CS2105

AY24/25 Sem 2

by ngmh

Introduction

- The Internet is a network of connected computing devices / hosts that run network applications which talk to each other with protocols
- Hosts access the Internet through access networks
- Circuit-Switching: Dedicated end-end resources reserved between source and destination
- Packet-Switching (e.g. Internet): Host breaks message down into packets which are passed from one router to the next
- Transmission delay $= packet\ length / transmission\ rate$
- Internet is a Network of Networks
- Protocol Layers: Application, Transport (Process-Process), Network (Routing of datagrams from host to host, Link (Data transfer between neighbouring network elements), Physical (Bits on the wire)

Delay and Loss

- Packets queue in router buffers to be sent out, might be dropped if capacity is reached
- Processing Delay: Time to check bit errors, and determine output link
- Queueing Delay: Time waiting in queue for transmission
- Transmission Delay: L/R, time for packet to be fully sent out
- Propagation Delay: d/s, time for packet to travel across physical link
- End-End Delay is the sum of these four factors
- $traceroute$ displays path from source to destination by sending a series of small packets with different TTL
- Throughput: How many bits can be transmitted per unit time
- 1 byte = 8 its, Micro, Milli, Standard, Kilo, Mega, Giga, Tera, B = bytes, b = bits

Application Layer

- Client-Server Architecture: Server waits for incoming requests and provides service to clients who initiate contact
- P2P Architecture: No dedicated server, end systems communicate with each other, highly scalable but difficult to manage
- Serivce Criteria:
  - Data Integrity: Can app tolerate data loss e.g. file transfer v/s audio streaming
  - Throughput: How much bandwidth does the app need e.g. multimedia
  - Timing: Is the app time sensitive e.g. online games
  - Security: Does the app need encryption and data integrity
- Process Identification:
  - IP Address: Globally unique, identifies host, IPv4 is 32 bits split into 4 bytes in dotted decimal, IPv6 is 128 bits split into 8 sets of 2 bytes in : hexadecimal
  - Port Number: Locally unique, identifies process, 16 bit number

HTTP

- HyperText Markup Language (HTML): What
- Uniform Resource Locator (URL): Where
- HyperText Transfer Protocol (HTTP): How
- A webpage has a base HTML file and other referenced objects with their own URLs
- Uses TCP as a transport service
- RTT: Time for a packet to travel from client to server and back
- HTTP/1.0: New connection established for each resource, time taken is $2RTT + transmission\ time$ for each
- HTTP/1.1: Pipelining: New request is made before receiving response of old requests as soon as resource is encountered
- HTTP/1.1 Persistence: Connection left open after sending response, subsequent messages use the same connection
- HTTP/2 Multiplexing: Response can come back in any order, even partially
- Request:

```
GET /~cs2105/demo.html HTTP/1.1\r\n
Host: www.comp.nus.edu.sg\r\n
User-Agent: Mozilla/5.0\r\n
Connection: close\r\n
```

- Response:

```
HTTP/1.1 200 OK\r\n
Date: Wed, 01 Jul 201508:47:52 GMT\r\n
Connection: Keep-Alive\r\n
Content-Length: 73\r\n
Content-Type: text/html\r\n
Keep-Alive: timeout=5, max=100\r\n
\r\n
<!DOCTYPE html>...
```

- Status Codes: 200 OK, 301 Moved Permanently, 304 Not Modified, 403 Forbidden, 404 Not Found, 500 Internal Server Error
- HTTP is stateless, and uses cookies in messages to maintain state
- Caching: Don't send object if client has up-to-date version of resource, check using If-modified-since header which can give 304

Domain Name System (DNS)

