CS 456 - S25 - Final Exam Cheat Sheet - Josiah Plett

This cheat sheet has been improved post-exam, for completion and relevancy to exam questions.

Good luck!

🟩**Performance Metrics** - End-to-End Delay:
d_{trans} = packet size / link rate
d_{prop} = link length / propagation speed (~2×10^8 m/s)
d_{proc} = node processing time (negligible)
d_{queue} = average queue length / service rate
Loss: fraction of packets dropped
Throughput: rate of successful data delivery end-to-end

🟩**TCP vs UDP**
TCP: reliable, connection-oriented, with flow and congestion control. Most popular. Header size: **20 bytes.**
UDP: connectionless, unreliable.
Neither guarantees *timing, throughput,* or *security*.

🟩**HTTP** - request-response model
Request: Method, URL, Version, headers, optional body
Methods: GET, POST, HEAD, PUT
Responses: 200 OK; 301 Moved Permanently; 304 Not Modified; 400 Bad Request; 404 Not Found; 505 Version Not Supported.
Non-persistent: each object requires a new TCP connection (2NRTT + sum of file transmission times).
Persistent: reuses one connection (first obj is 2RTT + file trans, next N-1 are RTT + file trans.
Persistentness is specified in the request header.

🟩**Web Caches**
Average response time = hit-prob•(cache-delay) + miss-prob•(server-delay + queuing-delay).
Queuing-delay scales with fraction of misses

🟩**HTTP/2 & HTTP/3**
HTTP/2: Decreases delay in multi-object reqs, mitigate head-of-line blocking by breaking objects into chunks.
HTTP/3: adds security and per-object congestion control over UDP.

🟩**Sockets** - IP address + port.
UDP: no handshake; receiver gets sender from packet.
TCP: Client contacts server, server dedicates socket.

🟩**Peer-to-Peer (P2P)**
Client Server distro time: $D_s \geq \max\{N\cdot F/U_s, F/d_{min}\}$
P2P distro time: $D_{p2p} \geq \max\{F/U_s, F/d_{min}, N\cdot F/(U_s+\sum u_i)\}$
d_{min} = *best client download time*. u_i = *client i upload speed*
BitTorrent tracks peers, peers prefer those with most/faster uploads to them, sometimes send to random.

🟩**Reliable Data Transfer (RDT)**
Stop-and-Wait (RDT 3.0)
Send one packet; wait for ACK (0 or 1). Timeout triggers retransmission. Utilization: $U = d_{trans}/(d_{trans}+RTT)$.
Go-Back-N (pipelined protocol)
Window of N packets; ACK(x) acks up to x; keeps N packets in flight; timeout resends the entire window.
Selective Repeat (pipelined protocol)
Independent ACK per packet; sender retransmits only unACKed packets; receiver buffers ACKs out-of-order.

🟩**Subnets & Forwarding**
Subnet notation (e.g. w.x.y.z/host). wxyz - host = subnet.
Routers forward based on the longest-prefix match.

🟩**TCP** - Transfer Control Protocol
Byte-oriented, cumulative ACKs -> seqnum per byte.
RTT Estimation
Exponential Weighted Moving Average: *(α≈0.125)*
$EstRTT_1 = (1-\alpha)\cdot EstRTT_0 + \alpha\cdot SampleRTT$
Deviation of RTT: *(β≈0.25)*
$devRTT = (1-\beta)\cdot DevRTT + \beta\cdot|SampleRTT-EstRTT|$
Timeout Interval = EstRTT + 4•DevRTT

🟩**TCP Optimizations**
Fast Retransmit: retransmit after 3 duplicate ACKs and no timeout; implies segment was lost.
Delayed ACK: wait a bit to combine ACKs, unless out of order then ACK immediately.

