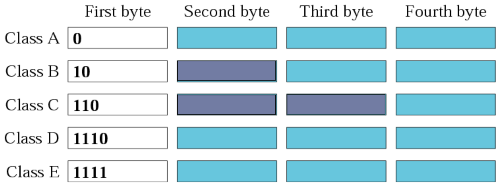
1 Introduction and Review

Example Web Request

- 32-bit IP address, 16-bit port #
- usually: well-known ports
- default router: DHCP or manual config
- Device driver reads eth header (send to IP)
- MSS = app payload size (=MTU – 40), agreed in TCP SYN
- MTU = IP payload size

Internet Address

- network + host address
- subnet mask
- 0.0.0.0 → this host (when first joining network)
- 127.0.0.1 → localhost
- 255.255.255.255 → limited broadcast (within subnet)
- .0: network identification
- network all 0s: current subnet
- host all 1s: direct broadcast to this subnet
- classful addressing



- move to classless interdomain routing (CIDR), any prefix length (variable length subnet mask, VLSM)
  . e.g. NUSNET 137.132, admin adds bits to divide into multiple subnets. Within each subnet, first/ last address not used (254 subnets, 254 hosts) as not sure if id/ broadcast is for whole NUSNET or one subnet
  . security, performance (limit broadcast)
- address blocks too small → many entries in routing table

Private IP Addresses

- globally non-routeable, any network class can use
- 10/8 (class A), 172.16/12 (class B), 192.168/16 (class C)
- Network Address Translation (NAT)
  . WAN/ LAN side translation
  . private IP/ port, external IP/port, protocol

2a Address Resolution Protocol

- Media Access Control (MAC) address in hop-to-hop delivery
- ARP broadcast: within a LAN
- To different subnet: R replace A's MAC
  . Proxy ARP: router replies w its own MAC
- carries ethernet header (type 8060)
- non-existing host: exponential retry, give up
- periodically refresh ARP cache
- Gratuitous ARP: send for its own IP
- no authentication, stateless (reply w/o request), nodes MUST update if entry exists (ARP poisoning)

2b Dynamic Host Configuration Protocol

- runs over UDP (server 67, client 68)
- static, auto (indefinite time), dynamic
- DISCOVER: broadcast to server, server does ping probe for IP address
- OFFER: broadcast w/ your-ip-addr
- REQUEST: broadcast, other servers can release
- ACK: send gratuitous ARP probe (in case duplicate IP assigned manually)
  . send 3, 4th broadcast to force update
- Client RENEW before lease expiry (request + ACK), fill in Client IP address
  . 50% timer, 87.5% timer = rebind (can request from any server)
- RELEASE: client move to another network
- client gets IP addr, netmask, gateway router, DNS
- Relay Agent: inefficient to have one DHCP server for each subnet
  . listens on port 67, 68
  . intercept DISCOVER, unicast to DHCP
  . add incoming IP to router-address, increase hop-count by 1 (notation for relay)
  . DHCP replies unicast to relay
- multiple DHCP servers: define scopes, for redundancy
- client can request specific IP (options), or specific server (Server IP)
- Flag format: 0 (unicast), 1 (broadcast), 15 0s
  . when client cannot accept unicast (boot)
- INFORM: additional info for client
- options: tag 0 for padding every option, end list with 255
  . e.g. tag53 = DHCP message type
  . e.g. tag51 = max lease time
- DHCP server stores (k, v) for each client
  . k = (IP-subnet, MAC)
  . v = (IP assigned, least time) – 0xFFFFFFFF means infinite
- address conflict avoidance – least recently used, ping probe, gratuitous ARP, reserve
- Lease expiration relative to client clock (expire time = request – ack)

2c VLAN (802.1q)

- physical → logical grouping
- between VLANs: router needed
- group using: port switching (static), MAC (too many entries in table), layer3 protocol (packets belong to VLANs, not stations)
- 2 methods: filter/ tag (more common)
  . ingress switch adds tag, egress removes
- 802.1q: add 4byte header after dest/src MAC
  . first 2 bytes=TPID (0x8100)
  . 3bit PCP (priority), 1bit DEI (drop indicator), 12bit VLAN ID (max = 4096 - 2 reserved)
  . double tagging to meet increased demand (or VxLAN, extensible with 24bit seg ID)
- default = VLAN0
- hosts must be in same subnet and VLAN
- trunk ports: allow switches to communicate, using VLAN Trunking Protocol (VTP)
  . server (create, delete), client (advertise configs), transparent (plain forwarding)
- architecture: 3tier → leaf-spine