- Translates between hostname and IP address
- Stored as resource records with different types (A = address, NS = nameserver, CNAME = canonical name, MX = mail exchange)
- Use nslookup or dig to find information
- Stored in distributed hierarchical databases, 13 root nameservers worldwide
- Top-level domain (TLD) servers: Responsible for domain suffixes and country domains
- Authoritative servers: Organisation's own DNS servers, provides mappings for oganisation's named hosts
- Local DNS Server: Does not belong to hierarchy, each ISP has one
- Recursive Query: DFS style query
- Iterative Query: BFS style, requests all come from local DNS server
- DNS Caching: Mapping is cached once nameserver leans about it, have TTL
- Runs over UDP/53

Sockets / Transport Layer

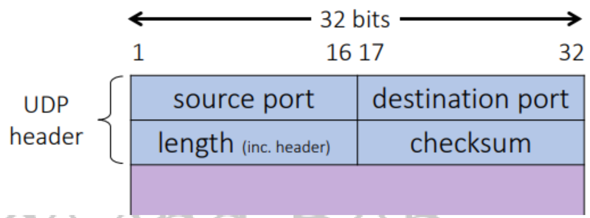
- Host runs applicaitons in multiple processes, which are top-level execution containers with independent memory space
- Threads run in a process and share the same memory
- Identified by IP Address and Port Number
- Sockets are the abstraction interface between processes in the application layer and transport layer protocols
- Datagram Socket:
  - Uses UDP, only one socket is needed
  - Each application creates a packet with recipient and OS attached return information
- Stream Socket:
  - Uses TCP, where connection is established between two processes
  - Data flows in continuous streams, separated into client and server, does not need to attach address information
  - Server creates welcome socket, and forks a new socket when contacted by a client
  - Client creates a socket to establish connection with server, and each connection has its own socket instance

Reliable Protocols / Delivery Transfer

- Transport layer resides on end hosts and provides process-process communication
- Network layer provides host-host, best-effort, and unreliable communication
- Unreliable channel might corrupt / drop / re-order / delay packets
- rdt 1.0: Reliable channel, no special handling needed
- rdt2.0: Channel with bit errors, use checksum to detect
- Requires ACK and NAK as well as retransmissions, but fails if acknowledgment gets corrupted
- rdt2.1: Add sequence number to handle duplicates, retransmits if acknowledgment is corrupted
- rdt2.2: Replace NAK with ACK of last correctly received packet
- rdt3.0: Packet can be lost, corrupted, or delayed, retransmits ACK or packet on timeout
- Utilisation: Fraction of time link is actually being used, $U = \frac{time\ sending}{total\ time} = \frac{d_{trans}}{d_{trans}+RTT}$
- Throughput: $\frac{L}{RTT+d_{trans}}$
- Pipelining allows multiple packets to be transmitted at once, requires buffering, increases utilisation
- Go-Back-N:
  - Cumulative ACK, ACK n means all packets $\leq n$ have been received
  - Keeps track of n unACKed packets, with timer for oldest one
  - On timeout, retransmit all packets, receiver ignores out of order packets
- Selective Repeat:
  - Each packet has a timer, and is retransmitted on timeout
  - Receiver individually acknowledges correctly received packets, buffering out of order packets
  - Has overhead from maintaining timers

User Datagram Protocol (UDP)

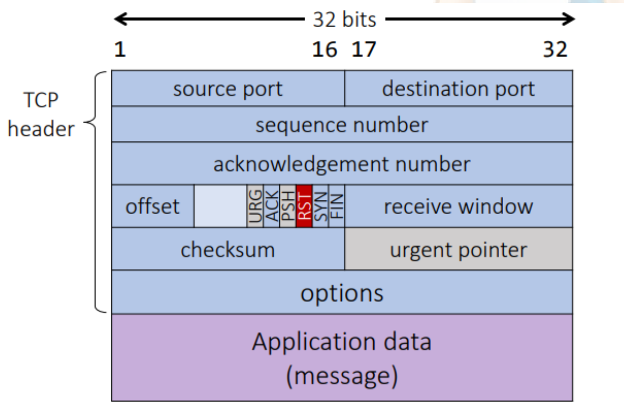
- Adds very little on top of IP
- Unreliable tranmission, requires RDT, no flow or congestion control
- No connection setup or state needed, faster and less resources needed
- Less overhead due to small header size, no congestion control
- Multiplexing: Allow multiple sockets to send data on the same transmission channel to the same socket
- Transport layer de-multiplexes using destination port number, directs it to correct UDP socket
- Checksum: 16-bits used to identify single bit flips
- Split segment into 16-bit integers, add with wrap around carry, then compute 1s complement
- At receiver, perform the same addition with checksum which should result with all bits set



