CS 118(Computer Networking) Notes

Chapter 1 - Computer Networks and the Internet

Introduction

- Internet is a computer network that interconnects hundreds of millions of computing devices throughout the world.
    - Internet is network of networks, with lots of interconnected ISPs.
- End systems or hosts are connected together by a network of communication links and packet switches.
- Different links transmit at different rates.
- End systems send packets of information to each other.
- 2 most prominent types of packet switches are routers and link-layer switches.
    - Link-layer switches = Used in access networks
    - Routers = Used in the network core. They run protocols in order to move data to their destinations.
- Sequence of communication links and packets switches traversed by a packet is known as a route.
- End systems access the Internet through Internet Service Providers (ISPs)
    - ISP is itself a network of packet switches and communication links.
        - Provides residential broadband access, WiFi, internet access to content providers.
        - Internet is about connecting end systems, and ISPs allow this to happen.
        - ISPs also have to be connected to each other so this results in a network of networks.
- End packets and packet switches run protocols that control the sending and receiving of information within the Internet.
- Transmission Control Protocol (TCP) and Internet Protocol (IP) are the two most important protocols in the Internet.
    - IP specifies the format of the packets that are sent and received.
- Important that everyone agrees on what each protocol entails.
- Internet standards were developed by the Internet Engineering Task Force and their standards documents are called requests for comments (RFCs)
    - They define all of the application level protocols.
- Distributed applications are those that involve multiple end systems that exchange data with each other.
- End systems provide an API that specifies how a program running on one end system asks the Internet infrastructure to deliver data to a specific destination program running on another end system.

- All activity in the Internet involves two or more communicating remote entities governed by a protocol.
- Protocols define the format and order of messages exchanged by two or more communicating entities and the actions taken on message transmission and receipt.
- Hosts = end systems.
    - Within the field of hosts, you have two categories, clients and servers.
- To send a message from one system to another, the source breaks the message up into packets.
    - Packets travel through communication links and packet switches (either routers or link-layer switches)
- Store-and-forward transmission is where the packet switch must receive the entire packet before it can begin to transmit the first bit of the packet onto the outbound link.
- Each packet switch has multiple links attached to it. For each link, the packet switch has an output buffer (stores packets the router is about to send) and could suffer queuing delays where there is possible packet loss.
- In the Internet, every end system has an address called an IP address.
- When the source system wants to send something to the destination, the source includes the destination's IP address in the packet's header.
- Each router then has a forwarding table that maps the destination IP address to the router's different outbound links.
    - There are a special set of routing protocols that automatically set the forwarding tables.
- Moving data through a network of links and switches can be done through circuit switching and packet switching.
    - In circuit switched networks, the resources needed along a path to provide for communication between the host and the end is reserved for the duration of the communication session between the systems.
    - Not reserved in packet switched networks as a session's messages use resources on demand and we'll have to wait for access to a communication link.
- People say packet switching isn't suitable for real time services because of the unpredictable delays. The pro side for packet switching is that there's a better sharing of transmission capacity and an easier implementation.
- Packet switching = going to a restaurant without a reservation but possibly having to wait
- Circuit switching = dealing with making a restaurant reservation.
- A link dedicates a frequency band to each connection for the duration of the connection. The width of the band is called the bandwidth. Bandwidth also refers to the transmission rate of a specific communication link.
- The time required to examine the packet's header and determine where to direct the packet is part of the processing delay.
- At the queue, the packet experiences a queueing delay as it waits to be transmitted onto the link.
- Transmission delay is time required for router to push out the packet or put all of the bits onto the link.

    - Size of the file (in bits) / bandwidth = transmission delay
- The time required from the beginning of the link to router B is the propagation delay.
    - Propagation delay is length of the link divided by the speed of light inside of the link.
- Nodal delay = processing delay + queue delay + transmission delay + propagation delay.
- Instantaneous throughput is the rate at which Host B is receiving some file.
- If the packet finds a full queue, then the router will drop the packet and it will be lost.
- Network designers organize protocols into layers.
- Service model is where a layer provides services to the layer above it.
- The Internet's transport layer transports application-layer messages between application endpoints.
- A passive receiver that records a copy of every packet that flies by is called a packet sniffer.
- The ability to inject packets into the Internet with a false source address is known as IP spoofing.

Chapter 2 - Application Layer

Application Layer

- Network application development is all about writing programs that run on different end systems and communicate with each other over a network.
    - The communication between these applications takes place between end systems at a application layer.
    - Application processes communicate with each other using application protocols.
- Network core devices (routers or link layer switches) function at lower layers (not the application layer).
- You can have two different application architectures when you're designing these network applications.
    - Client-server architecture - There is an always-on host called a server, which services the requests that are made from many other hosts, each called clients.
        - Server has a fixed and well known address, and because server is always on, a client can always contact the server by sending a packet to the server's IP address.
        - Big companies have data centers that hold a bunch of servers to handle all of the requests they get from clients.
        - Servers are reachable by IP address.
        - Clients initiate communication with the server.
    - P2P architecture - Minimal or no reliance on dedicated servers in data centers. Application instead exploits direct communication between pairs of hosts, called peers.

        - Traffic-intensive applications are based on P2P architectures.
        - Peers request service from other peers, provide service in return to other peers.
        - Has self-scalability in that each peer generates workload but each peer also adds service capacity.
        - Problems with P2P is that it isn't friendly to ISPs because there is a lot of upstream traffic that the network isn't used to, and it isn't very secure, and some users aren't ready to contribute their bandwidth and storage and computational resources.
    - Some applications use both as a messaging app may use a server to keep track of the IP addresses of all the users, but may use P2P for the sending of the messages.
- Networking application consists of pair of processes that send messages to each other over a network.
    - For each pair, one is the client (initiates the communication) and one is the server (waits to be contacted).
- When client process wants to communicate with server process, it needs to
    - Decide which transport protocol to use (UDP or TCP) depending on the needs it has (does it need reliability or speed).
    - Figure out server endpoint address.
    - Use API to connect/send/receive packets from the server.
- Through the Berkeley Socket API, a process sends and receives messages through the network through a software interface called a socket.
    - Socket is the interface between the application layer and the transport layer within a host. Also referred to as the API between an app and the kernel.
    - The application developer has control of everything on application side of the socket, but not on the transport layer side.
        - The developer can only choose the choice of transport protocol and some parameters.
        - The transport layer protocol has the job of getting the messages from the client to the socket of the receiving process.

socket(): Create a socket

bind( ): bind a socket to a local IP address and port #

connect( ): initiating connection to another socket

listen( ): passively waiting for connections

accept( ): accept a new connection

write( ): write data to a socket

read( ): read data from a socket

- Process = house, socket() = creates a door, bind() = ties door to an [IP, port number] pair, listen() = start waiting for incoming packet with matching port number.
- To identify the receiving process that the client needs to send information to, we need the address of the host and an identifier that specifies the receiving process in the destination host (which particular door in the house we want to send the information to)
    - Address of the host = IP address
    - Receiving process = Port number
- On the client side, you want to establish a socket using the socket() call, then connect the socket to the server with connect(), and then send and receive data.
- On the server side, you want to establish a socket using the socket() call, then bind the socket to [address, port number] using bind(), then listen for connections with listen(), and then accept connections with accept(), and then send and receive data.
- Normally what gets exchanged between clients and servers are web pages. The clients are normally web browsers, and the servers are the companies that provide the content on the website.

Transport Layer

- Transport layer assumes that application protocols take care of the data content. Transport layer's job is to deliver data between communicating ends.
- Transport layer cares about
    - Delivering data to the right application process
    - Delivery reliability
    - Congestion control
- Transport layer protocols can be classified along 4 dimensions.
    - Reliable data transfer - If protocol provides a guaranteed delivery service, it has this property. Some applications are loss-tolerant which are okay with some data loss.
    - Throughput - Rate at which the sending process can deliver bits to the receiving process. Applications with specific throughput requirements are bandwidth-sensitive applications. On the other hand, elastic applications can make use of whatever amount of bandwidth is available.
    - Timing - Guarantee that the messages will be received no more than some threshold of time later.
    - Security - Encryption of all the data for example.
- The two transport protocols available to applications is UDP and TCP (neither provides any encryption. There is an enhancement for TCP which is SSL and that does have encryption).
    - TCP is a communication-oriented service and a reliable data transfer service.
        - Communication-oriented: TCP has the client and server exchange transport-layer control information with each other before the application-level messages flow. This is the handshake. After the

handshake, a TCP connection exists between the sockets of the 2 processes.
        - Reliable data transfer: When one side passes bytes into socket, it can count on TCP to deliver those same bytes to the receiving socket, with no missing or duplicate ones.
    - UDP is a lightweight protocol that provides minimal services.
        - No handshaking between the two processes to start to communicate. UDP is said to be connectionless.
        - Unreliable data transfer service as there is no guarantee that the message will ever reach the receiving process.
        - Good in that you don't need to hold a connection for a long time or anything, but bad in the unreliability of data transfer.
- Mostly everything runs with TCP as their protocol because of that reliable data transfer, but a service like Skype may use UDP because it can afford data loss.
- Application layer protocol defines how an application's processes (running on different end systems) pass messages to each other.
- This protocol defines the type of messages exchanged (either request or response), the syntax of various message types (the fields in the message), the semantics of the fields (what each field means), and the rules for determining when and how a process send and responds to messages.

HTTP

- The Web's application level protocol is called HTTP.
    - HTTP defines the format and sequence of messages exchanged between browser and web server.
- As an application protocol, it assumes the network provides a way to send data to any hosts on the Internet. They don't care how the data is sent, but do care that it's sent reliably.
- The Web is a client server application that allows users to obtain documents from Web servers on demand.
- HTTP is implemented into two programs, the client program and a server program.
    - HTTP client first establishes a TCP connection with the server, server accepts TCP connection from client, and then they talk to each other by exchanging HTTP messages, and then the TCP connection is closed.
    - Client is the browser that requests, receives, and displays Web objects.
    - Server is the Web server that sends objects in response to requests.
- A Web page consists of objects, which are files that are addressable by a single URL (contains the hostname of the server that houses the object and the object's path name). Web pages have a base HTML and several referenced objects.
- Web browsers = client side of HTTP, Web servers = server side of HTTP
- HTTP uses TCP as its underlying transport protocol. HTTP client first establishes a TCP connection with the server, server accepts TCP connection from client, and then the

browser and server access TCP through their socket interfaces. The client then sends HTTP request messages into the socket interface. Once it's sent, the message is in the hands of TCP.

- HTTP is stateless because the server sends the requested files to clients without storing any state information about the client itself.
    - Because the HTTP server maintains no information about the clients, HTTP is a stateless protocol.
- Non-persistent connections are when all HTTP requests and their corresponding responses are sent on separate TCP connections.
    - TCP connection is closed after the server sends the object.
    - The connection should transport exactly one request message and one response message.
- Persistent connections are when multiple objects can be sent over the same TCP connection.
- Pipelining is when you send requests for multiple objects (as soon as you see them referenced in the index file) one after another without waiting for a response from the server.
- Parallel connections are when you're able to set up multiple TCP connections at once.
- 3 Factors in HTTP fetching are persistent connections, parallel connections, and usage of pipelining.
- HTTP/1.0: non-persistent connection (at least 2 RTT for every object) and no pipelining (I guess)
- HTTP/1.1: persistent connection, pipelining isn't supported by default, handles requests in strict sequential order (causes head-of-line blocking issues), big HTTP header with repetitive information.
- HTTP/2: All the good stuff with HTTP/1.1 + header compression (reduced overhead), multiple streams (within a single TCP connection) that can end data, but still has head-of-line blocking problem.
    - Streams are useful because you can break up one huge file into chunks and then send those through the different streams.
- QUIC: Runs over UDP, no handshaking, improved congestion control, uses streams so no head-of-line blocking, compresses headers.
- HTTPS: Ensures that you're receiving data from the place you think you're getting it from. Basically ensure secrecy of information exchanges. It provides authentication of the website that you're going to. HTTPS requests a certificate to make sure that the owner owns the domain name.
    - Each HTTPS certificate is associated with a domain name.
- Under HTTPS, sites can have domain validation (DV) and extended validation (EV) certificates where the Certificate Authority (CA) checks the right of the applicant to use a domain name and does a thorough vetting of the the organization
    - A domain validation certificate validates the identity of applicants by proving the control over a DNS domain.

    - Extended validation is different because it requires a verification of the applicant's identity by a certificate authority.
- HTTP Certificate States: Valid, expired, wrong host, untrusted root, etc.
- Round trip time (RTT) is the time it takes for a small packet to travel from client to server and then back to the client.
- Transmission time is the time that it takes between the server receiving the HTTP request and beginning to send the response.
- Each object in a non-persistent connection takes 2 RTTs (one to set up the TCP connection and one for HTTP request and response of the first index html file) + transmission time
- 2 types of HTTP messages: request and response
- First line of HTTP request message is the request line and the subsequent lines are the header lines.
- Request line has 3 fields
    - Method field - GET, POST, HEAD, PUT, DELETE, etc
    - URL field
    - HTTP version field
- "Connection: close" represents a non-persistent connection because the server will close the session after the object is sent.
- With a POST method, the body of the entity depends on what the user enters.
- When a server receives a request with the HEAD method, it responds with an HTTP message but leaves out the requested object.
- An HTTP response message has an initial status line, 6 header lines, and an entity body.
    - The entity body contains the object itself.
    - The status line has the protocol version field, a status code, and a corresponding status message.
- The "Date:" header indicates the time and date when the HTTP response was created.
- Common status codes
    - 200 OK: Request succeeded.
    - 301 Moved Permanently: Requested object has been permanently moved.
    - 400 Bad Request: Request couldn't be understood by server
    - 404 Not Found: Requested document doesn't exist on server
    - 505 HTTP Version Not Supported
- Since HTTP servers are stateless, they can't remember which clients have requested what type of data from the servers. All requests are self contained. In order for the server to restrict user access or serve content as a function of user identity, HTTP uses cookies to allow sites to keep track of users.