🟩**Flow Control**
Receiver advertises a window size *rwnd* and shares it in its ACKs, sender limits in-flight data accordingly.
Three-way Handshake: SYN, SYNACK, ACK
Teardown: FIN, FINACK

🟩**Congestion Control**
AIMD Additive Increase Multiplicative Decrease: +1 on success of full window, /2 on triple dup ACK, set to 1 on timeout (TCP value) then slow start threshold = wndw/2.
SS Slow Start: Double cwnd till loss or cwnd > ssthresh
TCP Cubic: cwnd growth based on cube of distance from curr window size before loss event
Delay-based: measured throughput near uncongested throughput? Increase linearly, otherwise decrease.
ECN Explicit Congestion Notification: routers mark packets with ECN flag on the ACK to receiver

🟩**QUIC** - Quick UDP Internet Connection
TCP-style congestion control/reliability at App layer over UDP with 1-RTT connection establishment w/built-in security. Avoids head-of-line blocking with multiple App-level streams multiplexed.

🟩**Routing Algorithms**
Link State
Each node floods link-cost information. Nodes compute shortest paths with Dijkstra's algorithm.
Distance Vector
Each node shares its distance estimates to all destinations with neighbors. Neighbors update via Bellman-Ford. Problems: link failure -> count-to-infinity, malicious node -> black hole.

🟩**IP Datagrams**
IPv4: Source + Dest, Time To Live (8 bits, prev. inf. loop).
IPv6: Source + Dest (128 bits), priority, flow label.
"Tunneling:" Wrap IPv6 in IPv4 if IPv6 isn't supported.

🟩 **Network Address Translation (NAT)**
Allows multiple devices in a private LAN to share one public/WAN IP. The NAT router keeps a translation table: (public IP:port) ↔ (private IP:port) for active connections.
- **WAN IP**: Public address visible on the Internet.
- **LAN IP**: Private address used inside local network.

Without special configuration, an external host **can't initiate** a direct connection to a NATed host because the NAT blocks unsolicited inbound traffic — it has no table entry until a local device sends something first.

🟩**Ethernet** (Data Link Layer)
Shared (old) or switched (modern) physical medium.
Frames carry source/destination MAC, type, and payload.
Media Access Control (MAC): 48-bit hardware address unique to each network interface (ie. internet connector).

🟩Interior / Inter-Domain Routing
Autonomous Systems (AS) run their own *interior* routing; *inter-domain* routing manages paths between ASes.

Open Shortest Path First (OSPF)
Link-state protocol used *inside* an AS. Link costs are set by the network administrator.

Border Gateway Protocol (BGP)
App-layer protocol for inter-AS routing over TCP via advertised paths. Advertisement contains: destination prefix, AS-PATH (current total path), NEXT-HOP.

eBGP: between ASes.

iBGP: propagate reachability internally in an AS.

Route selection: 1) local policy, 2) shortest AS-PATH, 3) nearest NEXT-HOP (hot-potato), 4) other tie-breakers.
"Hot-potato" routing minimizes cost to exit the AS.
BGP advertisements usually only forwarded if beneficial (eg. financial incentive).

🟩End-to-End Argument
The network's role is only packet transfer; reliability and other services belong at the endpoints (sender/receiver). Hence IP is *the* single network-layer protocol.

🟩Middleboxes
Routers that do extra stuff (firewalling, NAT, caching).

🟩Internet Control Message Protocol (ICMP)
Encapsulated in IP, carries error and control messages for network-level (e.g., "TTL expired," "Unreachable host").

🟩Software-Defined Networking (SDN)
Centralized controller codes routing rules for each node. Separates control from forwarding.

🟩Address Resolution Protocol (ARP)
Broadcasts a request across the LAN for a target IP's MAC — sent to ff:ff:ff:ff:ff:ff which is all 1s; "broadcast MAC address" — that IP replies with its MAC.

Local Area Network (LAN): local, privately managed.

🟩Link-Layer Switch
Self-learning device that forwards frames between devices in a LAN, based on a MAC-to-port table. If the destination is unknown, it broadcasts. Dedicated host connections prevent collisions.

🟩Virtual LANs (VLANs)
Port-based VLANs group ports into logical LANs. Trunk ports carry traffic for multiple VLANs, tagging with VLAN IDs (in Ethernet header); *isolation* and *organization*.

🟩Multiple Access Protocols (Data Layer)
Coordinate multiple senders on one medium.