Transmission Control Protocol (TCP)

- Connection-oriented, handshake must be established
- Reliable, in-order byte stream, segments have Maximum Segment Size (MSS) not including header
- Has flow control and congestion control

- Socket is identified by source and destination IP address and port
- Guaranteed delivery, but no guarantee on throughput or reliability
- Sequence Number: Byte number of first byte of data in segment
- Acknowledgment Number: Sequence number of next byte of data expected, cumulative ACK
- TCP Timeout Value: Too short causes retransmissions, too long causes slow reaction to loss
- Estimate RTT: Take Sample RTT and use it to calculate Estimated RTT
- $RTT_E = (1-\alpha)RTT_E + \alpha RTT_S$, typically $\alpha = \frac{1}{8}$, exponential weighted moving average
- Set retransmission timeout based on deviation of RTT and safety margin
- $RTT_{dev} = (1-\beta)RTT_{dev} + \beta|RTT_S - RTT_E|$, typically $\beta = \frac{1}{4}$
- $RTO = RTT_E + 4RTT_{dev}$
- Fast Retransmission: Resend immediately upon receiving 3 duplicate ACKs
- Connection is established using 3-way handshake: Sender sends TCP SYN and initial sequence number, server chooses initial sequence number and sends TCP SYN/ACK, client sends ACK and data
- Half-Open Connections: Vulnerable to SYN flooding or SYN/ACK flooding DoS
- Each side closes own side of connection, sends segments with FIN bit, can only receive data after sending it
- Flow Control: Receiver buffers data to application, telling sender how much data it can send, sender sends 0-data segment when buffer empties



Network Layer

- Provides communication service between any two hosts in the world
- Each host needs to be addressed
- Path between all pairs of hosts needs to be determined
- Need to define a protocol / service guarantee
- Router: Device that forwards packets between networks

Network Addresses

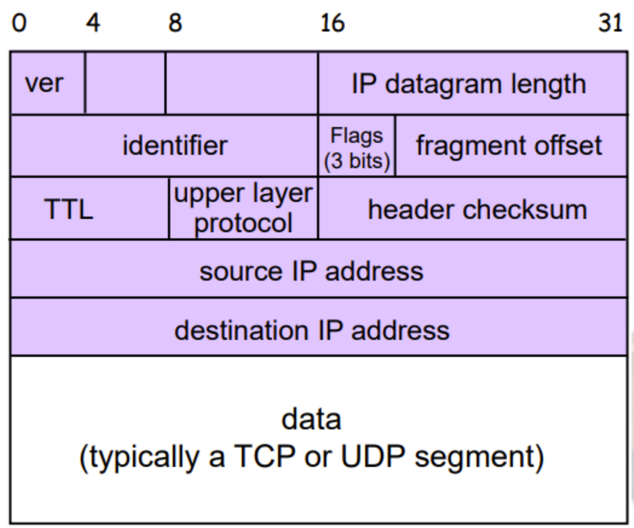
- IP Address: used to identify every interface of host, has to be globally unique, 32 bits
- Routers store forwarding tables with destination IP addresses and output links
- Address Aggregation: Use wildcards to specify range of addresses to reduce size of forwarding table
- Subnet: Network formed by a group of interconnected hosts which can reach each other without a router, single link between 2 routers also counts as a subnet
- IP Address is made up of network / subnet prefix and host ID, given in a.b.c.d/x where x is number of bits in subnet prefix
- Subnet Mask is used to determine which subnet an IP address belongs to
- ISPs own a consecutive block of IP addresses
- There are some special IP addresses, such as localhost, private, and broadcast addresses