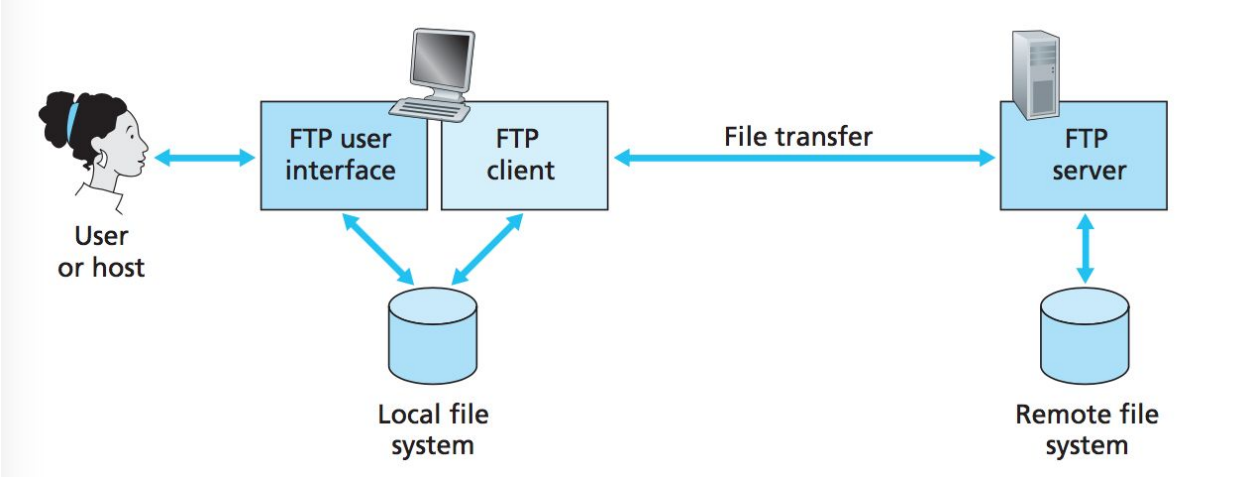
Cookies and Caches

- Cookies have 4 different components.
    - Header line in HTTP response message
    - Header line in HTTP request message

    - Cookie file kept on user's end system and managed by user's browser
    - Back-end database at the Web site
- When a first time request from a user comes into the Amazon Web server, the server creates a unique ID number and creates an entry in its back-end database that is indexed by ID numbers. Server then responds to Susan's client and sends back a message with a "Set-cookie" header that contains the ID number. The browser will see that header and then append a line to the special cookie file that it manages (which is basically a dictionary of all the websites the client has visited and the respective cookie numbers). Now, every time Susan goes to Amazon.com, the browser will consult that file, extract the ID number, and put that number in the cookie header line in the HTTP request. Since the cookie number will always be sent in the HTTP requests, the server will always know the it's Susan who is accessing the website.
- A Web cache or a proxy server is a network entity that satisfies HTTP requests on the behalf of an origin Web server.
    - It has it's own disk storage and keep copies of recently requested objects.
    - The proxy server is basically a middleman between the origin server and the client. The client goes to the proxy server first to check if it has the requested item in its cache, and if so, returns it to the client, and if not, goes to the origin server to retrieve it.
- The cache is technically both a client and server at the same time, since it can receive messages from the browser and can send messages to the origin server if it doesn't have the requested object in the cache.
- Web caching is great because
    - It can substantially reduce the response time for a client request, especially if bandwidth between client and origin server < bandwidth between client and cache.
    - Can reduce traffic on an institution's access link to the Internet.
    - Reduces load on the origin server.
- One problem that comes up is if the object in the cache of the proxy server is not the most updated version of the object (which would be stored in the origin server). To deal with this, the origin server and the proxy server exchange conditional GET messages where the origin server returns the most updated version of the object if it has changed since some given date.
- A Content Distribution Network (CDN) is what installs many distributed caches throughout the Internet.
    - This helps with localizing the traffic.
- Although caching can improve response times, there is a risk that the copy of an object residing in the cache may not be the latest version.
    - To solve this, you can use a HTTP request message called a conditional GET message which includes the "If-Modified-Since" header line.

FTP (File Transfer Protocol)

- In an FTP session, a user in a local host can transfer files to and from a remote host, but first needs to provide proper authentication.
- User interacts with FTP through an FTP user agent, which communicates with the FTP client, which then establishes a TCP connection with the FTP server (on port number 21), and now files are able to be transferred.

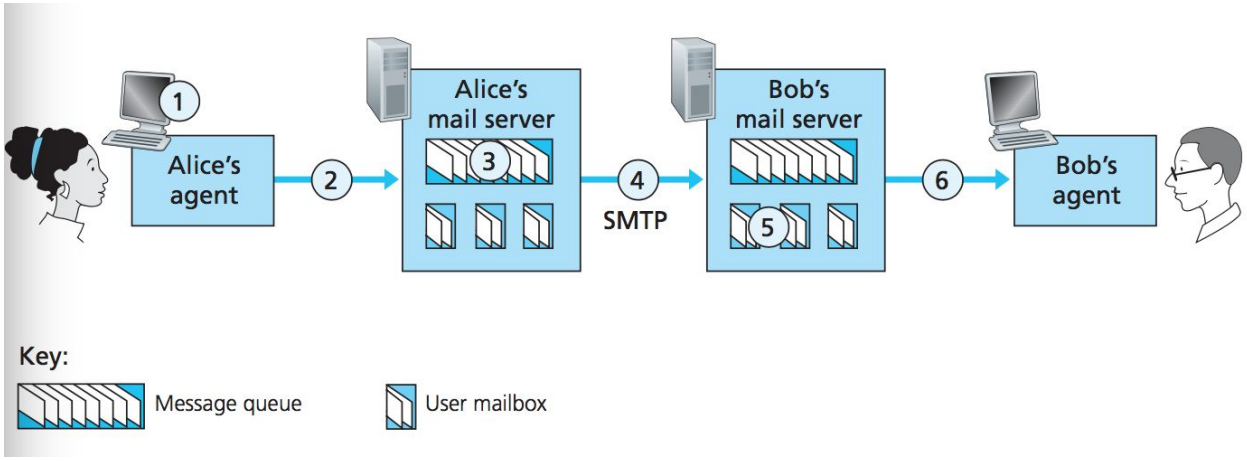


- HTTP and FTP are oth file transfer and application layer protocols that run on top of TCP (which is a transport layer protocol).
- FTP uses two parallel TCP connections to transfer a file
    - Control connection: Used for sending control information between two hosts, such as information for user authentication and commands.
    - Data connection: Used to actually send the file
- This usage of a separate control connection means that it's control information is out-of-band. Since HTTP sends request and response header lines into the same TCP connection, HTTP sends its control information in-band.
- FTP data connections are non-persistent in that FTP sends on file over the data connection and then closes that connection.
- FTP servers maintain state about a user, because it must associate the control connection with a specific user account and must have info on user's current directories.
- FTP commands and replies are similar to that of HTTP.
- Common responses in FTP
    - 331 Username OK, password required
    - 125 Data connection already open
    - 425 Can't open data connection
    - 452 Error writing file

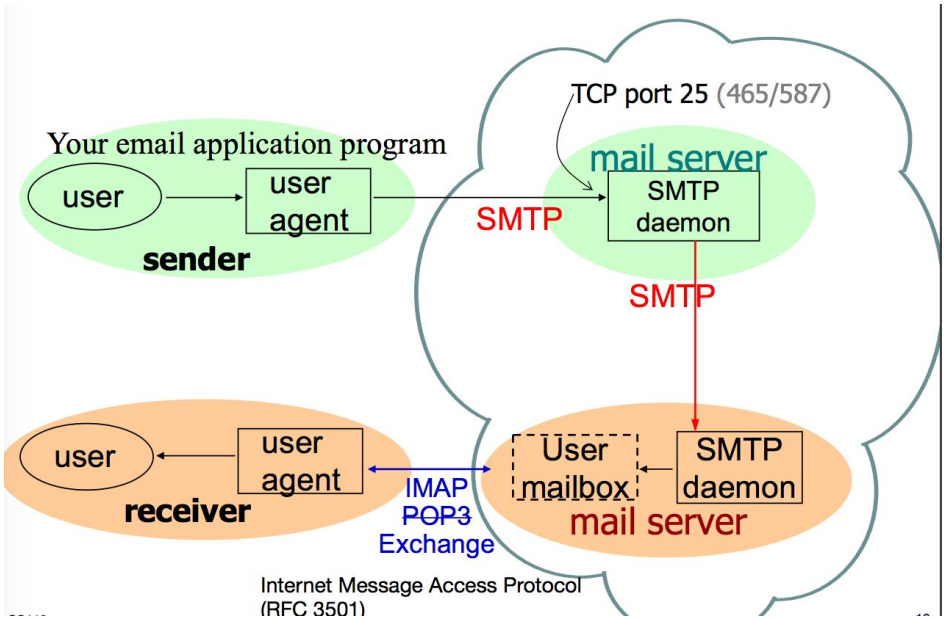
Email Protocols

- 4 Components of Email
    - User Agents: Mail apps that allow you to compose, edit, and read messages

    - Mail Servers: Composed of mailbox containing incoming messages and message queue of outgoing messages.
    - Simple Mail Transfer Protocol which transfers email messages from user clients to servers and between mail servers.
    - Mail Retrieval Protocols like IMAP to decide how the receiver user agent should download the incoming files.
- Simple Mail Transfer Protocol (SMTP) is the heart of Internet electronic mail.
    - Transfers messages from senders' mail servers to the recipient's mail servers.



- SMTP also runs on TCP (helpful because TCP has that reliable data transfer) as client side of SMTP sees the message from Alice's user agent, opens a TCP connection to an SMTP server (on Bob's mail server), does the handshake, and sends the message into the TCP connection, then the server side of SMTP receivers the message, and then Bob's mail server places it in his mailbox, and Bob's user agent displays the mail to Bob.



- The client SMTP is running on the sending mail server host. It needs to make a TCP connection with the server SMTP which is running on the receiving mail server host.

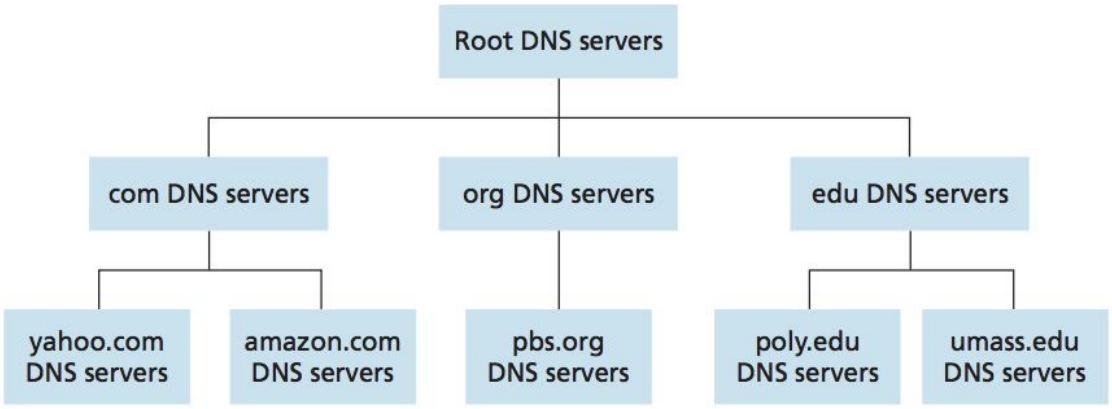
- SMTP uses persistent connections, as all messages can be sent over the same TCP connection.
- HTTP transfers files from a Web server to a Web client (browser) while SMTP transfers files (email messages) from one mail server to another mail server.
- HTTP is mainly a pull protocol as someone loads info on a Web server and users use HTTP to pull information from the server. (TCP connection is initiated by whoever (browser) wants to receive the file). SMTP, however, is a push protocol because the TCP connection is initiated by the machine that wants to send the file.
- SMTP also requires each message to be in 7-bit ASCII format.
- HTTP is different from SMTP because each object for HTTP has to be enclosed in it's own response message while you can fit multiple objects in one multipart message.
- The mail server is an always on machine and therefore it shouldn't and doesn't reside on the user's local PC.
- The final step in everything is a special mail access protocol that transfers messages from Bob's mail server to his local PC.
    - Post Office Protocol - Version 3: Simple mail access protocol. Client opens TCP connection with the mail server, user sends username and password, user retrieves messages, and after quit command, the server deletes all the messages that were marked to be deleted. Download and delete mode. Also is stateless across sessions.
    - Internet Mail Access Protocol: POP3 doesn't allow users to create remote folders and assign messages to these folders. Also keeps all messages in one place (the server). Keeps user state across sessions.
- DKIM is another security layer in SMTP and it allows the receiver to check that an email claimed to have come from a specific domain was indeed authorized by the owner of that domain.
- SPF is another security layer is that it is a mechanism to check that incoming mail from a domain comes from a host authorized by that domain's administrators.
- In order to secure email, sign with a valid certificate that is issued by a certification authority, who can check and vouch that it is you who sent the email.
- PGP is an encryption program that authenticates data communication.

