CS2105

AY24/25 Sem 2

by ngmh

Internet

- Network of connected computing devices
- Internet is NOT World Wide Web
- World Wide Web is one of the services that runs over the Internet

Network Edge

- Consists of end hosts, servers, etc.
- Hosts run network applications
- Applications communicate using protocols, which define format and order of messages as well as actions to take
- Hosts access the internet through access networks e.g. Residential, Institutional, Mobile
- Wireless Access Network: Connects hosts to router via base station / access point
  - Wireless LANs: Wi-Fi, Within building
  - Wide-area wireless access: 3G, 4G, Provided by telco (cellular) operator
- Physical Media: How hosts connect to access network
  - Guided: Solid media, e.g. Fiber
  - Unguided: Free propagation, e.g. Wi-Fi, cellular

Network Core

- Consists of ISPs, routers, etc.
- Mesh of interconnected routers
- Circuit Switching
  - Resources allocated and reserved for transmission
  - Setup required, but guaranteed performance
  - Circuit segment is idle if not used by call (no sharing)
  - Common in traditional telephone networks
  - Divide bandwidth into pieces by frequency and time
- Packet Switching (e.g. The Internet)
  - Host breaks message down into packets of length L and transmits on a link with transmission rate R (a.k.a bandwidth)
  - Packet transmission delay $= L/R$
  - Packets are passed along from one router to the next across links
  - Store and Forward: Entire packet must arrive at a router before being transmitted out
  - Routing and Addressing: Routers determine route taken by packets, which need to carry source and destination information
  - No setup needed, but resources are shared and transmission is best effort
- Connecting ISPs to each other is costly, so there are regional and global ISPs which act as middlemen
- Peering Link: Between 2 ISPs
- Internet Exchange Point: Middleman between 2 ISPs
- Content providers might also run their own network
- Authorities
  - IP Address and Internet Naming administered by Network Information Centre (NIC)
  - The Internet Society (ISOC): Leadership in Internet related standards
  - The Internet Architecture Board (IAB): Authority to issue and update technical standards regarding Internet protocols
  - Internet Engineering Task Force (IETF): Protocol engineering, development, and standardisation

Delay and Loss

- Packets queue in router buffers to be sent out
- When capacity is reached, arriving packet will be dropped and lost
- Lost packet may be retransmitted by previous node or source or not at all
- Processing Delay: Time to check bit errors, and determine output link
- Queueing Delay: Time waiting in queue for transmission, depends on congestion level
- Transmission Delay: L/R, time taken for packet to be fully sent out
- Propagation Delay: d/s, time taken for packet to travel across link
- These 4 factors form end-to-end packet delay
- Throughput: How many bits can be transmitted per unit time
- Link Capacity / Bandwidth is measured for a specific link
- Units: 1 byte = 8 bits, Micro, Milli, Standard, Kilo, Mega, Giga, Tera
- Capital B = bytes, Small B = bits

Protocol Layers

- Modularise large and complex systems
- Each layer servers their own purpose, with simple interfaces between them
- Layers:
  - Application: Treat Internet as a black box
  - Transport: Process-to-process data transfer
  - Network: Routing of datagrams from host to host
  - Link: Data transfer between neighbouring network elements
  - Physical: Bits on the wire / air

Application Layer

Architecture

- Client-Server Architecture
  - Server waits for incoming requests and provides requested service to client
  - Client initiates contact with server and requests service from it
- P2P Architecture
  - No always on server, end systems directly communicate
  - Peers request and provide service to other peers
  - Highly scalable, but difficult to manage
- Hybrid of both with centralised server and P2P, e.g. isntant messaging

Service Criteria

- Data Integrity: Whether they need reliable data transfer (e.g. file transfer) or can they tolerate data loss (e.g. audio streaming)
- Throughput: How much bandwidth does the app need (e.g. multimedia)
- Timing: Some apps require low delay to be effective (e.g. online games)
- Security: Encryption and data integrity