Network Address Translation (NAT)

- Public Addresses: Globally unique and routable
- Private IP Addresses: Not globally unique or routable, used within organisations
- WAN: The Internet, LAN: Local network
- All datagrams leaving local network through router have same source NAT IP address
- Within local network, hosts have own private IP addresses
- NAT translation table: Translates WAN side addresses and ports to LAN side addresses and ports
- Router replaces datagram information as necessary
- Easier to change addresses, and hosts within network are not visible to outside world
- Port number is 16-bit, NAT supports 2^{16} addresses

IPv4 Fragmentation

- Datgram: Ver is protocol version number, IP datagram length includes header, TTL is decremented at each hop, 20 bytes total



- Different links have different Maximum Transfer Units (MTU), maximum amount of data a link-level frame can carry
- IP Datagrams that are too large might be fragmented by routers to be reassembled at destination
- Uses identifier, flags, and fragment offset fields in header
- Each fragment shares the same identifier
- Flag is 1 if there is next segment, and 0 at last segment
- Offset is in units of 8 bytes, relative to beginning of original datagram

Dynamic Host Configuration Protocol (DHCP)

- Allocates IP addresses to hosts in network, renewable and reusable
- Runs over UDP, Server Port: 67, Client Port: 68
- Host broadcasts DHCP discover, Server responds with DHCP offer, Host request IP address DHCP request, Server sends address DHCP ACK
- Destination is broadcast address with correct port, original host IP is 0

Routing

- Look for longest prefix match in forwarding table ,and forward to corresponding link
- Routing: Finding least cost path between two vertices in graph, but every node only has information about immediate neighbours
- $C_{x,y}$: cost of link between x and y, $D_x(y)$: least-cost path from x to y
- Bellman-Ford: $D_\alpha(z) = min_{a \in N}\{c(\alpha, a) + D_a(z)\}$, min is taken over all direct neighbours a of α
- Distance Vector Algorithm:
- Each neighbour sends its distance vector to source telling it cost from self to target
- At each time period, all nodes receive the distance vectors from neighbours and compute their new local distance vector before sending it out
- Iterative, asynchronous, distributed, self-stopping when no updates
- Routing Information Protocol implements DV, using hop count as the cost metric, exchanging every 30 seconds over UDP/520
- Self-repair: Assume neighbour has failed after 3 minutes with no update
- Intra-AS routing: Find good path between two routers in same autonomous system, focuses on performance, uses RIP/OSPF
- Inter-AS routing: Handles interfaces between ASs, needs policy, uses BGP

Internet Control Message Protocol (ICMP)

- Used by hosts and routers to communicate network-level information such as error reporting or echo requests and replies
- Messages carried in IP datagrams, starting after IP header
- Header consists of Type + Code (Sub type) + Checksum + Others

Network Layer Services

- Delivers packets to receiving hosts, includes routing, IP, and ICMP
- Data plane: Local, per-router function, determines forwarding within router
- Control plane: Network-wide logic, determines how datagram is routed among routers on path between source and destination hosts
- Routers use longest prefix matching, performed using ternary content addressable memories (TCAMs), content addressable
- Best-effort basis
- Exists in every host and router, router examines header fields in all IP datagrams passing through it

Link Layer

- Send data between n nodes via cable
- Complete graph: Interconnect every pair of nodes, but each link needs to be addressed and not scalable
- Broadcast link: Shared link across multiple nodes, needs protocol, and error handling
- Sends datagrams between adjacent nodes over a single link
- IP datagrams are encapsulated in link-layer frames for transmission, and different links might use different protocols
- Possible services include framing, link access control, error detection and correction, and reliable delivery
- Link layer is implemented in an adapter / network interface card (NIC) on a chip
- Point-Point Link: Sender and receiver connected by dedicated link
- Broadcast Link: Shared medium, every node receives a copy of transmitted frames