DNS (Domain Name System)

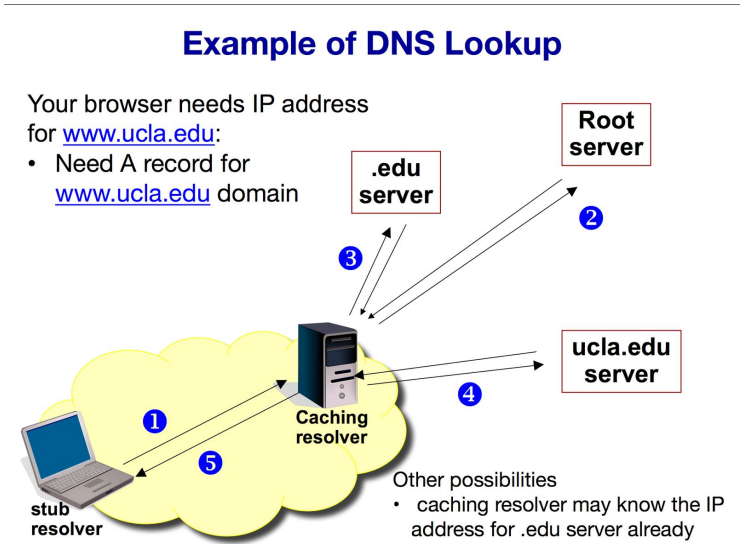
- Identifiers for hosts can be their hostname (cnn.com) or their IP addresses.
- IP addresses are hierarchical because we scan the address from left to right and then obtain more and more specific information about where the host is located.
- Main task of domain name system (DNS) is to be a directory service that translates hostnames to IP addresses.
    - It is a distributed database that is implemented in a hierarchy of DNS servers.
    - It is also a protocol that allows hosts to query the distributed database.
        - Protocol runs over UDP

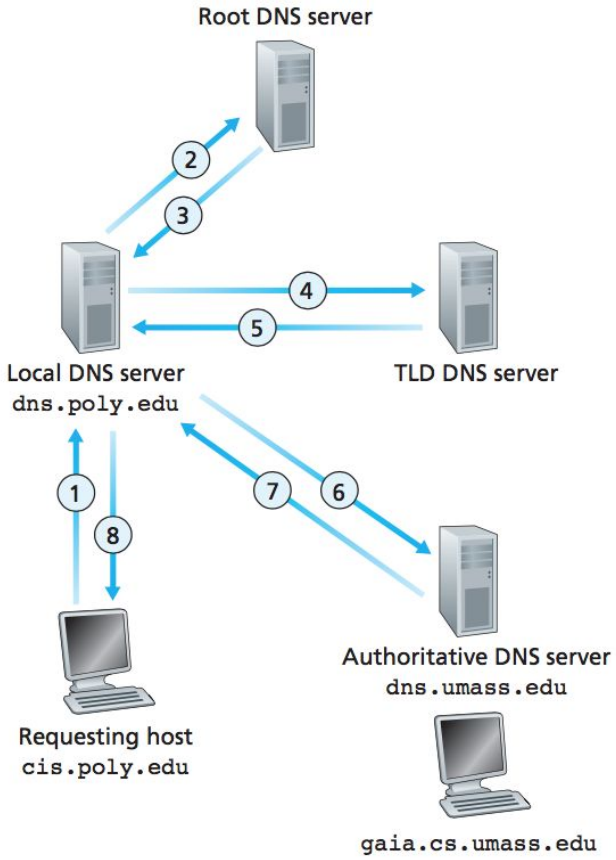
- ICANN manages the Internet resources such as addresses, protocol identifiers and parameters, and names.
- 4 Main Parts of DNS
    - Defines a hierarchical name space
    - Creates a distributed database implemented through a hierarchy of authoritative servers.
    - Caching resolvers look up the database
    - Use DNS query protocols
- DNS can be employed by HTTP
    - For example, if a browser requests some URL, it has to know what IP address corresponds to that URL and then send a TCP connection and all that stuff.
    - A DNS client basically takes the URL, extracts the hostname, and then sends a query to a DNS server to get back the IP address for that hostname, and then once the browser receives the IP address, it can initiate a TCP connection to the HTTP server process located at port 80 at that IP address.
- A host with a complicated hostname can have one or more alias names, but the original is the canonical hostname.
    - DNS can be used to get the canonical hostname for a supplied alias hostname.
- DNS performs load distributions for busy sites like cnn.com, as it can be replicated over multiple servers where each server runs on a different end system and each has a different IP address. A set of IP addresses is thus associated with one canonical hostname.
    - When clients make a DNS query, the server responds with the entire set of IP addresses, but rotates the ordering, so that the HTTP request will be sent to a different address.
- Applications that need to translate hostname to IP will invoke the client side of DNS, specifying the hostname that needs to be translated, normally though a call to gethostbyname().
- At its most basic level, DNS is just a mapping service and applications will just treat it as a black box. Under the hood, there are a lot of DNS servers and a whole distributed database. We don't use a single centralized server and database because it means there's a single point of failure, high traffic volume, only one physical location for the database, and maintenance would be huge.
- DNS thus uses a large number of servers organized in a hierarchical fashion and distributed around the world.
    - No single server has all the mappings for all the hosts in the internet.
- 3 Classes of DNS servers
    - Root DNS nameservers - 13 root DNS servers. Each server is technically a replicated server though. Lists the names and IPs of the authoritative servers for all the TLDs.
    - Top level domain (TLD) servers - Responsible for .edu, .gov, etc

    - Authoritative servers - Has to have mappings that map the hostname to IP addresses. These are provided by the individual domain owners. These contain the actual info



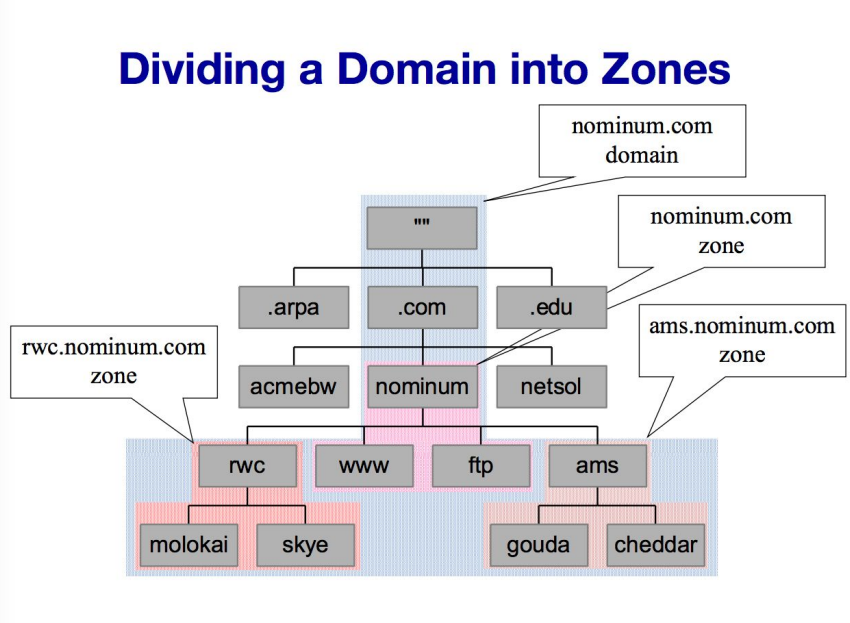
- A client will first contact the root server, which returns the IP addresses for TLD servers, for that specific TLD (.com, .org, etc), then the client contacts one of those servers, which returns the IP address of an authoritative server for the actual site (amazon.com), the client contacts that server, which returns the IP address of the hostname (www.amazon.com)
- Each ISP also has a local DNS server (same as caching resolver), which is where clients will send their first DNS query. This local DNS then follows the process of sending to root, get response, sending to TLD, etc
    - Basically, it's just another server in the whole process.
- Reverse DNS lookups are when you provide the IP address and want to get back the hostname.
- Recursive queries are where the replies to the query don't follow immediately. Iterative ones are where the reply is directly returned to the requesting source.
    - Query from requesting host to the local DNS server is recursive while the other queries (from local to root, local to TLD, etc) are iterative.





- DNS caching works to reduce delay (increase performance) and reduce the number of DNS messages that are flying around.
- When a DNS server receives a reply, it can caching the mapping in its local memory, so that the next time the mapping is requested, it can be processed quickly.
- The 2 main reasons that DNS has great scaling properties are:
    - Replication of authoritative name servers to distribute the load.

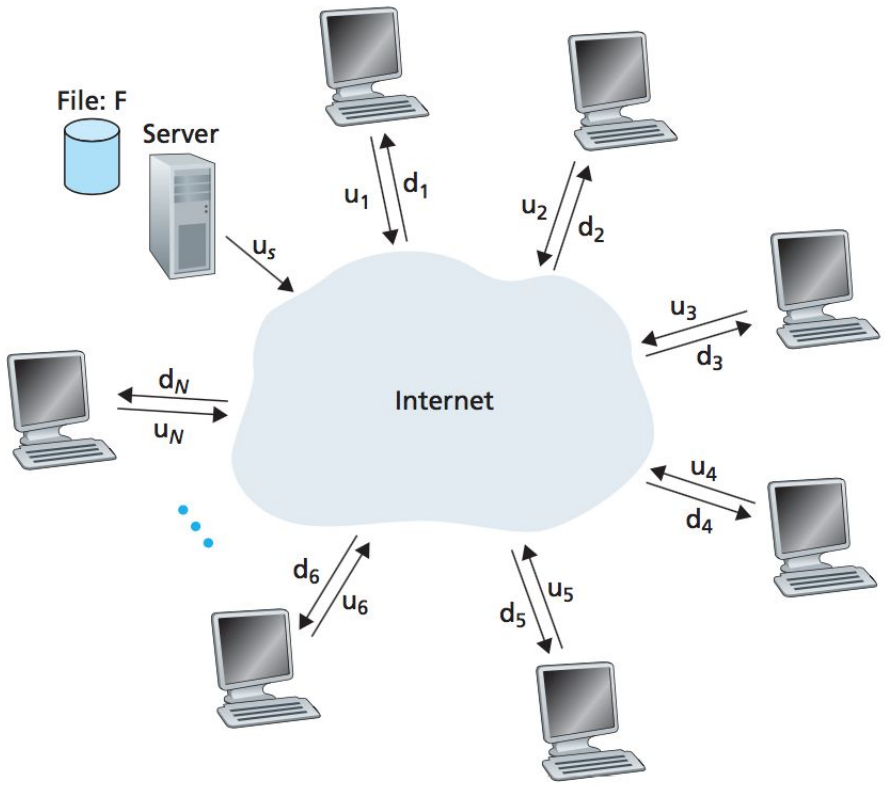
    - DNS caching (to save extra requests to different servers)
- DNS servers that implement the DNS distributed database store resource records (RRs) that provide hostname-to-IP address mappings.
    - (Name, Value, Type, TTL)
        - Type A records provide the standard hostname to IP address mapping
            - Ex) (relay1.bar.foo.com, 145.36.43.121, A)
        - Type NS routes DNS queries further along in the query chain. The Name is the domain and the Value will contain the name of an authoritative DNS server that knows how to obtain the IP addresses for hosts in the domain.
            - Ex) (foo.com, dns.foo.com, NS)
- If a DNS server is authoritative for a particular hostname, then the DNS server will contain a Type A record for that hostname.
    - A server can have both NS and A records.
- Two kinds of DNA messages are query and reply messages.
- First 12 bytes of DNS messages are headers, and then there's a question section which has info on the query being made, then there is the answer section on DNS replies, then the authority section (with records of other authoritative servers), and the additional section.
- Registrar is a commercial entity that checks the uniqueness of a domain name, and then enters the domain name into the DNS database (also called the registry), and makes you pay.
    - The Internet Corporation for Assigned Names and Numbers accredits the various registrars.
- Registrants are the regular people who pay for the service.



- Different name servers can serve different zones.
- The two types of nameservers are authoritative (which maintain the data) and caching (which store the data obtained from an authoritative server).

P2P (Peer to Peer) Architecture

- With P2P architecture, no reliance on always-on infrastructure servers.
- Communication is through peers, and peers are not owned by a service provider, but are laptops and desktops controlled by users.
- File distribution and a database distributed over community of peers are two applications of P2P architectures.
    - File Distribution: Distributing a large file from a single server to a large number of hosts, called peers.
        - In client-server, one server must send a copy of the file to each of its peers which places a lot of burden on the server.
        - In P2P, each peer can redistribute any portion of the file it has received.
    - Distributed database (Distributed hash table): Create a database that has millions of key-pair values over millions of peers, where each peer only holds a subset of all of the key/value pairs and any peer will be allowed to insert new key-value pairs into the database.
        - In client-server, we would just have all the pairs stored in one central server.



- In the above architecture, we see a peer to peer system where u represents upload rate and d is download rate.

- Minimum time to send N files in Client Server

$$D_{cs} \geq \max\left\{\frac{NF}{u_s}, \frac{F}{d_{min}}\right\}.$$

- Minimum time to send N files in P2P

$$D_{\text{P2P}} \geq \max\left\{\frac{F}{u_s}, \frac{F}{d_{min}}, \frac{NF}{u_s + \sum_{i=1}^{N} u_i}\right\}$$

- With P2P file distribution, any user can use a technique called rarest first as they try to decide which chunks of the file they should request from their neighbors and which chunks she should send.
  - Rarest first determines from the chunks she does not have, which are the ones that are rarest among her neighbors (chunks that have fewest repeated copies), and then request those chunks first.
- To determine which requests she should respond to, she gives priority to the neighbors that are supplying her data at the highest rate.
  - The top 4 (for example) peers are called unchoked.
  - The neighbor that is picked at random is called optimistically unchoked.
- Designing a Distributed Hash Table
  - Assign identifier to each peer
  - Assign each (key, value) pair to the peer whose identifier is the closest to the key.
  - The problem comes up where how do we know the peer who has the closest ID number to the specific key. Each peer technically would have to keep track of all other peers, but in order to avoid this, we can say that each peer is only aware of its immediate successor and predecessor.
  - Circular arrangement of peers is a special case of an overlay network.
  - Messages get sent in a circular fashion to see who is the best fit for the key.
- There is a tradeoff between the number of neighbors each peer has to track and the number of messages that DHT needs to send to resolve a query.
- You can also have a DHT with some shortcuts, which may help on cutting down the number of messages.
- Need to also periodically check whether peers are still there.
  - If one peer leaves, then that gap has to be accommodated.

Chapter 3 - Transport Layer

Media

- A lot of Internet video is based on HTTP protocol because of:
    - Very good scaling properties in replication (CDN boxes)
    - Caching.
    - These are similar to the great qualities of DNS
- In HTTP, video is split into different chunks and each chunk may have different qualities.
- Media Presentation Description (MPD) Data Model describes accessible segments and corresponding timing.
- The challenges with media streaming protocols are rate control (determining the right sending rate based on the available bandwidth), error control (video quality in the presence of packet loss), and continuous distribution (no guarantees on expectation or variance of packet delay).

