple mechanisms to build a reliable data channel on top of IP since IP packets may be delayed, dropped, or delivered with bit-errors. Such mechanisms include:
 - **Acknowledgement (ACK)** signals a received intact packet
 - **Negative Acknowledgement (NAK)** signals a received erroneous packet
 - **Sequence numbers** are used to ensure packet ordering and avoid duplication
 - **Timeout and retransmission** are used to recover from packet loss