Multiple Access Protocols

- If two or more nodes transmit simultaneously, collision occurs
- Ideal protocol: Collision free, efficient, fair, and fully decentralised, coordination must use channel itself

Random Access Protocols

- Nodes transmit at full speed with no prior coordination
- Protocol specifies how to detect collisions and recover from them

ALOHA

- Slotted ALOHA: All frames have equal size, time is divided into slots of equal length of time to transmit 1 frame
- Nodes have synchronised time, transmitting only at beginning of a slot
- Node retransmit frame in each subsequent slot with probability p if there is a failure
- Not collision free, and only efficient when one node is active, perfectly fair, decentralised
- Pure ALOHA: No time slots or synchronisation, nodes transmit frames immediately, waits for 1 frame transmission time before retransmission with probability p
- Not collision free, even less efficient, perfectly fair, decentralised
- Carrier Sense Multiple Access (CSMA): Listen before transmission, take note of other node's activity
- Defer transmission until channel is idle, but can still have collisions due to propagation delay

Carrier Sense Multiple Access (CSMA)

- CSMA/CD (Collision Detection): In CSMA, node does not stop transmission even when collision is detected
- With CD, abort transmission on collision detection and retransmit after random delay
- Adapt retransmission attempts to estimated current load, so probability of collision decreases
- Binary Exponential backoff: After m collision, choose K at random from $\{0, ..., 2^m - 1\}$, with $p = \frac{1}{2^m}$ before waiting K time units for retransmission (1 time unit is 512 bit transmission time for Ethernet)
- If frame size is too small, collision happens but cannot be detected, Ethernet uses 64B
- Can be avoided with $2max(d_{prop}) \leq d_{trans}$, where $max(d_{prop})$ is directly proportional to the diameter of the network, and d_{trans} is directly proportional to frame size
- CSMA(/CD) is not collision free, but is efficient, fair, and decentralised

Taking-Turns Protocols

- Polling: One node is designated as master node, and polls each of the nodes in round-robin fashion, telling them how many frames they can transmit
- Collision free, efficient, fair, but not decentralised with a single point of failure
- Token Passing: Special frame is sequentially passed from one node to next, sequentially
- Node holds on to token if it has frames to transmit, and sends a maximum number of frames before passing it on
- Collision free, efficient, perfectly fair, and decentralised, but token loss is disruptive, and ring can be broken by failure

Channel Partitioning Protocols

- Time Division Multiple Access (TDMA): Access channel in rounds, each node gets fixed length time slots in each round for data transmission
- Collision free, inefficient, perfectly fair and decentralised
- Frequency Division Multiple Access: Channel spectrum is divided into frequency bands, each node is assigned a fixed frequency band, unused transmission time in frequency bands go idle
- Collision free, inefficient, perfectly fair and decentralised

Error Detection and Correction

- EDC: Error detection and correction bits
- Not totally reliable
- Single Bit: In even parity scheme, sender include one additional bit so that total number of 1s is even
- Can detect odd number of single bit errors, does not work well as errors are often clustered together
- 2-Dimensional: bits are divided into i rows and j columns, compute parity bit for each row and column and total parity bit
- Can detect and correct single bit errors, and detect two-bit errors
- Cyclic Redundancy Check: Generate r bit error detection code for d digit number
- Use $r+1$ bit number G, known as generator
- Send data appended with r bit CRC
- Perform calculations modulo 2, same as XOR, repeatedly divide D by G to get R for sender
- Receiver divides sender message by G, should get zero remainder
- Easy to implement, can detect all odd number of single bit errors, CRC of r bits can detect all burst errors of up to r bits, and all burst errors $> r$ bits with $p = 1 - 0.5^r$