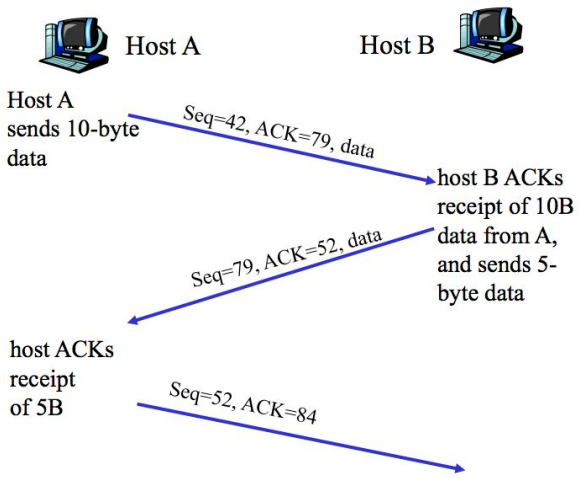
Transport Layer

- Transport layer lies in between the application and network layers.
    - Provides communication services directly to the application processes running on different hosts.
    - Transport layer provides communication between processes and network layer provides communication between hosts.
- First critical function of transport layer is to extend the network layer's delivery service between two end systems to a delivery service between two application layer processes running on the end systems.
- A transport layer protocol provides for logical communication between applications processes running on different hosts.
    - Application processes use the logical communication provided by the transport layer to send messages to each other, free from the worry of how the messages are physically transported.
- On the sending side, the transport layer converts the application layer messages it receives from the sending application process into transport-layer segments.
    - Breaks messages into smaller chunks and adds a transport-layer header.
    - That whole thing is now known as a transport-layer segment.
- You can think of network layer as being the postal service where it moves mail from house to house but not from person to person.
    - Application messages = letters in the envelopes
    - Processes = cousins
    - Hosts = houses
    - Transport-layer protocol = Ann and Bill
    - network-layer protocol = postal service
- Transport-layer protocols live in the end systems.
- A computer network may make use of multiple transport protocols, with each protocol offering a different service model to applications.

- If the network layer protocol can't provide delay or bandwidth guarantees for transport layer segments sent between hosts, then the transport layer protocol can't make the same guarantees for the application messages sent between processes.
- The Internet's network layer protocol is called IP, Internet Protocol
    - Provides logical communication between hosts.
    - Service model is a best-effort delivery service where the IP makes its best effort to deliver segments between communicating hosts, but makes no guarantees.
    - It is an unreliable service.
- The job of UDP and TCP is to extend IP's delivery service between two end systems to two processes running on end systems.
- Extending from host to host to process to process is called transport-layer multiplexing.
- TCP provides reliable data transfer and congestion control.
    - Congestion control prevents any one TCP connection from swamping the links and routers between communicating hosts.
        - Done by regulating the rates at which the sending sides of TCP connections can send traffic into the network.
    - TCP provides congestion control (reduces traffic overload in the network), flow control (receiver can set flow control window size to prevent sender from flooding), and connection setup.
- Transport layer receives segments from the network layer below it, and has the responsibility of delivering the data in these segments to the appropriate application process running in the host. In other words, the transport layer has to provide a multiplexing/demultiplexing service in order to pass data between network layer and the correct application level process.
    - A process can have sockets, which are basically doors where data passes from the network to the process and vice versa.
    - Therefore, the transport layer in the receiving host doesn't deliver data to the process, but rather delivers it to a socket.
- The job of delivering the data in a transport layer segment to the correct socket is called demultiplexing.
    - Aka when Bill receives all the mail, looks at who the mail is addressed to, and then distributing it to the right people.
- The job of gathering data chunks at the source host from the different sockets, encapsulating each chunk with header information, and passing the segments to the network layer is called multiplexing.
    - Aka when Ann collects all the mail from her brothers and sisters and then gives it to the post office service.
- Each transport layer data segment needs to have a source port number field and a destination port number field.
    - When data segments are sent from a source to a destination, the transport layer needs to look at these fields, and is then able to direct the information to the correct sockets. The kernel does this by maintaining a table and when data

packets come in, the kernel will look at the table and reroute the packet to the correct process based on the src/dest IP and port numbers.
        - The receiving host demultiplexes each segment to the appropriate socket by examining the segment's destination port number.
- Difference between TCP sockets and UDP sockets is that TCP sockets are identified by a 4-tuple (source IP, source port number, destination IP, destination port number)
- A server host can support many simultaneous TCP connection sockets with each socket attached to a process, and with each socket identified by it's own 4-tuple.
    - When a TCP segment arrives, all 4 fields are used to demultiplex the segment to the appropriate socket.
- Even if two different clients have the same source port numbers, that's okay because the 4-tuple will include information about the IPs of both clients, so the server will be able to make that distinction between the two.
- When clients send segments to a server, all segments will have destination port 80.
    - Server distinguishes the segments from different clients using source IP addresses and source port numbers.
- The 3 types of errors to fight in reliable data transfer are corrupted bits in a packet, packet loss, and packets arriving out of order.
    - The 3 components in reliable data transfer to address the problems are sequence number, ack number, and retransmission timers.
- (UDP > TCP) for DNS specifically because
    - Finer application level control over what data is sent and when it is sent.
    - No connection establishment/no handshaking. In DNS, the idea is that you don't very *need* a consistent connection. You're just very sending one small request or response.
    - No connection state.
    - Smaller packet header overhead (TCP - 20 bytes, UDP - 8 bytes of header overhead in each segment)
    - DNS replies can fit in one packet
- UDP also has checksums to detect bit errors in the transmitted message.
- TCP is a reliable data transfer protocol that is implemented on top of an unreliable IP end to end network.
- Reliable data transfer protocols are based on retransmission protocols like ARQ (Automatic Repeat reQuest) where the receiver sends control messages to let the sender know what it has received correctly.
- 3 protocol capabilities are required in ARQ protocols to handle the presence of bit errors.
    - Error detection: Mechanism to allow receiver to detect when bit errors have occurred. UDP uses the Internet checksum field. You can also use sequence numbers to uniquely identify individual pieces of data.
    - Receiver feedback: ACK and NAK acknowledgment replies.
    - Retransmission: Packet that is received in error at the receiver will be retransmitted by the sender. Each data packet also has a sequence number to determine whether or not the received packet is a retransmission.

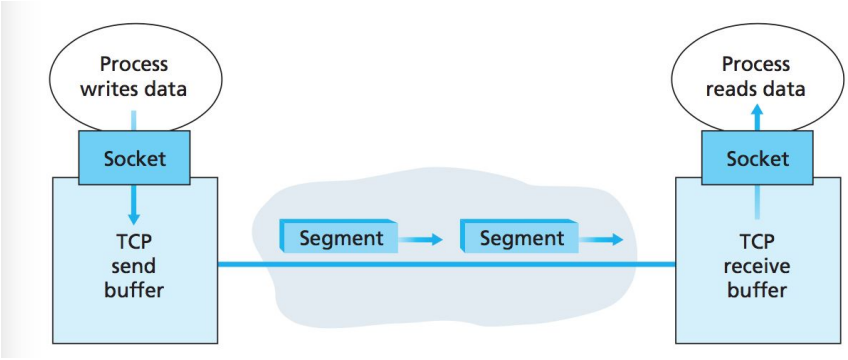
- In Go-Back-N protocol, the sender is allowed to transmit multiple packets without waiting for acknowledgement, but can't more more than N unacknowledged packets in the pipeline.
    - N is referred to as the window size.
- In GBN, an acknowledgment for a packet with sequence number n will be taken to be a cumulative ACK, indicating that all packets with sequence number up to and including n have been correctly received at the receiver.
    - If a timeout occurs, sender will resend all packets that have been previously sent but not yet acknowledged.
- In GBN, if the packet received is not in order, then the receiver will discard the packets and resend an ACK for the most recently received in order packet.
- Selective repeat protocols are different because they have the sender retransmit the packets that it suspects were received in error at the receiver.
    - SR also buffers out of order packets.
- Ack numbers and seq numbers



TCP

- TCP is connection oriented because before one application process can begin to send info to another, the two process must first handshake with each other.
- A TCP connection provides full-duplex service, in that if there is a TCP connection between Process A on one host and Process B on another host, then application layer data can flow from A to B at the same time as application layer data flows from B to A.
- A TCP connection is always point-to-point, meaning that there is a single sender and a single receiver.
- TCP goes by a 3-way handshake to establish a connection between 2 hosts.
    - Client host sends TCP SYN segment to server, which carries initial seq and doesn't carry data. The server receives this SYN, replies with a SYN ACK and SYN control segment. The client host then sends SYN ACK.

- When a client wants to send data, it passes through the socket and TCP directs the data to the connection's send buffer. The max data that can be grabbed and put in a segment is limited by the maximum segment size (MSS).



- TCP pairs each chunk of client data with a TCP header, which forms a TCP segment, which are passed down to the network later.
- The acknowledgment number that Host A puts in it's segment is the sequence number of the next byte Host A expects from Host B.
    - TCP also provides cumulative ACKs.
- In order to determine what to set the timeout timer at, we need to estimate the RTT.
- Keep in mind TCP only measures SampleRTT for segments that have been transmitted only once.
- Such an average, as the one above, is called an exponential weighted moving average.
- DevRTT measures the variability of the RTT values.

$$\text{difference} = \text{SampleRTT} - \text{SRTT}$$

$$\text{SRTT}' = (1-\alpha) \times \text{SRTT} + \alpha \times \text{SampleRTT}$$

$$= \text{SRTT} + \alpha \times \text{difference}$$

$$\text{DevRTT}' = (1-\beta) \times \text{DevRTT} + \beta \times |\text{difference}|$$

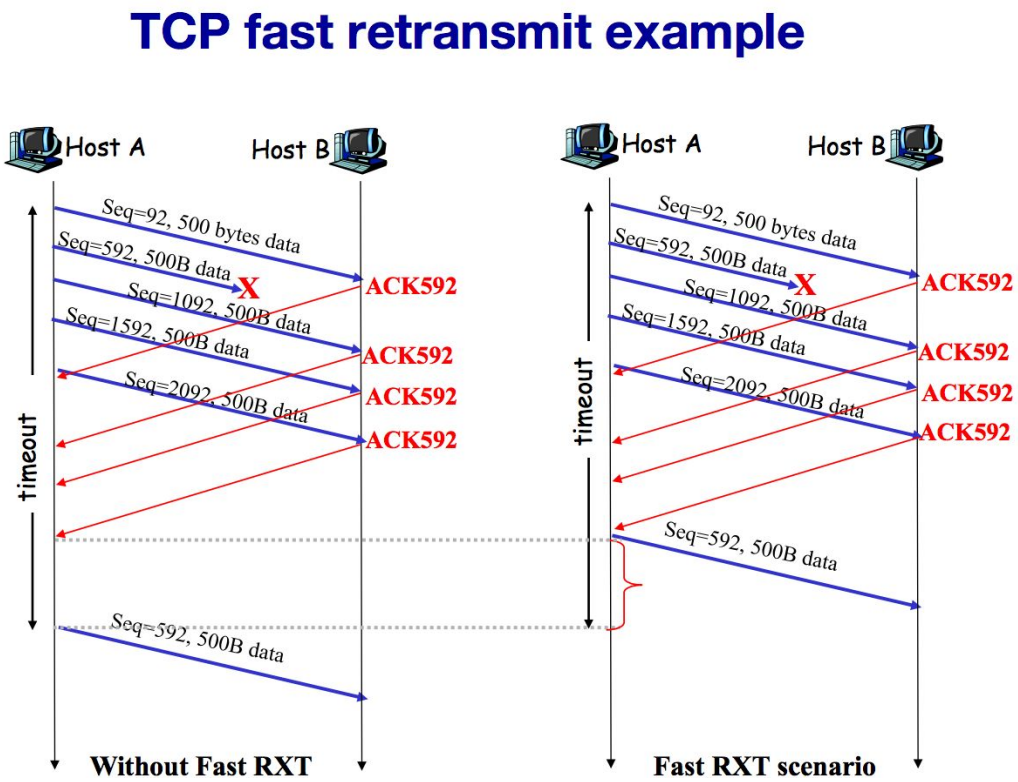
$$= \text{DevRTT} + \beta\,(|\text{difference}| - \text{DevRTT})$$

- TCP determines the time for timeout using the following formula.

$$\text{TimeoutInterval} = \text{EstimatedRTT} + 4 \cdot \text{DevRTT}$$

- In the case of retransmission, do not take the RTT sample (don't update SRTT or DevRTT). Double the RTO value after each time out, and take the RTT measure again upon the next data transmission.
- In terms of initial values, if the SRTT is too small, then the RTO will be small and we'll have unnecessary retransmissions and if the values are too large, we will wait too long before retransmitting.

- Important to remember that TCP creates a reliable data transfer service on top of IP's unreliable best effort service.
- The timeout works by TCP responding to the timeout by retransmitting the segment that caused the timeout, and then TCP restarts the timer.
- Whenever a timeout occurs, TCP retransmits the not-yet-acknowledged segment with the smallest sequence number.
    - Each time TCP retransmits, it sets the next timeout interval to twice the previous values, rather than deriving it from the previous equation.
- TCP can also detect that something is wrong whenever it receives duplicate ACKs (it doesn't necessarily have to wait for a timeout). Fast retransmit is when TCP will retransmit a missing segment before that segment's timer expires.
    - If a sender receives 3 dup ACKs for the same data, it takes this as an indication that the segment following the segment that has been ACKed 3 times has been lost.