3a Internet Protocol Options

- Differentiated Service (DS, 6bits)/ Explicit Congestion Notification (ECN, 2bits) – "type of service"
- HLEN: 4 bits, indicates options present
- TTL = hop count, 8 bits but usually << 100
- IP datagram transmitted row by row (network byte, big endian – small to big)
- IP payload must be at least 46 bytes, or padded (ethernet min. frame size for CD)
- Max. transmission unit (MTU) for fairness

IP fragmentation – at router or sending host
  . 3bit frag flags – D (do not frag), M (more)
  . fragment length must be divisible by 8
  . offset starts at 0, add fragment_length/8

Options – 1byte type, 1byte length
  . starts with 0 → copy only in first frag
  . 6 options defined, max 32
  . Ptr (starts at 4)
  . Record-route [type7]: routers add in IP of outgoing interface, contains Ptr (starts at 4) – Ptr = location to record IP address
  . Max = record 9 IP max. HLEN = 15 (max value), hence 60 bytes max. 3 bytes for compulsory information (type, length, ptr)
  . Strict-source-route [type137]: route to take, dest IP swapped with IP at offset of Ptr (e.g. dest1 = router1 incoming, Ptr = 4)
  . Loose-source-route [type131]

3b Internet Control Message Protocol

- carries IP header
- error-reporting back to source
  . copies previous IP header (contains dest address) and first 8 bytes of IP payload (contains source/ dest port)
- query messages (e.g. echo)
- only generated for first IP fragment
- not generated for ICMP packets, multicast/ broadcast/0.0.0.0, 127.x.x.x
- e.g. source-quench: dropped due to congestion, slow down sending
- e.g. time-exceeded: TTL exceed, fragments not complete
- e.g. ICMP redirect: router receives and sends packet through same interface
- e.g. timestamp query: synchronise two clocks (original ts, received ts, transmit ts) – RTT must be known
- e.g. echo request/ reply (ping): test network reachability

- e.g. traceroute: send UDP datagrams, progressively increase TTL
  . time exceeded → port unreachable
  . TCPtraceroute: due to firewalls

4 Network Applications

Hypertext Transfer Protocol (HTTP)

- port 80 over TCP, stateless
- 1.1: persistence, can have pipelining
- "HOST" in header if proxy cache used, identifies server that client is requesting from
- GET /index.html HTTP/1.1\r\n
- HTTP/1.1 200 OK\r\n (delimiter)
- Cookies: used for state management
  . response: set-cookie
- Web caches (proxy server)
  . reduce load on server
  . for each request, proxy sends conditional GET to server (if-modified-since date = date of last-modified object in cache)
  . reply = 304 Not Modified (empty body)

File Transfer Protocol (FTP)

- port 21 for control connection, 20 for data connection
- active mode: client specifies port for data connection, server connects to port
- passive mode: "EPSV" (due to firewall blocking ports)
- new data connection for each file transfer

Email Protocols (SMTP, POP, IMAP)

- User Agents, Message Transfer Agents (MTA), Message Access Agents (MAA)
- SMTP: upload mail to SMTP server, SMTP server has an MTA client to send to mailbox
- mail access using POP/ IMAP
- Web-based: HTTP → SMTP → HTTP
  . only mail servers use SMTP
- Post Office Protocol v3 (POP): download and delete, stateless across sessions
  . No active connection needed to read mail, reduce server storage
- Internet Mail Access Protocol (IMAP)
  . preserves user state across sessions (messages kept at server)

Domain Name System (DNS)

- hierarchical name space to avoid conflict
- in-addr.arpa.: reverse mapping (IP → name)
- domain names always end with "."
- domain = subtree of domain name space
- hierarchical storage, distributed databases
- AUTHORITY: authoritative NS
- over port 53, UDP/ TCP (zone transfers)
- local DNS server: acts as proxy w/ cache
- iterative = reply w/ name of server to contact (root/ local)
- recursive; e.g. authoritative NS
- ***Zone***: DNS responsibility/ authority area
  . primary server: create, maintain zone file
  . secondary server: zone transfer (TCP) – keeps a copy of the information
- ***Resource Records (RR)***
  . A: name to IP
  . NS: name of authoritative NS for domain
  . CNAME: canonical
  . MX: mail server for domain
  . SOA: start of authority (primary NS), contains email address of admin, only 1 SOA
  . PTR inverse mapping
- Registrars: accredited by ICANN
  . provide name, IP of authoritative NS
  . NS record, A record for NS
  . add A and MX record to authoritative NS
- Respond to changing IP address
  . Dynamic DNS, DHCP-configured hosts can update their IP address in the master file

5 Socket Programming

- UDP connections from multiple clients can be concurrent to same server port
- TCP connections: multiple ephemeral ports
- create socket descriptor, bind to IP + port

6 IoT Protocols and Applications

- Cyber-physical domain
- Challenges: security (privacy), legal, integration, network constraints, expertise
- Link layer: 6LOWPAN/ Bluetooth (low power needed)

CoAP: over UDP/ DTLS (secured)

- all 1s: no options
- Built-in discovery: .well-known/core
- CON (firmable)/ NON/ ACK/ RST
  . response can be piggybacked
- uses option delta: difference in option #
- Token to synchronise transactions
- congestion control (exponential rtx to 247s)
- Observerable (notify on state change)
- Block transfer (larger resource) – parties decide on block size [nr, m-more coming?]