Protocols

- They define:
  - Types of messages exchanged
  - Rules for when to send and respond messages
  - Message Syntax: What fields are in messages and how are they delimited
  - Message Semantics: Meaning of information in fields
- Open protocols: Defined in RFCs and allow for interoperability
- Proprietary: Privatized and not released to public
- Identifying Network Processes:
  - IP Address (Globally Unique): Identifies Host, IPv4 has 32 bits split into 4 bytes in dotted decimal notation, v6 has 128 bits split into 8 sets of 2 bytes in hexadecimal notation
  - Port Number (Locally Unique): 16 bit number, 1 to 1023 are reserved
  - Port numbers are assigned by IANA
- User Datagram Protcol and Transmission Control Protocol

The Web

- WWW Consortium: Global Open standards
  - HyperText Markup Language (HTML): Explains how it should be interpreted
  - Uniform Resource Locator (URL): Explains how to locate resources
  - HyperText Transfer Protocol (HTTP): Explains how to access those resources
- Webpage: Base HTML file and other referenced objects (which each have their own URL)
- Resource are requested using HTTP which uses a client / server model

HTTP

- Uses TCP as a transport service
- Round-Trip Time (RTT): Time taken for a packet to travel from client to server and back
- HTTP/1.0: New connection is established for every resource requested, time taken is $2RTT+$ transmission time
- HTTP/1.1 Pipelining: New request is made before receiving response of old requests, as soon as a referenced object is encountered
- Persistent HTTP: Server leaves connection open after sending response, connection is reused, can be as little as one RTT
- 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
- Uses Cookies to keep track of state, server does not store info about connection
- Caching: Don't send object if client has up-to-date version of resource
- Use Conditional Get to determine, with If-modified-since: <date> header in request, which might get a 304 response with no object if up-to-date

DNS

- Domain Name System
- Translate between hostname and IP address
- Stored as Resource Records with different types (A = address, NS = nameserver, CNAME = canonical name, MX = mail exchange) and TTL
- Stored in distributed hierarchical Databases
- 13 Root name servers worldwide
- Top Level domain servers for each domain suffix
- Authoritative servers: DNS Servers belonging to organisations, provides authoritative hostname to IP mappings for organisation's named hosts
- Local DNS Server: Does not strictly belong to hierarchy, each ISP has one
- Recursive Query: Query is forwarded upwards until it is found
- Iterative Query: Query is made separately to each level up of the chain from local DNS server
- DNS Caching: Mapping is cached once name server learns about it

Socket Programming

- A host runs several processes: Top level execution containers with independent memory space

- Each process can have multiple threads which share the same memory
- Processes in the same host communicate with inter-process communication defined by OS while those in different hosts communicate by exchanging messages according to protocols
- Processes are identified by IP Address + Port Number

Sockets

- Sockets are the abstraction interface between processes and transport layer protocols
- Stream Socket uses TCP, while Datagram Socket uses UDP
- UDP Socket:
  - Only one socket is needed, communications is shared
  - Application creates packet with recipient, OS attaches return info
  - Receiver identifies sender by extracting address information
- TCP Socket:
  - Data flows in a continuous stream
  - Connection established between 2 processes
  - Server listens on socket and forks a new one when contacted by client, distinguishing by client address info
  - Client creates socket to establish a connection with server
  - Every connection has its own socket instance

Reliable Protocols

- Transport layer provides process-to-process communication
- Network layer provides host-to-host, best-effort and unreliable communication
- Unreliable channel might corrupt / drop / re-order / delay packets
- RDT: Reliable Delivery Transfer
- Finite State Machine: Describe sender and receiver of a protocol
- rdt1.0: Reliable channel
- rdt2.0: Channel with bit errors
  Use checksum to detect, requires ACK / NAK and retransmissions, but this could be fatal if ACK is corrupted
- rdt2.1: Add sequence number to handle duplicates, retransmits if ACK / NAK is garbled
- rdt2.2: Replace NAK with ACK of last correctly received packet
- rdt3.0: May flip bits, as well as lose or delay packets
  Sender retransmits if ACK reaches timeout, duplicates handled with sequence number