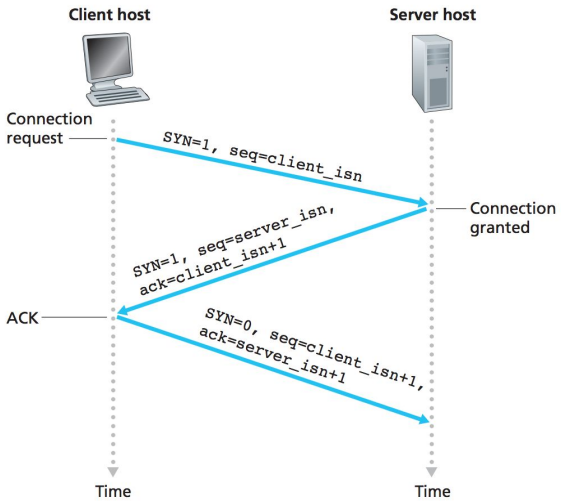
- TCP is like GBN a lot because of that cumulative ACK, but TCP implementations will differ correctly received but out of order packets. Therefore, it doesn't have to do a lot of retransmissions.
- Selective acknowledgment is a TCP feature that allows a receiver to ACK out of order segments selectively rather than just cumulatively ACKing the last correctly received in order segment.

Flow Control and Congestion Control

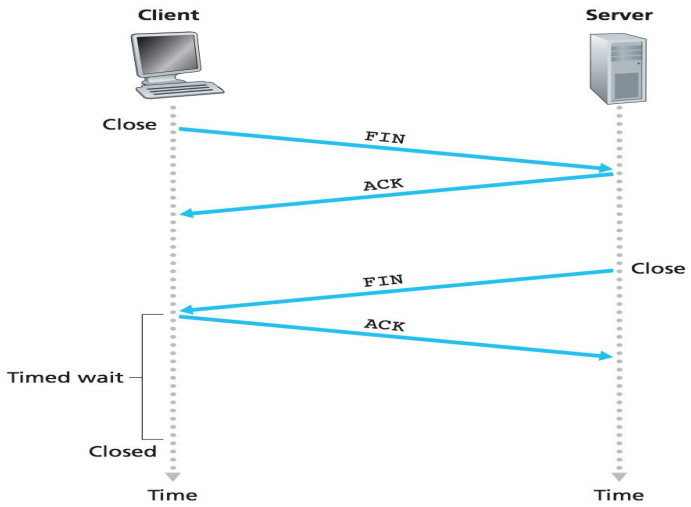
- Hosts on each side of a TCP connection set aside a receive buffer for the connection. The TCP connection will read bytes, place data in the buffer, and then the application process will read the data from that buffer at some later time.
- TCP provides a flow control service to applications which eliminates the possibility of the sender overflowing the receiver's buffer.
    - It matches the rate at which the sender is sending against the rate at which the receiving application is receiving. Basically it prevents the sender from overrunning the receiver by transmitting too much too fast.
    - It is able to provide this service by making the sender maintain a variable called receiver window, which gives the sender an idea of how much space is left in the receiver buffer.
    - Receiver informs sender of amount of free buffer space it has, and sender keeps the amount of transmitted, unACKed data no more than the most recently received RcvWindow value.

$$\texttt{LastByteRcvd} - \texttt{LastByteRead} \leq \texttt{RcvBuffer}$$

- The above statement has to be true in order for TCP to not overflow the allocated buffer.



- The above shows a typical TCP 3 way handshake.

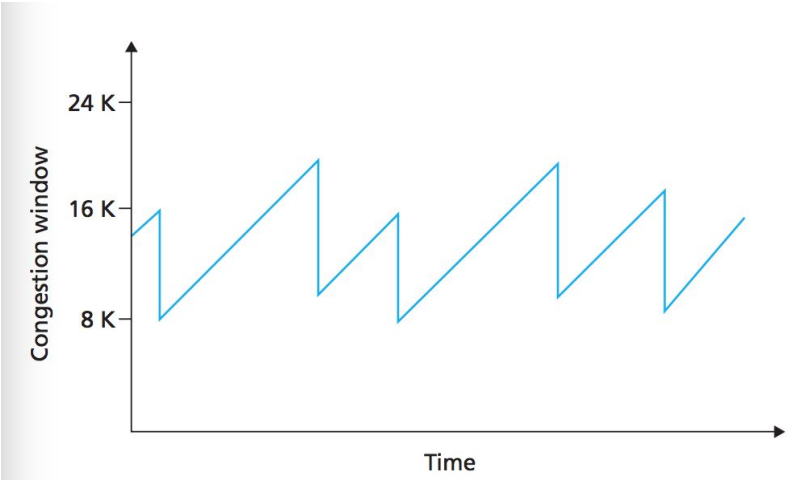


- The above shows the communication that happens when a connection is terminated.
- Congestion control is when too many clients are trying to send data at too high of a rate.
- TCP approaches congestion control by having each sender limit the rate at which it sends traffic into its connection as a function of the perceived network congestion.
    - Little congestion = increase sending rate
    - Lot of congestion = decrease sending rate
- The congestion window imposes the constraint on the rate at which the sender can send traffic into the network.
- Disregarding the receive buffer, the sender can have a send rate of cwnd/RTT bytes/sec.
- The real problem here is how do we analyze how much congestion there is in the network at a given time, and then based on that info, how do we adjust cwnd accordingly? And also, how do we communicate with all the senders? They all have to be on the same page.
- There are 3 main ways TCP answers these questions.
    - A lost segment implies congestion, and hence, the sender's rate should be decreased when a packet is lost (loss event).
    - An acknowledged segment indicates the network is delivering the segments to the receiver, and hence the sender's rate can be increased when the ACK arrives.
    - Basically, the above is a way of bandwidth probing where the sender increases the rate until the congestion onset begins, and then backing off that rate.
- The TCP congestion control algorithm has slow start, congestion avoidance, and fast recovery.
    - Slow Start: When a connection begins, the value of cwnd is set to be 1 MSS. The send rate will thus start slow (multiply cwnd by 2 for every successful ACK) but will grow exponentially. If there is a loss event, the sender sets the value of cwnd back to 1, sets ssthresh to cwnd/2 and begins the process again.
    - Congestion Avoidance: Rather than doubling the value of cwnd every RTT, TCP sometimes will increases the cwnd by only 1 in situations where you know that if you double the cwnd, you're most likely to encounter a loss event.

        - Ssthresh defines the boundary between slow start and congestion avoidance (When cwnd < ssthresh, use slow start . Else use congestion avoidance).
        - If all packets get ACKed successfully, increase cwnd by 1
        - If there is some loss event, set cwnd to cwnd/2.
    - Fast Retransmit: If sender receives 3 duplicate ACKs then it will resend the segment before the timer expires.
    - Fast Recovery (This is in TCP Reno): When a loss is detected through 3 duplicate ACKs, we reduce cwnd by half (regardless of whether in slow start or congestion avoidance).
    - Basically, in TCP Reno, regardless of whether you're slow start or congestion avoidance, if there are 3 dup acks, then always set ssthresh to cwnd/2 and set cwnd = ssthresh, and set state to congestion avoidance.

State	Event	TCP Sender Action	Commentary
Slow Start (SS)	Received ACK for previously unacked data	CongWin = CongWin + MSS If (CongWin > Threshold) set state to "Congestion Avoidance"	Resulting in a doubling of CongWin every RTT
Congestion Avoidance (CA)	Received ACK for previously unacked data	CongWin = CongWin+MSS * (MSS/CongWin)	Additive increase, resulting in increase of CongWin by 1 MSS every RTT
SS or CA	Loss event detected by 3 duplicate ACK	Threshold = CongWin/2, CongWin = Threshold, Set state to "Congestion Avoidance"	Fast recovery, implementing multiplicative decrease. CongWin will not drop below 1 MSS.
SS or CA	Timeout	Threshold = CongWin/2, CongWin = 1 MSS, Set state to "Slow Start"	Enter slow start
SS or CA	Duplicate ACK	Increment duplicate ACK count for segment being acked	CongWin and Threshold not changed

- TCP congestion control is referred to as an additive-increase, multiplicative-decrease form of congestion control.
- TCP throughput is measured through cwnd/RTT.



- A congestion control mechanism is said to be fair if the average transmission rate of each connection is approximately R/K, where R is the rate and K is the number of connections.
- Calculating SRTT, DevRTT, and RTO
    - SRTT = SRTT + alpha * (SampleRTT - SRTT)
    - DevRTT = DevRTT + beta * (abs(SampleRTT - SRTT) - DevRTT)
    - RTO = SRTT + 4*DevRTT
    - In case of retransmission, don't update SRTT and DevRTT.
    - Double the retransmission timer value after each timeout.
    - Once you get your first sample RTT, SRTT = sample RTT and DevRTT = SRTT/2
- All in all, transport layer has the main job of multiplexing and demultiplexing data. UDP basically does this, but TCP does a whole lot more in terms of reliability.

Chapter 4 - Network Layer

Network Layer

- Network layer provides host to host communication services.
- In the network layer, packets can get:
    - Lost: Detected by alarm timer in absence of ACK
    - Corrupted: Detected by the checksum
    - Duplicated: In TCP, you just ignore the 2nd one
    - Reordered: Detected by assigned packet with sequence numbers.
- Recovery for all of these is simply retransmitting.
- The network layer has functions of forwarding and routing.
    - Forwarding involves the transfer of a packet from an incoming link to an outgoing link within a single router. A packet arriving from H1 to Route R1 must be forwarded to the next router on a path to H2.

    - Routing involves all of a network's routers, as the packet can take different trips from its source to destination node. The network layer must determine the particular route, and the paths are called routing algorithms.
- Forwarding is the router local action of just moving a packet from input to output, but routing refers to the network wide process that determines the paths the data takes.
- When you have H1 trying to communicate with H2, the network layer in H1 takes segments from the transport layer in H1, encapsulates them into a datagram, and then sends datagrams to a nearby router. H2 will then receive the datagrams from another router and it will extract the transport layer segments, and then deliver them to the transport layer at H2.
- The primary role of routers is to forward datagrams from input links to output links.
- Every router has a forwarding table, where a router will examine the incoming packet, and look at the table to see where it should be routed to.
    - Routing algorithms basically determine how these values in the table get set.
- Packet switch refers to a general packet switching device that transfers a packet from input link interface to output link interface.
    - Link layer switches (layer 2 devices) are a type of packet switch and they base their forwarding decisions on values in the fields of the link layer frame.
    - A router (layer 3 devices) is another type and it bases forwarding decisions based on values in the network layer field.
- A third networking layer function (besides forwarding and routing) is connection setup, which is basically a handshake requirement between routers.
- However, network layer can also provide connectionless service or connection service between two hosts, just like transport layer protocols like UDP and TCP
    - Differences between network and transport layer are that you can have both connectionless and connection services in network layer.
    - Virtual circuit networks = connection service
    - Datagram networks = connectionless service
- A virtual circuit system consists of a path between source and destination host, VC numbers (one for each link along the path), and entries in the forwarding table in each router. Network sets up a connection and then delivers packets over the connection.
- An important note is that the VC numbers are different in each link, and as the packet passes through the links, it will keep track of a VC number field in its header.
- In a VC network, the routers must maintain connection state information for the ongoing connections.
    - Entries must be added and removed from the routers' forwarding tables.
- 3 Phases in a virtual circuit
    - VC Setup: Sending transport layer contacts the network layer, sends address info, network layer determines path between sender and receiver, and VC numbers for the links.
    - Data transfer: Packets actually start to flow along the VC.
    - VC teardown: Sender informs network it wants to terminate the VC. Network layer tells other end system and updates forwarding tables accordingly.

- In a datagram network, one end system will stamp the destination end system address onto the packet and then send it into the network.
    - These networks use a longest prefix matching rule to determine where to route packets and how to setup the forwarding tables.
- Difference between VC and datagram is that a VC network will update forwarding tables very quickly (every time connection statuses change), while datagram networks will update them every 5 min or so.
- Each host is connected to a subnet, which is basically a network of connected hosts, normally having some association with each other.
    - I guess each subnet can get a address block where the first x bytes are for the network ID and then the next (32 - x) blocks are for the specific host IDs.

    - Organizations that subscribe to a specific ISP get a sub block from their ISP's address block.
    - IP subnet is the portion of the address that is considered as network ID by the local site.
    - An address in that network will have the form a.b.c.d/x where x is the number of bits in the network ID portion of the address.
        - The network mask is 2^32 - 2^(32-x)
        - It's basically just x number of 1s followed by (32-x) number of 0s. So, the netmask for 200.23.16.0/23 is 255.255.254.0
        - This subnet mask indicates the portion of the address that is considered the network ID.
- Special Addresses
    - 0.0.0.0/8 is "this network"
    - 255.255.255.255/32 is the broadcast address of "this network"
    - First address of the network is 192.168.1.0 for 192.168.1.0/24 and the last address is 192.168.1.255.
        - First address = network address
        - Last address = broadcast address
- Routing table is needed on every IP host because it is needed to determine where to send the packet.
    - When looking for forwarding table entry for the given destination address, use the longest address prefix that matches the destination address.
    - Forwarding tables in IP routers map each IP prefix to the next hop links.
    - The simplest algorithm is to order the items by length of the network, scan the table one entry at a time, and see if (destination & mask) == (entry & mask)
- When sending something from source A to destination B, you can first check the subnet mask of A and of B to first see whether or not they are on the same net. If (A addr & mask) == (B addr & mask), then they are on the same network, and you can use link

layer protocol to send data to B. Otherwise you send the packet to the normal router, who forwards it to the next hop according to the routing table.