<u>MQTT</u> publish/ subscribe topics with broker

7 Wireless LANs

- Full duplex difficult with wireless channels
- High f, more signal attenuation
- IEEE 802.11(ac)
- MIMO: multiple-input/output
- Beamforming: phasing for constructive intf
- CSMA/CD collision window: min. transmission time = 2 * propagation delay, exponential backoff
- Wireless: hidden node problem, signal attenuation, cannot use CSMA/CD

CSMA/CA

- Distributed Coordination Function (DCF)
- Try 16 times
- Wait IFS → Wait contention window (0 to 2^K -1) → Wait timeout for ACK
- Reservation Scheme (RTS/ CTS)
  . (S) DIFS → RTS → (D) SIFS → CTS → (S) DIFS → Data → (D) SIFS → ACK
  . CTS sent to all nodes, w/ info e.g. how long tx is needed
  . NAV: no carrier sensing
- ***Interframe Space*** – Distributed, Short, Point
  . DIFS: async contending access (longest)
  . SIFS: immediate response (given priority)
  . PIFS: used by controller in PCF scheme
- Exposed Node Problem: C conservative idle
  . after timeout, C sends RTS, but A is sending data, C cannot hear D's CTS
- Point Coordination Function (PCF) – poll
  . for time-sensitive tx (runs on DCF)
  . contention-free using Access Point as Point Coordinator
  . Start contention-free period using PIFS, unpolled PCF hosts go into NAV mode
  . Contention using DIFS, so DCF no starve

Network Infrastructure

- BSS: stations with same coordinating fn
  . no AP = adhoc network/ IBSS
  . AP = infrastructure network
  . separate collision domains
  . BSSID 6 octets, (MAC of AP/ random)
- ESS: one or more BSS, with a Distribution System (DS)
- Service Set Identifier (SSID)/ ESSID
  . 32 octet "network name"

8a TCP Congestion Control

- duplex, multiplexed, connection-oriented, reliable, flow-control, byte-stream service
- ACK: next expected byte (cumulative)
  . pure ACK: no data (SYN/ FIN – 1 byte)
  . SEQ: first byte sent (exclude ACK)
- duplicate data = discard
- Terminate: FIN → FIN/ACK → start 2 MSL (max segment lifetime) timer
  . ACK lost: server rtx FIN, client rtx ACK
  . Active close → can still receive data
- RST deny connection, RST+ACK abort
- Simultaneous Open/ Close
- Fast rtx: triple duplicate ACK
- Flow control: receive window (rwnd)

Congestion Control

- capacity estimation, self-clocking (ACK arrived → packet reached, lost packet → congestion)
- congestion window (cwnd) – unit = MSS
- ***Slow Start*** (after congestion, when session starts): increase cwnd by 1 with each ACK (effectively exponential)
  . ssthresh: max(2, min(s_cwnd, r_rwnd)/2) after loss, cwnd = 1
  . initial ssthresh = rwnd, 64KB (window scale now used to increase max rwnd)
- ***Capacity Proving*** (congestion avoidance): additive increase after ssthresh
- TCP Reno: Fast Recovery (3DUPACK)

  . cwnd/2 (multiplicative decrease)
  . (cwnd=ssthresh), additive increase
- ***TCP New Reno***:
  . cwnd = ssthresh + X MSS (X = no. dup ack, including first 3)
  . new data ACK: end fast recovery, cwnd = ssthresh

8b Open Shortest Path First (OSPF)

- Link-state routing based on bandwidth
  . vs RIP DV, hop count limit of 15
- Entire topology known (Dijkstra)
- Grouped into areas with Area BRs to limit flooding of link-state info, size of db
- Boundary of AS: ASBR

Neighbour Discovery and Maintenance

- Join: Send "Hello" to every interface (multicast to OSPF routers at 224.0.0.5 every 10s) – over IP, protocol 89 with 24byte header
- 40s no receive → link failed/ neighbour crash
- "Hello", IP/24, S (neighbours who said hi)

Exchange of Link-State Information

- Link State Advertisements (LSAs)
- add 20byte LSA header after OSPF header
- When: new neighbour, link down, cost changes, basic refresh every 30s
- sent by reliable flooding (TCP) w/ explicit ACK, sequence, timestamp
- LSA Database: R1 LSA, R2 LSA
  . identical within area
- Routing Table:
  . N2 IP2 U (outgoing interface of router)
  . N3 IP3 UG (incoming intf of gateway)