Pipeling

- Utilisation: Fraction of time link is actually being used, $U = \frac{d_{trans}}{d_{trans}+RTT}$
- Throughput: $\frac{L}{d_{trans}+RTT}$
- Pipelining: Allows multiple packets to be transferred at once, requires buffering

Go-Back-N

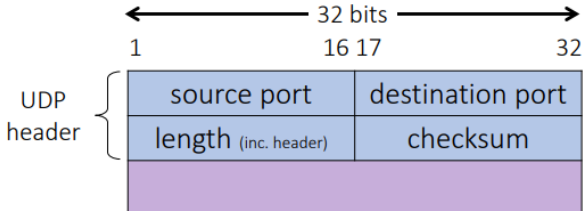
- Cumulative ACK: ACK n means all packets $\leq n$ have been received
- Keeps track of n unACKed packets, with a timer for oldest packet
- On timeout, all packets are retransmitted
- Receiver discards out of order packets

Selective Repeat

- Each packet has a timer, and sender retransmits individual packets on timeout
- Receiver individually acknowledges all correctly received packets, buffering out of order packets
- Maintaining timers could be costlier and less efficient

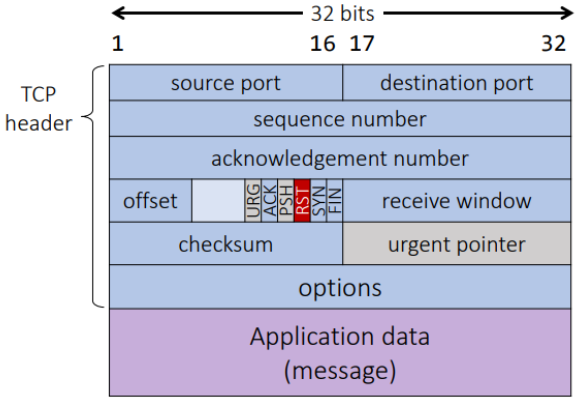
User Datagram Protocol

- Adds very little on top of IP
- Downsides:
  - Unreliable transmissions, users are loss tolerant and rate sensitive
  - No flow or congestion control
- Benefits:
  - No connection setup needed, reduces delay
  - No connection state to be maintained, reduces resources neede
  - Small header size, less overhead
  - No congestion control, can blast as fast as desired
- Unreliable transmissions, users are loss tolerant and rate sensitive
- Multiplexing: Electronic process that allows more than one electrical signal to be sent using only one connection
- Transport Layer Multiplexing: Data from multiple sockets can be sent using one transmission channel to the same socket
- Checksum: Identify single bit flips during transmission
  - Split segment into 16-bit integers
  - Add integers with wrap around carry
  - Compute 1's complement by flipping all bits
  - At receiver, perform the same addition to get the result with all bits set



Transmission Control Protocol

- Benefits:
  - Connection-oriented, handshake must be established
  - Reliable, in-order byte stream, segments have Maximum Segment Size (MSS) which does NOT include header
  - Flow control: Sender cannot flood receiver
  - Congestion control: Throttle sender if network is overloaded
  - Guaranteed delivery
  - No guarantee on throughput
  - More resources needed and increased delay
- 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
  - Estimating RTT: $RTT_E = (1-a) \cdot RTT_E + a \cdot RTT_S$, typically $a = 1/8$, exponential weighted moving average
  - Deviation of RTT: $RTT_{dev} = (1-b) \cdot RTT_{dev} + b \cdot |RTT_S - RTT_E|$, typically $b = 1/4$
  - Retransmission Time Out: $RTO = RTT_E + 4 \times RTT_{dev}$ using estimate and safety margin
- Fast Retransmission: If 3 duplicate ACKs are received, resend segment immediately
- Connection Establishment: 3-way handshake
  - Client sends TCP SYN and initial sequence number
  - Server chooses initial sequence number, sends TCP SYN and ACK
  - Client sends ACK
- Half-Open Connections: Vulnerable to SYN Flooding or SYN/ACK flooding
- Closing Connection: Each side closes their own side of connection, by sending segments with FIN bit, after which they can no longer send data
- Flow Control: Receiver buffers data to application, and tells sender how much data it can send, sender can send 0-data segment to check when buffer empties



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