- Network Address Translation (NAT) is the short term solution to the problem of the depletion of IP addresses.
    - It's a way to conserve IP addresses since it can be used to hide a number of hosts behind a single IP address and so those hosts are using private addresses.
    - Basically, there is a single NAT box that has a public IP address and for every packet that comes in, it will determine which of the private hosts in the network it wants to send the packet to. If one of the private hosts wants to send a packet to the outside world, it will send it to the NAT, the NAT will send it out with its own source address, and when the reply arrives, the destination addr will be the NATs address, but when the packet arrives, the NAT will look at a translation table and determine which of the private hosts to redirect the packet towards.
    - Somewhat of a security solution too because private addresses can't be reached from the outside, only if communication is initiated from the inside.
- To summarize, a NAT must replace source IP and port of every outgoing packet to the NAT IP and a new port (This mapping will be stored in a table). Then, for the incoming packets, it will replace the destination NAT IP and port with the corresponding entry in the NAT table so that the info gets to the correct host.
- Problems with NAT
    - Increased complexity
    - Single point of failure
    - Problems with hosts not being visible.
- Solutions for NAT traversal from outside to inside
    - Statically configure NAT to forward connection requests at a given port to server
    - Universal Plug n Play which allows the host behind a NAT to learn the public IP and ass/remove port mappings
    - Relaying
- Two ip networks can be connected to each other through a process called tunneling which is where routers of one network will encapsulate its packets and put all of its packets with the other networks destination address into the tunnel and vice versa. It's a general solution to build a virtual wire across the global internet.
- The Dynamic Host Configuration Protocol is what gives an IP address to a computer. It gives
    - IP address for the host
    - IP address for default router
    - Subnet mask
    - IP address for DNS caching resolver
- Whenever a computer needs an IP, it needs to broadcast a DHCP discovery message and there is some DHCP server that has to respond to it.
- Different subnets may have different MTUs and so if a packet goes from a host in one subnet to a host in another subnet and H1's subnet has a larger MTU than that of H2,

then H2's subnet may have to chop that packet up into different fragments and it'll have to get reassembled at the host.

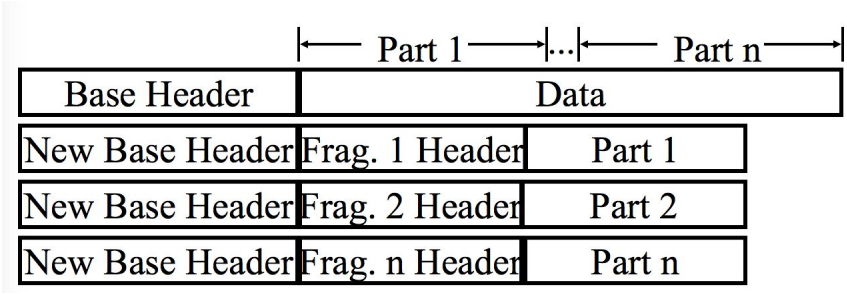
Inside of a Router

- There are 4 components to a router
    - Input ports - Performs physical layer function of terminating an incoming physical link at a router, performs some link layer functions, and performs the lookup function to see where to route the packet to.
    - Switching fabric - Connects the router's input ports to its outputs ports. This fabric is completely contained within the router, so it's like a network inside of a network.
    - Output ports - Store packets received from switching packets and transmits packets on an outgoing link.
    - Routing processor - Executes the routing protocols, maintains routing tables, and does some network management.
- The above all implement the forwarding function, which can be collectively referred to as the router forwarding plane.
- Switching can be accomplished a couple of different ways.
    - Switching via memory: The simplest, earliest routers were computers with switching was done by the CPU.
    - Switching via a bus
    - Switching via an interconnection network
- Packet queues can form at both the input ports and the output ports.

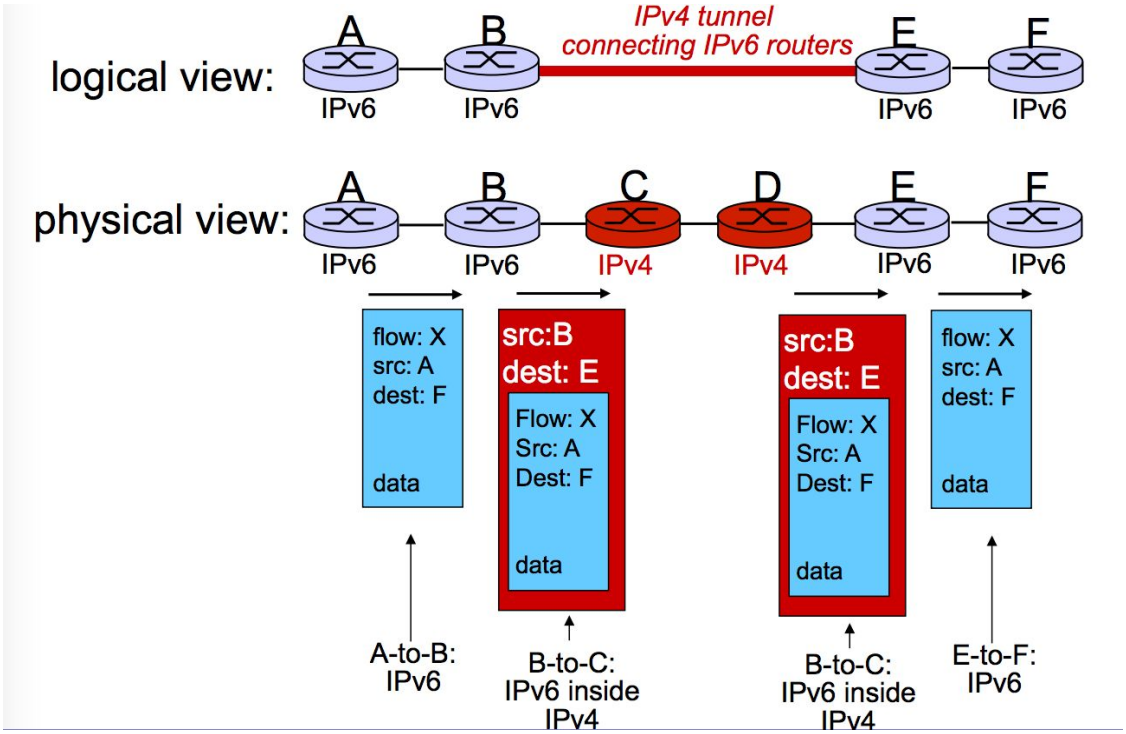
IP (Internet Protocol) and IPv6

- A network layer packet is referred to as a datagram.
- The datagram contains information on version number, header length, type of service, length, identifier, flags, fragmentation offset, time to live, protocol, header checksum, source IP, destination IP, options, and data/payload.
    - An IP datagram has a total of 20 bytes of header, assuming no options.
    - If that datagram carries a TCP segment, then it carries a total of 40 bytes of header (20 bytes for IP header + 20 for TCP header)
- Since the maximum amount of data that a link latere frame can carry is called the maximum transmission unit (MTU), we might need to fragment the data in the IP datagram into smaller datagrams called fragments.
    - This is different from TCP and UDP because those promise complete and unfragmented segments from the network layer, so basically it's the network layer's job to take these fragments and reassemble them before sending them up to the transport layer.
- The boundary between the host and a physical link is called an interface.
- Motivation for IPv6 was 32-bit address space exhaustion.

- IPv6 packets have the following characteristics
    - Fixed length 40 byte header (no more header length and options field)
    - Address length of 128 bits instead of 32 bits.
    - Source and destination addresses are 16 bytes now
    - No more header checksum and fragmentation fields
    - TTL is changed to Hop Limit and Protocol to Next Header
    - Options are outside the basic header, indicated by the Next Field.
    - New field called Flow Label which identifies packets in the same flow.
- Fragmentation in IPv6 requires each fragmented packet to have a fragmentation header.



- When writing IPv6 addresses, you can
    - Skip leading zeros of each word
    - Skip one sequence of zero words (and replace with double colon)
    - Leave the last 32 bits in dot-decimal
    - Specify a prefix by /length
- IPv6 special addresses
    - ::/128 - Unspecified
    - ::1/128 - Loopback
    - ::ffff:0:0/96 - IP4-mapped address
- No broadcast address in IPv6 because the function of it is superseded by multicast.
- You can represent IP addresses in dot-decimal, dot-hexadecimal, dox-octal, decimal, hexadecimal, and octal
- Dual stack is the solution to having the Internet operate with mixed IPv4 and IPv6 routers since all of them can't upgrade at the same time.
- Tunneling is also an option for dealing with routers that have different versions.



- ICMP is the Internet Control Message Protocol
    - It is used by hosts and routers for feedback, status checking, and error reporting.
    - Used for pings and destination unreachable messages
- Messages are carried in IP packets.
- An example ICMP usage is the ping command and another is traceroute.
- Traceroute sends a series of UDP segments (each with an increasing TTL) to destination and when the nth packet arrives to the nth router, the router discards the packet, and sends a “TTL expired” message to the source. Then, when the message arrives, the source calculates the RTT. The stopping criterion is when a UDP packet eventually arrives at the destination host, at which point the destination host returns ICMP port unreachable message.
- The TTL value specifies the max number of hops that the packet can make.

Routing Algorithms

- Routing algorithms operate in network routers and exchange and compute the information used to configure these forwarding tables.
    - Their job is to, given a graph, to find the least cost path from a given node to all the other nodes in the graph.
- A host i attached directly to one router, which is the default router.
- When a host sends a packet, the packet is transferred to the default router.
    - Router on the destination host is the destination router.
- Purpose of a routing algorithm is to, given a set of routers with links connecting those routers, find a “good” path from default to destination.

    - “Good” in most cases means the lowest cost.
- A global routing algorithm is one that computes the least cost path by using complete global knowledges about the network.
    - These are referred to as link state algorithms because they have all of that info.
    - Each node broadcasts link-state packets to all the other nodes in the network so that they all know the topology and the costs.
    - The link state (LS) routing algorithm that is used is based on Dijkstra's because of the fact that every node has complete information, so we can use this method.
        - One of the issues with this algorithm is if you run LS for all the routers at the same time, you could have a situation where all the routers will switch direction at once, causing cost to skyrocket, and then all of them will try to switch again in hope of that 0 cost, but instead they just increase the cost even more.
        - Each router can compute the shortest paths to all destinations.
    - After k iterations of an LS algorithm, a node knows the best paths to k destinations.
- Decentralized routing algorithms are computed in an iterative and distributed manner where no single node has complete information about the costs of the links in the network.
    - The distance vector (DV) algorithm is used.
    - Distributed: Each node receives some information from its neighbors, performs a calculation, and then distributes the results of the calculation back to the neighbors.
        - Basically, each router computes its shortest paths based on the shortest paths of all its neighbors.
    - Iterative: Process continues until no more information is exchanged between the neighbors.
    - Algorithm is asynchronous because the nodes don’t operate in lockstep with each other.

$$D_x(y) = \min_v\{c(x,v) + D_v(y)\} \quad \text{for each node } y \text{ in } N$$

    - A node basically updates whenever it sees a cost change in one of its neighbors or when it receives a distance vector update from some other neighbor.
    - The algorithm is susceptible to routing loops because each node only has local information.
    - Poisoned reverse tries to solve the particular looping problem by setting one of the costs to infinity.
- Split horizon is where a node doesn’t tell a node that is in between it and the destination that it can reach the destination.
- There are a couple of differences between LS and DV

    - Message complexity: LS requires each node to send its information to all of the nodes in the network. DV requires message exchanges between directly connected neighbors at each iteration.
    - Speed of convergence: LS is an n squared algorithm and DV is an algorithm that can converge slowly and can have routing loops and has the count to infinity problem.
    - Robustness: Route calculations are separate in each router for LS so that's good. An incorrect calculation in one node, however, can spread to every node in the network and can cause a lot of problems (in DV).
- Scale and administrative autonomy are two concerns that come with these relatively simplistic routing algorithms.
- To solve both the above problems, routers can be organized into autonomous systems with each AS consisting of a group of routers under the same administrative control.
    - They all run the same routing algorithm and have information about each other.
    - Scale gets solved because an intra-AS router needs to only know about the routers within that AS.
    - Each AS is under the same administrative and technical control and also can contain multiple subnets.
- The routing algorithm running within an autonomous system is called an intra autonomous system routing protocol.
    - Each of the routers inside of an AS know how to forward packets along the optimal path to any destination within the AS.
- Gateway routers have the job of forwarding packets to destinations outside of the AS.
    - AS's intra-AS routing algorithm has determined the least cost path from each internal router to the gateway router.
- The protocols between different AS systems don't need to be the same.
- Inter-AS routing protocols handle the communication between gateway routers in different AS systems.
- Static routing algorithms are ones that change slowly over time, often as a result of human intervention.
- Dynamic routing algorithms change as the topology or the link costs change.
- Load sensitive algorithms have link costs that vary dynamically to reflect the current level of congestion in that link.
    - Load insensitive ones don't care about the current congestion.
- Each router contains a flow table that is computed and distributed by a logically centralized routing controller.
    - It defines the router's match and action rules.
- Routing algorithms must recover from packet losses in routing data delivery, monitor link and neighborhood nodes status, and flush obsolete information out of the system.

Routing in the Internet

- Two routing protocols have been used a lot within an AS in the Internet.
    - Routing Information Protocol (RIP): DV protocol that uses hop count as a cost metric and has a max hop count of 15. Routing updates are exchanged between neighbors approximately every 30 seconds using a RIP response message.
    - Open Shortest Path First (OSPF): LS protocol that uses flooding of link state information and a Dijkstra shortest path algorithm to determine a shortest path tree to all subnets. With OSPF, a router broadcasts routing information to all other routers in the autonomous system. Every node broadcasts a piece of the topology graph and when you assemble the pieces together, you get the complete graph.
- In OSPF, every router has an LSP entry per neighbor router and has its own LSP.
    - When neighboring routers discover each other for the first time, they exchange their link state databases.
    - Neighbors send HELLO message to each other periodically, and not receiving that message for a long time means failure and triggers a new link state update to the neighbors.
        - In the absence of failure, send out an update every 30 minutes.
- LSP flooding is when the node will forward each received new LSP to all neighbor nodes, but the one that sent it.
- In RIP, each router maintains a RIP table known as a routing table which includes the router's distance vector and the router's forwarding table.
- If a router does not hear from its neighbor every 180 seconds, the neighbor is no longer reachable.
- The Internet is made up of a large number of autonomous systems or AS.
    - Stub AS: end user networks
    - Transit AS: Internet service provider
- Intra AS is within a campus or within an ISP
    - Intra domain routing: RIP, OSPF
- Inter AS is between ISPs
    - Intra domain routing: BGP
- In terms of the routers' forwarding tables, intra-AS sets the entries for internal destinations and inter-AS and intra-AS set entries for external destinations.
- An OSPF AS can be configured hierarchically into areas, which run their own OSPF LS algorithm with each router broadcasting to all the other routers in the area.
    - One OSPF area is configured to be the backbone area which has the job of routing traffic between the other areas in the AS.
- Border Gateway Protocol is the standard inter-AS routing protocol in today's Internet.
    - Gives each AS a means to obtain subnet reachability from the neighboring ASs, propagates the reachability information to all routers internal to the AS, and determines good routes to subnets based on the reachability information and on AS policy. It also advertises its own prefixes to the rest of the world.
- In BGP, a pair of routers exchange information over semi permanent TCP connections.
    - The two routers at the end of the connection are called BGP peers.