(Backup) Designated Router - Transient Link

- Routers only have DR/ BDR as neighbour
  . Use 224.0.0.6 for Network LSA

9a Network Security

- Internet built with trusting users in mind
- Sniff packets in promiscuous mode
- CIA, non-repudiation
- MAC = keyed hash
- Digital Signature: trusted 3p needed
- Authentication: challenge-response w/ nonce (let user encrypt random r, prevents replays)
- Symmetric key: use Key Distribution Center (KDC), everyone shares key w/ KDC, which generates and communicates session keys

$$K_{A\text{-}KDC}(R1,\ K_{B\text{-}KDC}(A,R1))$$

- Asymmetric key: use Certificate Authority (CA) hierarchy to certify public keys
- DoS measures: ingress filtering ("firewall"), SYN-flood: only allocate resources after client ACK (use SYN cookies to sync)

Network-Layer Security

- encryption not needed for ALL messages
- IPSec: create uni-directional Security Association (logical connection)
  . Transport: encode transport payload in IPSec header/ tail
  . Tunnel: protects original IP header, e.g. VPN (need not lay private connections)

Secure Sockets Layer (SSL)

- Runs over TCP (e.g. HTTPS): server auth, data encryption, optional client auth
- SSL hello after TCP setup → certificate → client sends encrypted pre-Master Secret
- MS used to generate 4 keys (bi-directional encrypt + auth, so compromises are limited) via key derivation function
- Encrypt/ auth algorithms set in "Hello"

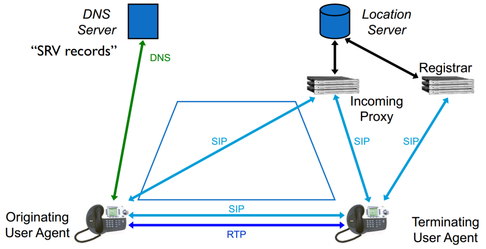
Pretty Good Privacy (PGP) – secure email
- Ks used as public key encryption is slow

10 Multimedia Protocols (SIP, VoIP)

- Is the media stored (unicast, on-demand), live (multiple unicast) or interactive?
- Separate web server and media server
- Session Description Protocol (SDP): metafiles, runs over SIP
  . media type/ format, transport protocol (RTP/ UDP…), how to receive files (address, port...)
- Real-Time Streaming Protocol (RTSP): steup, play, pause, teardown (over TCP as reliability needed for signaling protocol)
- Jitter: real-time delay; use Client-side Playback buffer (arrival time != play time)
- Real-time Transport Protocol (RTP): timestamp, sequencing, mixing (join audio + video), (over UDP)
  . Uses temporary even-# UDP port
  . In conjunction w/ RTCP for sender to monitor network conditions

Session Initiation Protocol (SIP)

- UDP port 5060
- App-layer signalling protocol
- Server types; redirect, proxy, registrar, location, session border controller
- INVITE, ACK, OPTIONS, CANCEL, BYE, REGISTER (inform registrar of IP)
- sip:bob@IP/email/phone#
- set up connection session, then use RTP to communicate (VoIP = SIP + RTP)
- via proxy: A invite proxy, proxy invites B
  . when A does not know B's IP address
  . proxy keeps track via registrar (which in turn uses location server as database)
  . A's proxy contacts B's registrar, invite is forwarded to B on A's behalf



  . DNS lookup for registrar/ incoming proxy
  . e.g. Asterisk SIP Server

CODEC (encode/ decode signals)
- GPS Enhanced G.711

Telephony – ITU-T H.323 (packet-based networks) e.g. Public Switched Telephone Networks (PSTNs)
- Gatekeepers (within Internet) – similar function as registrar, location server, proxy (address resolution) ↔ Gateway ↔ Telephone Network
- SIP has better extensibility and lower complexity

11 SDN

- Under-specification (best-effort) of network layer → difficult to change (closed hardware)
- Separate data plane (forwarding) and control plane (routing), management plane (config)
- Control plane establish state for data plane
- Logically centralised controllers
- Application Control Plane (North) ↔ Network Control Plane (Network OS) ↔ (South) Switch

OpenFlow (Southbound Interface)
- Central management using a controller
- Flow-based control using flow tables
- match → actions → update counters (stats)
- default: table-miss flow entry, else drop

Network Virtualisation (e.g. mininet)
- Network slicing: physical infrastructure shared by many virtual networks
- Server virt.: separate different functions
- Vswitch: switch between VMs (e.g. ovs)
- FlowVisor: proxy between multiple switches and multiple controllers