Local Area Network (LAN)

Ethernet

- Network that interconnects computers within a geographical area
- Ethernet is the dominant wired LAN technology
- Ethernet Frame:

8 bytes	6	6	2	46 - 1500	4
Preamble	Dest Addr	Src Addr	Type	Data	CRC

- Address are in Media Access Control (MAC) address
- If frame matches destination address or broadcast address, NIC passes it to network layer protocol, if not it discards it
- Size ranges from 46-1500 bytes, in line with MTU and minimum frame size
- Type Indiates higher layer protocol, allowing Ethernet to multiplex them
- Preamble starts with $AAAB$ in hex, and is used to synchronise receiver and sender clock rates, important if there is a long string of bits with the same value
- Ethernet is unreliable, and uses CSMA/CD
- Bus Topology: Broadcast LAN, all transmitted frames are received by all adapters connected to the bus, but single point of failure and slow
- Star Topology:
- Hub: Nodes are connected to hub, a physical-layer device that acts on individual bits rather than frames, cheaper but slow due to collisions

Link Layer Switches

- Switch: Nodes are directly connected to a switch, which is a layer-2 device that works on frames rather than bits, store-and-forward, with no collisions
- Uses CSMA/CD to access link, is transparent, and plug-and-play (does not require configuration)
- Examines incoming frames MAC address and selectively forward to one or more outgoing links
- Nodes have dedicated direct connection to switch, which buffers packets
- Switch has a switch table which maps MAC address of host to interface to reach host, stored with TTL
- Switch learns which hosts can be reached when receiving a frame from sender, recording it in switch table
- When a frame is received:
  - Record the incoming link and MAC address of sending host
  - Index switch table using MAC destination address
  - If entry is found, forward it to interface indicated only if destination on segment is different from which frame arrived
  - If entry is not found, broadcast frame to all interfaces except arriving interface

Link Layer Addressing and ARP

- MAC Address: Every adapter has one, adapter uses it to check destination MAC address of frame and filters if necessary
- Typically 48 bits, burned in the Read-Only memory of an NIC, written in hexadecimal pairs, broadcast address is all 1s
- Each IP node has an Address Resolution Protocol (ARP) table, sorting mapping between IP address, MAC address, and TTL
- Sending within same subnet:
  - If source knows destination MAC address from ARP table, create a frame with it and simply send it
  - If not, broadcast an ARP query packet with destination IP address
  - Only destination will reply with its MAC address, sent to source
  - Source caches destination IP and MAC address mapping in ARP table
- Sending to another subnet:
  - Source creates IP datagram with source and destination IP addresses
  - Source creates link layer datagram with router's MAC address as destination address, frame contains IP datagram
  - Router removes link layer frame, and passes it to IP layer
  - Router forwards datagram with IP addresses to receiving router
  - Receiving router creates link layer frame with destination MAC address, containing original IP datagram, forwarding to destination

Network Security

- Intruders or eavesdroppers might edit messages
- Listen, delete / modify, add messages / impersonate
- Repudiation: Proving that transaction did not happen between two entities
- Confidentiality: Only sender and intended receiver should understand message contents
- Message Integrity: Sender and receiver want to ensure message is not altered without detection
- Authentication: Sender and receiver want to confirm identity of each other