- A BGP session that spans two ASs is called an external BGP session and a BGP session between routers in the same AS is called an internal BGP session.
    - BGP uses eBGP and iBGP to distribute routes to all the routers within ASs.
    - The route is made of prefix + attributes
- BGP has a routing policy that a provider advertises all prefixes to its customer AS's, and a customer does not advertise prefixes between providers.
- A customer being dual homed means that it is a customer AS attached to two provider networks.
- Intra-AS routing focuses on performance, and inter-AS routing focuses on policy.
- In BGP, an AS is identified by its globally unique AS number (ASN).

Broadcast and Multicast Routing

- Delivering packets at the link layer can be unicast (one receiver), broadcast (everyone receives), and multicast (select groups receive).
- In broadcast routing the network layer provides a service of delivering a packet sent from a source node to all other nodes in the network.
    - (Replicate at source) One approach to implementing this is though N way unicast routing where the source node just makes N copies of the packet to send to N locations.
    - (Replicate at branch points) Another approach is to use flooding, where the source node sends a copy of the packet to all of its neighbors. When a node receives that broadcast packet, it duplicates it, and forwards it to all of its neighbors.
        - The one flaw is that if there are cycles in the graph, then duplicate copies will be made.
        - Another flaw is the broadcast storm where there is an endless multiplication of broadcast packets.
- Multicast routing is being able to send a copy of a packet to a *subset* of the other network nodes.
- In multicast routing, IGMP is a protocol that lets local routers know that there are members in the group. The goal is to establish multicast group memberships.
    - One router is elected the "querier" on each local network and this querier periodically sends Membership Query messages to all-systems group with a TTL of 1. When hosts receive it, they start a random timer for each group they want to join, and when the timer expires, the host membership report to that group (The other members of the group hear this, and stop their timers, so one report message per group is sent in response to a query).
    - A host sends a leave group message to the group if and only if it's the most recent host to report membership in that group.
- Some multicast routing protocols are DVMRP and PIM
- To control broadcast storms, you can use sequence number controlled flooding where each node will maintain a list of the source address and sequence number of each

broadcast packet it has already received. If an incoming packet is on the list, the packet is dropped. If not, it is duplicated and then forwarded.

- Another approach is reverse path forwarding. When a router receives a broadcast packet, it transmits the packet on all its outgoing links only if the packet arrives on the link that is on its own shortest path back to the source. A node N forwards packet from source S if it arrived on shortest path from N to S.
- Both of the above approaches don't completely avoid the transmission of redundant broadcast packets.
- You can also try to create a minimal spanning tree for the whole network, but it's very expensive to maintain and to create.
    - You can create a center based spanning tree where you pick a center node, and each node sends unicast join message towards the center node. The message gets forwarded until it arrives at a node already on the spanning tree.
    - You can also build a tree of shortest path routes from each source to all receivers using Dijkstra's algo.
- In multicast communication the problems are how to identify the receivers of the multicast packet and how to address a packet sent to these receivers.
    - In order to determine where to send the packets (and more importantly how to store that destination address information), packets will use address indirection where a single identifier is used for the group of receivers, and a copy of the packet that is addressed to the group using this identifier is delivered to all of the multicast receivers associated with that group.

Chapter 5 - The Link Layer

Link Layer

- The link layer focuses on how packets are sent through the individual links that make up the connection path.
    - It transfers one node to a physically adjacent node over a link.
    - IP packets are encapsalated into layer 2 frames.
- Any device that runs a link layer protocol is defined as a node.
    - Nodes can be hosts, routers, switches, etc.
- Over a given link, the transmitting node encapsulates the datagram in a link layer frame and transmits the frame into the link.
- Link layer protocol can provide the services of framing, link access, reliable delivery, and error detection/correction.
- The link layer is implemented in a network adapter, or a network interface card.
    - These adapters have addresses. Link layer addresses and referred to as MAC addresses.
- The link layer controller is inside of the adapter and it implements many of the link layer services.

- On the sending side, the controller will take a datagram from memory, encapsulate it in a link layer frame, and then transmit the frame into the communication link, which follows a particular link access protocol.
    - On the receiving side, the controller can do some error checking before extracting the packet and transferring it to the network layer.
- Byte Oriented Framing Protocol says to delineate each frame with a byte of special bit sequence 01111110
- In case of the situation where that special sequence occurs in the data, we can do some byte stuffing to prevent confusion.
    - HDLC Byte Stuffing
        - Frames start and end with 01111110
        - Sender stuffs extra 0 bit after it sees a sequence of 5 bits.
        - On receipt, the receiver will look through the segment that it gets, and removes this bit (unless there are 6 1 bits which means a frame boundary).
    - PPP Byte Stuffing
        - Replace 0x7e with 0x7d 0x5e
        - Replace 0x7d with 0x7d 0x5d
        - Frames start and end with 0x7e
    - COBS
        - Start off with 1 + #non-zero bytes that follow
        - All the zero bytes are changed to 1 + #non-zero bytes that follow
        - Everything ends with 00
- Bit level error detection and correction is detecting and correcting the corruption of bits in a link layer frame sent from one node to another physically connected node.
    - The simplest form of error detection is the use of the parity bit, which is basically a number equal to the total number of 1s in the bits that are even.
    - A 2-D method is also used when instead of just having a long string of bits, you created a 2-D of all the bits arranged in row/col form.
- The ability of the receiver to both detect and correct errors is known as forward error correction.
- Another error detection method is using cyclic redundancy check codes.
- There are two types of network links, point to point links and broadcast links.
    - Point to point links consist of a single sender and a single receiver.
    - Broadcast links can have multiple sending and receiving nodes all connected to the same single shared broadcast channel.
        - The problem with this is called the multiple access problem which refers to coordinating the access of multiple sending and receiving nodes to a shared broadcast channel.
- Traditional TV is an example of one way broadcasting where you have a single source node that is transmitting to many receiving nodes.
- Multiple access protocols force nodes to regulate their transmission into the shared broadcast channel.

- A collision is where two source nodes broadcast at the same time causing all of the receiving nodes to receive multiple frames at the same time.
- All multiple access protocols can be classified as channel partitioning protocols, random access protocols, and taking turns protocols.
    - Channel partitioning protocols rely on the idea that nodes get dedicated transmission rates of R/N and nodes must wait for their turn in the transmission queue. Time division multiplexing, frequency division multiplexing, and code division multiple access are the 3 channel partitioning protocols.
    - Random access protocols have the transmitting nodes always transmit at the full rate R of the channel, and when there is a collision each node repeatedly retransmits its packet (after waiting a random delay) until it gets through without a collision. (Slotted) ALOHA and CSMA (CA and CD) are the two protocols in this group.
        - Aloha is basically where if a node has data to send, it will send the whole frame immediately. If there is a collision, it will retransmit the frame again with a probability of p. Probability of success of transmitting is basically P(node transmits) * P(no other node transmits in the time period)
        - Slotted Aloha is where there are slots, and nodes transmit in those slots.
        - In CSMA CD, it receives datagram from network layer, creates the frame, starts transmission if channel is idle, transmits, and if it detects another transmission from another node, it will abort and send a jam signal (makes all other transmitters aware of the collision), and then enter binary exponential backoff (if first collision choose delay of {0,1} slots, if second collision choose delay of {0,1,2,3} slots, if tenth collision choose delay of {0,1,2,3...1023} slots).
    - Polling protocols (a subset of taking turns protocols) is where there is one required master node which polls each of the nodes in a round robin fashion to tell them that they can transmit up to some number of packets/frames. The good part is that it eliminates collisions and empty slots. The bad part is that there is a polling delay where the master has to notify the other node to start sending and if the master node fails, then everything goes down as well.
- Types of persistence
    - 1-persistent is where when the node is ready to transmit and the channel is idle, it transmits immediately. If busy, it continually waits until it's idle and then sends. If there is a collision, then it waits random amount of time until it tries again. 1-persistent is used in CSMA/CD systems (Ethernet).
    - Non-persistent is where when the node is ready to transmit and the channel is idle, it transmits immediately. Then, waits random amount of time until it tries again.
    - P-persistent is where when the node is ready to transmit and the channel is idle, it transmits a frame with probability p. If busy, waits till the next time slot. P-persistent is used in CSMA/CA systems (WiFi).
- Carrier sensing is when a node listens to the channel before transmitting.

- Collision detection is when a transmitting node listen to the channel while it is transmitting.
- The channel propagation delay of a broadcast channel is the time it takes for a signal to propagate from one of the nodes to another.

Switched Local Area Networks

- The adapters in all hosts and routers have link layer addresses, which are called LAN addresses, physical addresses, or MAC addresses.
- An adapter's MAC address has a flat structure and doesn't change no matter where the adapter goes.
- When a sending adapter does want all the other adapters on the LAN to receive and process the frame it's going to send, then the adapter inserts a MAC broadcast address into the destination address field.
    - Link layer basically uses MAC addresses to deliver packets to destinations.
- Translating between network layer addresses and link layer addresses is the job of the Address Resolution Protocol (ARP).
- ARP packets are used to query all the other hosts and routers on the subnet to determine the MAC address corresponding to the IP address that is being resolved.
    - ARP packets can be encapsulated within a link layer frame packet.
- Basically, when A knows B's IP address, it needs to also know its MAC address. It will broadcast an ARP query message containing B's IP address and a destination MAC of FF-FF-FF-FF-FF-FF. B receives this query and replies with its MAC address.
- Each router and host has an ARP table in its memory, which contains the mappings of IPs to MAC addresses.
    - Each table entry represents a node on the LAN
        - < IP address, MAC address, TTL >
- Ethernet technologies seek to provide connectionless service to the network layer so that when adapter A wants to send a datagram to adapter B, then A will encapsulate the datagram in an Ethernet frame and send the frame into the LAN, without first handshaking with B.
    - Ethernet is both connectionless and unreliable because there are no acks or nacks sent between the two adapters.
- A hub is a link layer device that acts on individual bits rather than frames. Packets come in one port/link, and go out all the other ports/links.
- Repeaters are physical layer devices where bits coming in one link go out all other links at the same rate.
- The role of an Ethernet switch is to receive incoming link layer frames and forward them onto outgoing links. The switch is transparent to the hosts and the routers in the subnet. They are unaware of it.
    - These switches need forwarding tables so that they know whether nodes are reachable or not. Each switch has a switch table where each entry is
        - < MAC address of host interface to reach host, time stamp >

    - The tables are built by self learning. When a switch gets a data frame, if it's not already in the table, then record the link and MAC address of the sending host. If a table entry is found for the destination, then forward it to the corresponding interface and if it's not found, then forward to all interfaces except the arriving interface.
- Filtering is the function that determines whether a frame should be forwarded or just droped.
- Forwarding determines the interfaces to which a frame should be directed, and then moves the frame to those interfaces.
- Both filtering and forwarding are done with a switch table.
- Switches are better than broadcast links because they eliminate collisions, can have heterogeneous links, and provide easy network management and security.
- Switches and routers are both store and forward devices.
    - Routers are in the network later and they examine IP headers. They support arbitrary topologies, have efficient support for multicast routing, and require IP address configuration.
    - Switches are in the link layer and they examine Ethernet headers. They are transparent, they isolate collision domains, can connect Ethernets of different speeds, and have a constrained topology.
    - Switches do well in a small setting, and routers are used in large networks.
    - To build their respective forwarding tables, routers run routing protocols and switches implement self learning algorithms.

Chapter 6 - Wireless and Mobile Networks

Wireless Networks

- The elements of a wireless network are wireless hosts (the devices that run applications), wireless communication links (the way the host connects to a base station or to another wireless host), and a base station which is responsible for sending and receiving data to and from a wireless host that is associated with that base station.
    - Cell towers and access points in wireless LANs are examples of base stations.
- A wireless host is associated with a base station when the host is within the wireless communication distance of the base station and the host uses that base station to relay data between the host and the larger network.
    - When the host is associated with a base station, it is operating in infrastructure mode, since all the traditional network services are provided by the network to which a host is connected.
    - In the cases where the host doesn’t have the associated base station, the hosts themselves need to do routing, address assignment, name translation, etc. This is ad hoc mode. Nodes can only transmit to other nodes within link coverage. The

nodes organize themselves into networks and exchange information about who can reach whom.

- The two main challenges of wireless communication are the fact that it is wireless (handling communication over that type of link) and that it is mobile (handling the mobile user who changes point of attachment to network).
    - A wireless link has a decreased signal strength, has to deal with interference signals from other sources, and has multipath propagation.
- When a mobile host moves beyond the range of one base station and into the range of another, it needs to change its point of attachment into the larger network. This process is called a handoff.
- Wireless networks are characterized according to whether packets in the network coss exactly one wireless hop or multiple wireless hops and whether there is infrastructure such as a base station in the network.
- The difference between a wired link and a wireless link is that with a wireless connection there is decreasing signal strength and interference from other sources (those that transmit with the same frequency) and multipath propagation were portions of the EM wave reflect off objects and the ground, causing a blurring of the signal at the receiver.
- From the receiving host's perspective, the signal to noise ratio is a relative measure of the strength of the received signal and background noise created by the environment.
    - SNR measured in decibels.
    - Larger SNR means it's easier for the receiver to extract the transmitted signal from the noise.
- The bit error rate is the probability that a transmitted bit is received in error at the receiver.
- For a given modulation scheme, the higher the SNR, the lower the BER.
    - A sender can increase SNR by increasing its transmission power. The advantage is that it lowers BER, but the disadvantage is that more energy must be expended by the sender, and the transmissions are more likely to interfere with that of another sender.
- For a given SNR, a modulation technique with a higher bit transmission rate will have a higher BER.
- SNR and BER are likely to change as a result of mobility or due to changes in the environment.
- The higher and time varying BER is a characteristic of wireless links.
- Hidden terminal problem is when nodes A and C are transmitting to B, but since there is a physical object in between A and B, it might prevent A and C from hearing each other's transmissions, which are likely causing some sort of interference.
- Fading is when the signals for A and C are not strong enough to detect each other, but they are strong enough to interfere with each other at B.
- CDMA is a medium access protocol that makes sure that signals sent by multiple senders don't interfere at the receivers. It is used a lot in the wireless world.

