

The Internet is an example of a **computer network**

Link • Communication links for transmission

Host/Endpoint • Computer running applications of end user

Router • Computer for routing packets from input link to another output link

Network • A group of hosts, links, routers capable of sending packets among its members, Collection of interconnected machines

Packet • A unit of data transmission (ex: 1500 bytes)

We use the term **switching** to denote physically moving data from one link to another

• **Circuit Switching: (reserve resources, needs admission control)**

1. Setup: Control message sets up a path from origin to destination 2. Accept signal informs source that data transmission may proceed 3. Data transmission begins 4. Entire path remains allocated to the transmission (whether used or not) 5. When transmission is complete, source releases circuit

• **Message Switching**

- Header: metadata about message
- Message hops from node to node while allocating only one link at a time while allocating only one link at a time
- Analogy: Postal Service
- When the entire message arrives at a router it is sent to the next router
- If the next router is busy it is held in a queue

• **Packet Switching: (does not reserve resources)**

- Messages are split into smaller pieces called packets
- Pipelining means multiple parts of the message are sent concurrently. Provide higher utilization of link resource

• **Store and forward switching: (Internet uses it)**

- Router waits for all message bits to arrive before sending the first bit to outgoing link

• **Cut-through switching:** sends bits as they arrive

Comparisons:

- Header overhead (how many bits are in metadata) **Circuit < Message < Packet**
- Total Delay
 - Short Bursty Messages: **Packet < Circuit**
 - Long Continuous Messages: **Circuit < Packet**

Measuring Networks

- Packet size: length of packet in bits or bytes, including the header
- Bandwidth: For a single link, the amount of data it can transmit per unit type. **The number of boxes put on belt per minute**
- Propagation delay: The time needed to move one bit across. **The time takes for first box to travel the length of belt**
 - Imposed by communication medium (link length)
- Transmission delay: Time from the first bit leaving the sender to the last bit leaving the sender: **packet size / bandwidth rate(transmission rates)**
 - Determines by link bandwidth and packet size
- Queueing delay: Time the packet is waiting for transmission
- Total packet delay: Time from first bit from sender to last bit received at receiver
 - propagation delay + queueing delay + transmission delay
- Shipment transmission time = N / rate
 - The next box is put on the belt ($1/\text{rate}$) minutes after the last
- Total transfer time = transmission time + propagation delay

Layering

- Software and hardware for networking are arranged in layers which provides **modularity**
- Each layer has **well-defined function and interacts** with outer layers through interfaces
- Functionality is implemented in **protocols**
 - Message format: structure of messages exchanged with an endpoint
 - Actions: operation upon receiving, or not receiving, messages
- Layers include:
 - Application layer: useful user-level functions
 - Transport layer: provides guarantees to apps
 - Network: best effort global packet delivery
 - Link: best effort local packet delivery
- Generally endpoints have all four layers while routers only have network and link layers

Application "addresses"

- IPv4 has 32 bits, IPv6 has 128 bits
- We also need an identity for the application at the endpoint: **port number**
- A **socket** represents the connection between the OS/network and the application process. Think of it as an API to the network
- A app-layer connection is a 4-tuple: (IPA, portA, IPB, portB)

Client-server Architecture

- Server: an always on host that has a permanent IP address, provides service to the world. Compute clusters to scale to many users
- Client: communicates with the server, has a dynamic IP address and may be intermittently connected. Not communicate directly with other clients

P2P (Peer to Peer)

- Peers: intermittently connected hosts that directly talk to each other. There is little to no reliance on always-up servers. Many applications use a **hybrid** model

DNS

- We use alphanumeric names to refer to hosts and map them to IP addresses
- This process is called Address Resolution
- Use distributed and hierarchical databases to enable scaling. Root, TLD, Authoritative

DNS protocol

- Client & Server model over Port 53 on server. DNS server of IP address already known
- Two types of messages. **Queries** (Standard 0x0), (Updates 0x5), **Responses** have the same message format: QR = 0 for query, 1 for response.

Query Types

- Iterative query & Recursive query

DNS records

- Type=A: name is hostname, value is IP address
- Type=AAAA: name is hostname, value is IPv6 address
- Type=NS: name is the domain name, value is the hostname of the authoritative name server for the domain
- Type=CNAME: name is the alias for the canonical name, value is the canonical name
- Type=MX: value is associated mail server

Caching and updates

- Once a name server learns the name to IP address mapping it caches the mapping
- Entries will time out after a period of time
- Root servers aren't visited that much due to caching
- caches are **closer** to users than the origin servers and has **higher bandwidth**

Web and HTTP

- Follows the client server model. Client requests web objects. Server servers
- Each object is addressable by a uniform resource locator (URL). Can also name an application **process** on the server
- **HTTP method type:**
 - GET: Input is uploaded in URL field of request line. Return the requested resource in the entity body of the response along with response headers
 - POST: Input is uploaded to server in entity body. The URL of the request identifies the resource that **processes** the entity body
 - HEAD: Return all the response headers in the GET response, but without the resource in the entity body
 - PUT: PUT: the URL of the request identifies the resource that is **contained in** the entity body
 - DELETE