Confidentiality

- Cryptography: Allow a sender to disguise data so that intruder cannot gain information from it, while allowing receiver to recover original data from disguised data
- m: plaintext messsage
- $K_A(.)$: encryption algorithm with key K_A, $K_A(m)$: ciphertext
- $K_B(.)$: decryption algorithm with key K_B, $K_B(K_A(m)) = m$
- Algorithms and keys are agreed upon beforehand, which can be symmetric or asymmetric
- Casesar Cipher: A substitution cipher where one thing is substituted for another, fixed shift of alphabet, only 25 possible keys
- Monoalphabetic Cipher: Substitute one letter for another, $26!$ mappings possible, but susceptible to dictionary attack or statistical analysis
- Attacks: Ciphertext only, known-plaintext (has plaintext corresponding to ciphertext), chosen-plaintext (can get ciphertext for chosen plaintext)
- Polyalphabetic Encryption: Compose multiple mappings on each other in a cyclic pattern for each character
- Block Cipher: Encrypt message in blocks of K bits, use a one-to-one mapping to encode a block, $2^K!$ keys
- Data Encryption Standard: 56-bit symmetric key, 64-bit block, cracked in less than a day

- Advanced Encryption Standard: 128-bit blocks, with 128/192/256 bit keys, takes very very long to break
- Need many symmetric keys, one for each pair of individuals

RSA

- Public Key Cryptography: Sender uses public encryption key known to all, receiver uses a private decryption key only known to them
- Requirements: Need $K_B^+(.)$ and $K_B^-(.)$ such that $m = K_B^-(K_B^+(m))$, and impossible to compute K_B^- from K_B^+
- RSA Encryption:
  - Choose two large prime p, q
  - Compute $n = pq, z = (p-1)(q-1)$
  - Choose $e < n$ which is relatively prime with z
  - Choose d such that $ed \ mod \ z = 1$
  - The public key is (n, e) while the private key is (n, d)
  - To encrypt, compute $c = m^e \ mod \ n$, to decrypt compute $c^d \ mod \ n = m$, works as $(m^e \ mod \ n)^d \ mod \ n = m^{ed} \ mod \ n = m$ and Fermat's Little Theorem $a^p \equiv x \ mod \ p$
- RSA is computationally intensive, compared to DES which needs K_S
- Use RSA to transfer K_S, before using K_S as symmetric key for DES this session, known as session key

Message Integrity

- Hash Function: $H(.)$ takes in an input m and produces a fingerprint $H(m)$ with fixed length
- Internet Checksum: Produces 16-bit fingerprint, has collision, used to identify accidental errors rather than attacks
- CRC is better, but still poor, and biased to input (minor changes in input produce minor changes in output)
- Cryptographic Hash function: Hash function where it is computationally infeasible to find differing x, y such that $H(x) = H(y)$ so that it is hard for intruders to substitute
- e.g. MD5 (128-bit), SHA-1 (160-bit), both have been broken
- Cannot send $(m, H(m))$ as attacker can replace it
- Message Authentication Code: Send $(m, H(m+s))$ where s is a secret key only known to receiver and sender
- Passwords are hashed and stored, checked with equality of hashes, cannot be recovered
- Other uses include checking software integrity, timestamping as proof of work, and data integrity
- Hashing is one-way, fast, and is not random

Authentication

- Digital Signatures: Cryptographic technique similar to hand-written signatures
- Must be verifiable and unforgeable
- Useful RSA property: $K_B^-(K_B^+(m)) = m = K_B^+(K_B^-(m))$
- Simple Digital Signature: Bob signs m by encrypting with private key, Alice can verify by applying his public key
- Computationally expensive to public-key-encrypt long messages, so consider signing message digest instead, different from MAC
- Weakness of public key: Impostor can pass our public key and claim to be someone else, so public key needs to be shared securely
- Distribution Methods: Public announcement on website or in avilable directory, Public Key Infrastructure (PKI)
- PKI: Consists of Certificate and Certificate Authority (CA)
- Certificate: Digital document that contains minimally identity of owner, public key of owner, time window of validity, and CA signature
- CA: Issues and signs digital certificate to websites, maintains a directory of public keys, has won public-private key pair as well, assumed to be securely distributed to all entities involved
- CA is a bottleneck as verifiers need access to directory server of CA
- With CA, can distribute public key with certification from CA using $K_{CA}^-(H(K_B^+))$, however K_{CA}^+ also needs to be securely shared
- CA needs to be created to certify other CAs, keep a list of trusted CAs, such as Trusted Root Certification Authorities
- CA binds public key to an entity E with certificate, E provides proof of identity
- Firewall: Isolates organisation's internal net from larger net, choosing which packets can pass through
- Prevent DoS, illegal access of internal data, and only authorised users
- Stateless packet filters, stateful packet filters, application gateways
- Stateless packet filter: Router filters packet by packet, deciding whether to drop based on header fields search as source and destination IP address or port number, ICMP message type, TCP SYN and ACK
- Access Control Lists (ACL): Table of rules, applied top to bottom on incoming packets, action condition pairs
- Firewalls cannot detect IP spoofing, and could be a bottleneck
- Secure e-mail: Generate symmetric key, encrypt message and key with recipient public key, who can then use it to retrieve message
- Authentication, Message Integrity: Sender digitally signs message, sends both message in clear and digital signature
- Secrecy, Authentication, Message Integrity: Sender users own private key, recipient public key, newly created symmetric key