Wifi

- IEEE 802.11 wireless LAN is known as WiFi.
- There are 3 802.11 standards for wireless LAN technology They all use medium access protocol, CSMA/CA. They all use the same frame structure for their link layer frames, and they all have the ability to reduce their transmission rate in order to reach out over greater distances. They also allow for infrastructure and ad hoc modes.

Standard	Frequency Range (United States)	Data Rate
802.11b	2.4–2.485 GHz	up to 11 Mbps
802.11a	5.1–5.8 GHz	up to 54 Mbps
802.11g	2.4–2.485 GHz	up to 54 Mbps

- 802.11a LANs have a shorter transmission distance for a given power level and suffer more from multipath propagation.
- The building block of the 802.11 architecture is the basic service set (BSS), which contains one or more wireless stations and a central base station known as an access point.
    - The network administrator chooses a frequency for a specific AP.
- Each 802.11 wireless station has a MAX address stored in the adapter.
- Wireless LANs that deploy APs are referred to as infrastructure wireless LANs where the APs are the infrastructure and the wired Ethernet connections connect the APs and a router.
- The stations can group themselves together to form an ad hoc network which is a network with no central control.
- Each wireless station needs to associate with an AP before it can send or receive network layer data.
- Installing an AP means that there is a one or two word Service Set Identifier known as an SSID.
- A WiFi jungle is any physical location where a wireless station receives a sufficiently strong signal from two or more APs.
- For a device to get internet access, the station needs to join one of the subnets and needs to associate with exactly one of the APs. This creates virtual wire between itself and the AP.
    - Then it might need to authenticate itself to the the AP. This could be based on a MAC address or through usernames and passwords.
- In order for the AP to communicate with all the devices out there, it periodically sends beacon frames that include the SSID and MAC address.
- The process of scanning channels and listening for beacon frames is known as passive scanning. This is something the APs send.

    - Active scanning is when the device broadcasts a probe frame that is received by all of the APs. This is something the hosts send.
- From the receiver's side, there will be a lot of senders and thus multiple access protocols will need to be used.
    - 802.11 uses random access protocol referred to as CSMA with collision avoidance.
        - We can't use collision detection because there are often weak received signals that are difficult to receive while transmitting, and we can't sense all collisions because of the hidden terminal case.
        - The station first needs to sense the idle channel, transmit a frame after Distributed Interframe Space (DIFS). If it's busy at that time, choose a backoff value, count down while the channel is idle, and then transmit the entire frame and wait for an acknowledgment. If the receiver doesn't get it, the frame is retransmitted with a larger interval for the backoff value.
        - From the receiver's perspective, once it gets a frame, then it will send an ACK after SIFS.
    - Basically, each station will sense the channel before transmitting and will refrain from transmitting with the channel is busy.
- Difference between Ethernet and 802.11 is that 802.11 uses collision avoidance while Ethernet uses collision detection. 802.11 also uses link layer ACK/retransmission scheme because of the high BER cause by the fact this is a wireless network.
    - 802.11 doesn't do collision detection because that would require the ability to send and receive at the same time which isn't really possible given the strengths of sent and received signals in wireless connections.
- Once a station in 802.11 wireless LAN begins to transmit a frame, it does so in its entirety.
- 802.11 uses link layer acknowledgment which says that when a destination station gets a frame, it waits a period of time called Short Interframe Spacing (SIFS) and then sends a ACK frame. It the sender doesn't get this frame within a period of time, it will retransmit.
- One difference between collision detection and avoidance is that in detection, the station begins transmitting as soon as the channel is idle. With collision avoidance, the station doesn't transmit until the countdown is finished.
    - In detection, the idea is that we want to transmit as quickly as possible and we're okay with collisions because if we detect one, we can just abort the transmission.
    - In avoidance, we can't detect collisions and we don't abort and we transmit frames in their entirety. Thus, we want to avoid collisions as much as possible. The backoff counters make sure of this. If the values are different, one station will start transmitting, the other will freeze its counter, and then transmit once the other station is done.
- RTS/CTS exchange (also called optimal collision reduction) is mainly used to reserve the channel for the transmission of a long data frame.

- 802.11 fights the hidden terminal problem by allowing stations to send request to send and clear to send control frames in order to reserve access to the channel.
    - If a station requests an RTS, then the AP can send back a CTS and then instruct all the other stations to not send.
- 802.11 Frames are made up of a payload, which is an IP datagram or an ARP packet, and a 32 bit cyclic redundancy check.
    - The frame also has 4 address fields, where Address 1 is the MAC address of the wireless station that receives the frame, Address 2 is the MAC address of the station that transmits the frame, and Address 3 contains the MAC address of the router interface.
- Companies and universities will often deploy multiple BSSs within the same IP subnet to increase the physical range of a wireless LAN.
    - When a host moves from BSS 1 to BSS 2 and if the host is not a router, it can keep the same IP address and maintain its TCP connections. As the host moves away from AP1, it will scan for a stronger signal, and receive beacon frames from AP2, disassociate with AP1, and associate with AP2 while keeping IP address and maintaining TCP connections.
    - If it is a router, then it needs a new IP address in the subnet in which it is moving, and connections would have to be terminated.
- 802.11 also has advanced features like rate adaptation and power management.
- A mobile has a home agent and a permanent home IP address. When a mobile moves to a new location, it obtains a new care of address and informs its home agent of its new IP address.
- Every host has a home, and when the mobile moves outside home, it gets a foreign IP and connects to a foreign agent which communicates with the home agent. Other hosts can then contact the home agent to learn where the mobile is.
    - Indirect routing: Others send packets to home agent, which forwards to mobile. The home agent does this by tunnelling the packet to the foreign agent by encapsulating the packet that we originally got. The permanent address is used by the correspondent to send packet to mobile. The care of address is used by home agent to forward packet to mobile.
    - Direct routing: Others learn mobile's foreign address and directly send packets to mobile. Mobile can get the care of address from a foreign DHCP server.
- When the mobile moves to another network, it registers with a new foreign agent, the new foreign agent registers with the home agent, and the home agent updates the care of address for the mobile.
- Cellular networks are composed of cells and mobile switching centers.
    - Cells cover geographical regions and they have a base station, and mobile users attach to network through the base station.
    - The mobile switching center connects cells to wide area network and handles mobility.
- Bluetooth and Zigbee are two personal area networks.

- Bluetooth is an 802.15.1 network that operates over a short range, at low power and cost. They are ad hoc networks.
- Zigbee is also 802.15 and it's targeted at lower powered, lower data rate, and lower duty cycle applications than Bluetooth.
- Over 80% of cellular subscribers use GSM (Global System for Mobile Communications) standards.
- Cellular networks are partitioned into cells (geographical areas), where each cell contains a base transceiver stations that transmits signals to and receives signals from the mobile stations in its cell.
    - Coverage area of this cell depends on transmitting power of BTS, power of user devices, physical obstacles, and height of base station antennas.
- A GSM network also has a base station controller (BSC) which services several base transceiver stations.
    - Its job is to allocate BTS radio channels to mobile subscribers and perform paging (finding the cell in which a mobile user is resident) and perform handoff of mobile users.
- The home network is the network of cellular provider you subscribe to. The visited network is the network in which the mobile currently resides
- The mobile switching center (MSC) plays the central role in user authorization and accounting.
- When trying to switch base stations while in a car, the handoff is initiated by the old BSS.
- The motivation behind VPNs are that institutions often want private networks for security.
- IPsec is a network protocol suite that authenticates and encrypts packets of data sent over a network. The two protocols are
    - Authentication Header protocol is where it provides source authentication and data integrity but not confidentiality.
    - Encapsulation Security Protocol is where it provides source authentication, data integrity, and confidentiality.
- 3G data services leave the existing core GSM cellular voice network untouched, and adds additional cellular data functionality in parallel to the existing network.
- Two types of nodes in 3G core network are serving GPRS Support Nodes (SGNs) and Gateway GPRS Support Nodes (GGSNs).
    - SGSN delivers datagrams to and from mobile nodes in the network to which the SGSN is attached. It provides user authorization and handoff, maintaining the cell information.
    - GGSN connects multiple SGSNs into the larger Internet.
- The 3G radio access network is the wireless first hop network that we see as a 3G user.
- Three approaches to privacy are
    - WEP (Wired Equivalent Privacy) which uses RC4 encryption
    - 802.1x Access control which is a general purpose network access control mechanism.
    - WPA (Wireless Protected Access) which uses ^ and TKIP encryption.

- Man in the middle attacks are where the attacker can terminate the victim's SSL/TSL session at her host and reconnect to the actual site.

Chapter 8 - Security in Computer Networks

- Email systems should be able to provide confidentiality, secure authentication, message integrity, and receive authentication.
- For confidentiality, both parties can use symmetric key technology so that if both parties have the key, nobody else would be able to read the message.
    - Keys don't, however, work for long sessions. In these cases, you'd need session keys.

Other Stuff

- Stream sockets use TCP
- Datagram sockets use UDP
- Application Layer Protocols - HTTP (Hyper-Text Transfer Protocol), HTTPS (Hyper-Text Transfer Protocol), SMTP (Simple Mail Transport Protocol), DNS (Domain Name System)
- Transport Layer Protocols - TCP (Transport Layer Protocol), UDP (User Datagram Protocol)

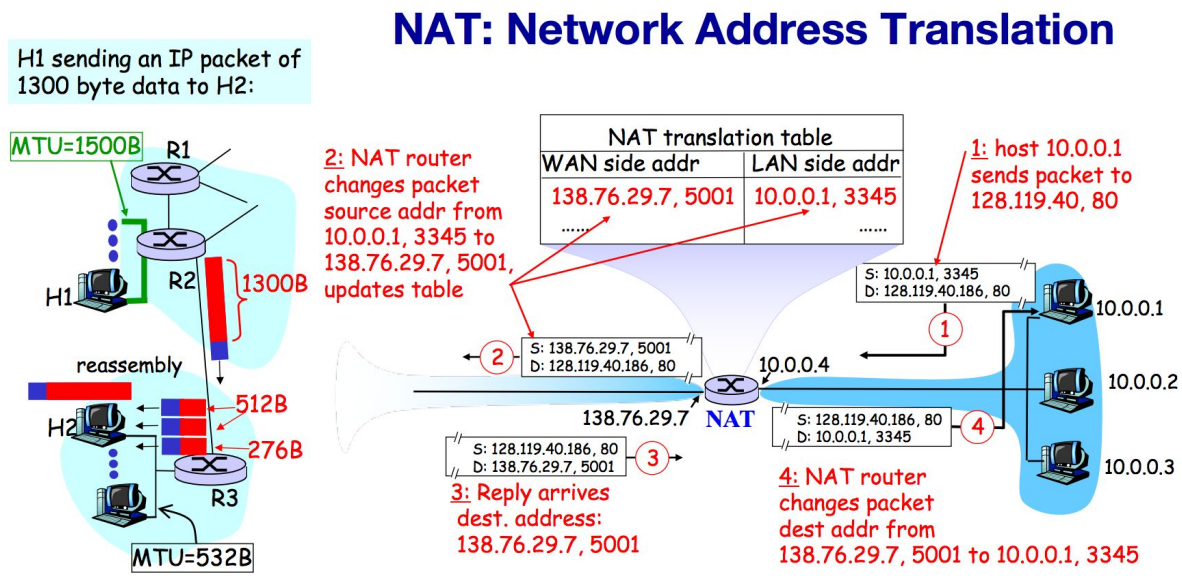
Big Picture

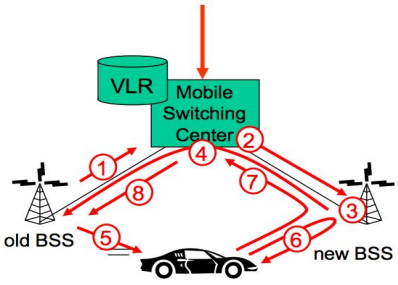
- Application protocols assume the network provides a way to send data to hosts on the internet (basically send data between processes of different end systems), don't care how the data is sent, care whether it gets there reliably, and runs on top of transport protocols that take care of how the data is sent.
- Transport protocols assume that application protocols take care of data content, so that transport protocols just have to deliver data between communicating ends, and they do care about delivering to the right receiver, reliability of the delivery, and congestion control.
- Network protocols are the ones that handle the actual delivery of the packets. Their job is to forward data from the source to destination.
- Link Layer protocols define how to get a particular packet sent across a medium (transferring data between directly connected network elements).
- Physical: Bits "on the wire"

NDN Notes

- The classic IP model is based on host to host communication. NDN seeks to instead have a model where you just have hosts request for information (not sending the request to anyone in particular), but having that data get returned to you by someone else in the cloud.

Helpful Images





1. old BSS informs MSC of impending handoff, provides list of 1+ new BSSs
2. MSC sets up path (allocates resources) to new BSS
3. new BSS allocates radio channel for use by mobile
4. new BSS signals MSC, old BSS: ready
5. old BSS tells mobile: perform handoff to new BSS
6. mobile, new BSS signal to activate new channel
7. mobile signals via new BSS to MSC: handoff complete. MSC reroutes call

8 MSC-old-BSS resources released