Cookies

- HTTP is stateless by default (the server doesn't remember anything)
- Collaboration between client and server to track user state. Rely on four components
 - i. Cookie header line of HTTP **response** message
 - ii. Cookie header line in HTTP **request** message
 - iii. The cookie file is kept on the user endpoint, managed by their browser
 - iv. The back-end DB will map the cookie to user data at the web endpoint

Web caches

- Caches reduce response time for client requests & Caches reduce traffic on an access link
- Caches can be implemented in the form of a proxy server
 - Configure an HTTP proxy on computer's network settings
 - Browser will send all HTTP requests to proxy (cache)
 - On hit, the cache will return the object
 - On miss, the cache will fetch the object, store it, and return it
- **Conditional GET** guarantees cache content is up-to-date while still saves traffic and response time whenever possible
- Date in the cache's request is the last time the server provided in its response header **Last-Modified**

CDN (Content Distribution Networks)

- CDNs are a global network of web caches. Provisioned by ISPs or content providers
 - Reduce bandwidth requirements on content providers. Reduce cost of maintaining origin servers. Reduce traffic on a network's internet connection. Improves response time for user requests
- CDN servers are spread out geographically to help reduce propagation delays
- **CDN uses DNS**, it runs its own DNS server (CDN name servers)
 - DNS provides an additional layer of indirection • Instead of returning IP address, return another DNS server (NS record)
 - The second DNS server (run by the CDN) returns IP address to client

SMTP

1. User agents
 - Client facing app that reads the mail
2. Mail Servers
 - A mailbox contains incoming messages for a user
 - A message queue has outgoing mail messages from a user
 - The sender mail server makes a connection to the receiver mail server at their IP address and port 25
3. SMTP Protocol: client/server protocol

- Used to send email messages
- Client talks to user agents and mail server
- Server receives mail messages

SMTP: deals with delivery/storage between mail servers. Focuses on push

Mail access protocols: Pull from mail servers

- POP: Post Office Protocol (extremely simple)
- IMAP: Internet Mail Access Protocol (more features)
- HTTP: Used by web-based emails

Video Streaming

- **Traditional applications** (HTTP(S), SMTP) • Delay tolerant but not loss tolerant • Data used after transfer complete
- **Multimedia applications** are often **real time**
- Quantize
 - How many levels or bits to represent each sample
 - More levels: more accurate representation of signal, more bits to store & need more bandwidth to transmit
- Digital image: array of **pixels**. **Resolution:** number of pixels
- Coding: use redundancy within and between images to decrease # bits. Algorithm: **codec**
 - Spatial: send two values: color and number of repeated values (N)
 - Temporal: send only differences from frame i to $i + 1$
- Video **bit rate:** effective number of bits per second of the video after encoding
 - **bit rate** = number of samples * number of bits per sample. (bits/sec)
 - **$\log_2(\text{level}) = \text{bits per sample}$**
 - Higher bit rate == better to perceive
 - CBR: (constant bit rate): fixed bit-rate video
 - VBR: (variable bit rate): different parts of the video have different bit rates
- **On-demand streamed** video/audio
 - Can begin playout before downloading the entire file
- **Conversational** voice or video over IP
 - interactive human-to-human communication limits delay tolerance
- **Live streamed** audio, video
- Continuous playout constraint
 - once video playout begins at client, time gap between frames must match the original time gap in the video
 - Clients have a **client-side buffer** of downloaded video to absorb variation in network conditions
 - **Playout delay:** if too small, nothing in the buffer to show, cause stalls
- **Initial playout delay tradeoff:** buffer starvation less likely with larger delay, but also incur a larger delay until the user begins watching
- **Adapt bit rate per segment** through collaboration between the video client (e.g., your browser) and the server
- **Adaptive bit-rate (ABR) video:** change the bit-rate (quality) of next video segment based on network and client conditions

Dynamic Adaptive Streaming over HTTP (DASH)

- Adaptive: Perform video bit rate adaptation
- Dynamic: Retrieve a single video from multiple sources
- Content **divided** into segments (time)
- Algorithms to determine and request varying attributes for each segments
- Media Presentation Description (**manifest**)
 - Periods: Durations of content
 - Adaptation set: functionally equivalent content
 - Representations: codecs, bit rates, etc
 - Functionally **equivalent:** RSes of given AS. Functionally **different:** different ASes

Transport services and protocols

- Provide a communication abstraction between **application processes**
 - **send side:** transport breaks app messages into **segments**, passes to network layer
 - **recv side:** reassembles segments into messages, passes to app layer
- **Network** layer: abstraction to communicate between **endpoints**. Network layer provides best effort packet delivery to a remote endpoint.
- **Transport** layer: communication abstraction between processes. Delivers packets to the process.
- Application connections are identified by **4-tuple**:
 - Source IP address • Source port • Destination IP address • Destination port
- Transmission Control Protocol (**TCP**)
 - Connection-based: the application remembers the other process talking to it.
 - Suitable for longer-term, contextual data transfers, like HTTP, file transfers, etc.
 - Guarantees: reliability, ordering, congestion control
- User Datagram Protocol (**UDP**)
 - Connectionless: app doesn't remember the last process or source that talked to.
 - Suitable for single req/resp flows, like DNS.
 - Guarantees: basic error detection