Tutorial Content

Message Segmentation

- Without message segmentation, the whole packet must be retransmitted if there are bit errors that cannot be tolerated
- Without message segmentation, huge packets are sent into the network which routers have to accommodate for, and smaller packets have to queue behind
- However, packets must be put back in sequence at the destination
- Many smaller packets must also carry their own headers which causes some overhead

Topology

- Minimum links: Simpler and cheaper, but has many points of failure that could crippler network, along with having longer paths between nodes
- Maximum links: More robust and faster travel, but is expensive

DNS

- Suppose n DNS servers are visited each with RTT of D_{DNS}
- Let D_{Web} denote RTT between local host and server of each object
- For five objects and three DNS servers:
- Non-persistent HTTP with no parallel TCP connections: $3D_{DNS} + (5+1) \times 2 \times D_{Web}$
- Non-persistent HTTP with parallel TCP connections: $3D_{DNS} + 2 \times D_{Web} + 2 \times D_{Web}$, as the HTML file must be first fetched before which the 5 objects can be fetched in parallel
- Persistent HTTP with pipelining: $3D_{DNS} + 2 \times D_{Web} + D_{Web}$, as the HTML needs to be fetched first before each of the 5 objects can be fetched in parallel over the same connection
- DNS Cache Poisoning: Rogue DNS records are introduced into DNS resolver's cache, causing name server to return an incorrect IP address and divert traffic to the attacker

Sequence Numbers

- Large sequence numbers are used to prevent collisions
- TTL is specified in IP packet header to prevent packets from circulating
- Increases by number of bytes sent, not with segments sent

Encryption

- Suppose Alice wants to send encrypted mail to Bob by following these steps:
- Generates a random session key K_S
- Encrypts the session key with Bob's public key K_B^+ to get $K_B^+(K_S)$
- Hashes the message m with hash function H to get message digest $H(m)$
- Encrypts hash with Alice's private key K_A^-, obtaining digital signature $K_A^-(H(m))$
- Encrypts message m concatenated with $K_A^-(H(m))$ using K_S to get $K_S(m \oplus K_A^-(H(m)))$
- Transmits $K_S(m \oplus K_A^-(H(m))) \oplus K_B^+(K_S)$ to Bob
- Then Bob should:
- Use $K_B^-(K_B^+(K_S)) = K_S$ to recover the seesion key
- Decrypt the message with K_S to get $K_S(K_S(m \oplus K_A^-(H(m))))$, retrieving m and $K_A^-(H(m))$
- Use Alice's public key K_A^+ to recover $K_A^+(K_A^-(H(m))) = H(m)$
- With m, compute $H(m)$ and verify that it is correct
- This ensures confidentiality, integrity, and authenticity

Other Stuff

- Remember link layer MTU also includes IP header
- AES and 128-bit key: Maintaining large table is computationally expensive, so block ciphers typically use functions that simulate randomly permuted tables