Channel Partitioning: TDMA (time slots), FDMA (frequency bands).

Carrier Sense Multiple Access (CSMA):
- **Random Access:** just plain listen before sending. (eg. Aloha & Slotted Aloha →)
- **CSMA/CD: d**etect **c**ollisions, abort and back off exponentially ($k \in [0,2^m-1]$).
- **CSMA/CA:** wireless variant — **a**void **c**ollisions by using ACKs, send if idle, back off if busy/no-ACK.
- **Reservations:** RTS/CTS (Request to Send, Clear to Send) handshake to reserve the medium.

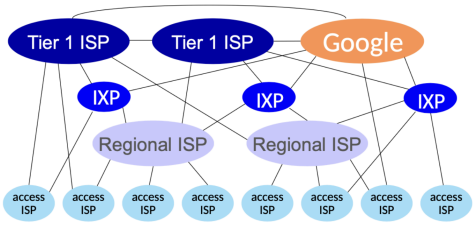
🟩Dynamic Host Configuration Protocol (DHCP)
Hosts broadcast DHCPDISCOVER; servers reply DHCPOFFER. The host requests with DHCPREQUEST; server confirms via DHCPACK. Renewal only requires REQUEST and ACK. It is in *App Layer*.
DHCP also supplies subnet mask, gateway, DNS server.

🟩Domain Name System (DNS)
Hierarchical name. IP mapping: Root → TLD → authoritative servers → hosts. Local DNS resolvers from ISP cache queries and perform iterative lookups.

Iterative: client→local, local→root, local→tld, local→etc
Recursive: client→root, root→tld, root→company, etc.
Resource Records requests: Name, Value, Type, TTL.
- **A:** hostname → IPv4 address
- **CNAME:** alias → canonical name
- **NS:** domain → authoritative name server
- **MX:** domain → mail exchange server

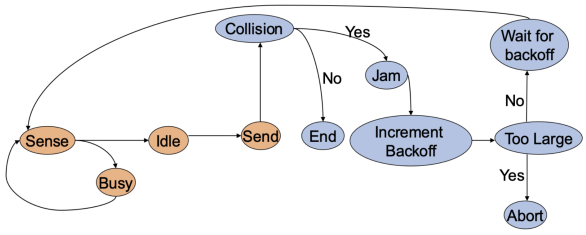
Messages: request and reply share the same format. Multiple questions and answers per message; transaction IDs match them.

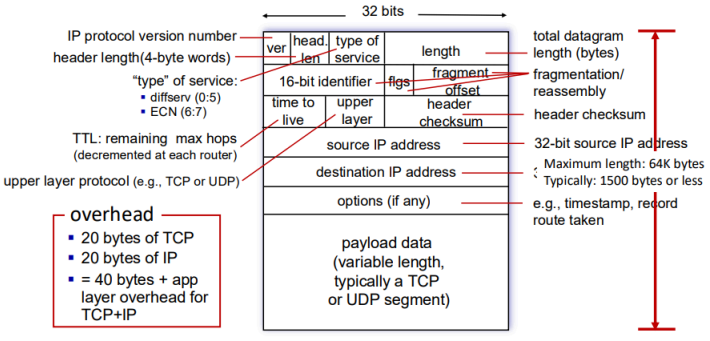
🟩Internet Service Provider (ISP)
Company that provides Internet access. Connects customer networks (LANs) to internet, assigns IPs, might offer DNS, email, and hosting services.



🟩Aloha (pre-CSMA wireless random access)
$p(1-p)^{2(N-1)}$ … Max efficiency possible: f(p) = 1/(2e) = 0.18

🟩Slotted Aloha
Probability *any* node has success = $Np(1-p)^{N-1}$
Max efficiency: p that maximizes above. Usually, p = 1/N
For many nodes, take N → ∞; f(p) = 1/e = 0.37

🟩CSMA/CD


🟩IPv4 Datagram Format


🟨Remaining things you should know:
- **Error Correction:** How 2D error correction works.
- **Error Detection:** CRC encoding + decoding.
- **Summary:** Write the full steps from connecting your PC to a new LAN to reading a webpage.