12/1:

What is a network?

Collection of node and links that connect them.

TV, transportation, sewage. Think of the links and nodes for each of these.

How are computer networks different from other types of networks? Transportation in a city. Stations are the nodes, with train tracks being the links. Trains would be packets with the data in it.

Key computer network actors:

* Users. They interact with networks through application.
* App Developers create network applications. Worried about getting that shit working.
* Administrators. Operate and manage networks. Grow overlap between users and admins. Interested in the different characteristics of the network
* Designer. Design and build network devices and rules for communication.

Application of computer network:

* World Wide Web (WWW)
  + Email clients
  + Social networks
  + Just the links of all servers together. Does not exist as this magical cloud.
  + Internet “killer app”
    - More precisely – suite (platform) of applications
* Audio / streaming services
  + Instant messaging
  + File sharing
* Online gaming

Structure of URL example:

* 17 messgaes are involved to process single page (object) request:
  + 6 messages to translate the server name into IP address;
    - Cose.latech.edu -> 138.47.28.18
  + 3 messages to set up TCP connection;
  + 4 messages for HTTP request and acknowledgment;
  + 4 messages to tear down TCP connection.

Applications: Audio/video:

* Streaming audio/video
  + Video on demand, internet radio.
  + Delivery of streaming conent is different from fetching a webpage or an object.
* Real-time audio/video
  + Telecommunication, VolP.
  + Delivery of real-time content is different from precuessing of streaming data.
  + Zoom, skype, discord video
* Diversity of requirements drives how networks support different typers of appications.

General requirements

* Scalability
  + Adding more nodes to network
  + Node addressing and messages routing
* Efficienty
  + All nodes sharing the network.
  + Several nodes sharing a link
* Support of services
  + App-to-app communication through channels
  + Reliablilty issues
* Manageability
  + Automating network management
  + Stability vs. feature velocity

Foundations

Direct Links:

* What is a computer network
  + Group of computer ststems that are linked together through communication channels
* Computer systems – nodes
  + End poins
    - Hosts and servers
  + Redistribution points
    - Hubs, switches, routers
* Communicational channels – links
  + Wired links
  + Wireless links

Indirect links

* Switched network and networks of networks (internetwork) can be arranges with indirect links

Classification by scale

* Networks are frequently classified by their scale:
  + Wired
    - Local aera network (LAN) – around one specific router
    - Metropolitan area network (MAN) – covers sections of a city
    - Wide area network (WAN) – largest area covered. Fiber optic more common
  + Wireless
    - System networks – Bluetooth, RFID, infrared
    - Wireless LAN (WLAN) – Wifi tech
    - Wireless WAN (WWAN) – 4G, 5G
  + Internet

Concept of internet

* Internet is a network of networks
  + Billions of hosts (nodes) and communication links
    - Data is being transmitted by routers and switches
  + Protocols control sending and receiving of messages
    - TCP, UDP, IP, HTTP
  + Standards govern protocols operations
    - Request for comments (RFC)
  + Internet is an infrastructure for network applications
    - Provides services and rules of how to use them
      * Services – communication between applications
      * Rules – application programming interfaces (APIs)
* End nodes connect to interne via access internet service providers (ISPs)
* Access ISPs must be interconnected so hosts can exchange data
  + Connecting access ISPs to each other is not feasible
* Solution – regional, global and Tier-1 ISPs, internet exchange points (IXP), and content providers form complex “Internet Hierarchy”

Protocols

* Protocol (set of rules) defines:
  + Format of network messages
  + Order of messages sent and received among nodes
  + Actions taken on message transmission and receipt

Network edge: Components

* Components ar the network edge:
  + Hosts (end Points)
    - Client
    - Server
  + Access networks
    - Wired/wirless

12/5:

Network core: mesh of interconnected routers

Data can be moved through the network in two ways: Circuit or packet switching. Packet switching more common.

* Packet switching
* Hosts exchange messages with each other
* Messages are broken down into packets
  + Packet size = L bits
* Packets travel through communication links and routers/switches
* Communication links pass packets at a transmission rate (throughput or bandwidth)
  + Transmission rate = R bits/sec
* Packet transmission rate = L/R sec
  + Time needed to transmit L-bit packet through communication link.
* Packet Switching: Store-and-Forward
* Store-and-Forward transmission
  + Switch/router must receive entre packet before it can transmit first bit onto outbound link
    - Bits of a packet are sorted (“Buffered”) by the switch/router
* Delay and Packet loss
  + If arrival rate to link exceeds transmission rate of link for a period of time:
    - Packets will queue and wait to be transmitted on link (queueing delay)
    - Packet will be dropped (packet loss) if memory (buffer) fills up
* Routing and forwarding are the key network-core functions
  + Routing determines source-destination rout taken by packets
    - Generates routing/forwarding table
* Forwarding moves packets from router input link to appropriate router output link
* Hosts share communication links using multiplexing/de-multiplexing process
  + Time division multiplexing
  + Frequency division multiplexing
  + Statistical multiplexing
* Statistical Multiplexing:
  + Cost-effective way fro multiple hosts to share a communication link
  + Allow host to share link over time
* Newtork performance is determined by following measures:
  + Delay (laterncy)
    - How long it takes a message to travel from one end of the network to another
    - Round-trip time (RTT) is frequently used insead of one-way delay
  + Throughput (bandwidth)
    - Amount of data per second that can be transferred by communication link

Day 4:

* Network mask is used to split IP address into a network prefix and host identifier.
  + By applying and operation on IP address
* Originally, UP addresses were divided into classes:
  + Class A:
    - 8 Bit for network prefix, 24 bit for host identidier.
    - Address range: 0.0.0.0 – 127.255.255.255
    - Network mask: 255.0.0.0
  + Class B:
    - 16 bit and 16 bit
    - Address range: 128.0.0.0 – 191.255.255.255
    - Network mask: 255.255.0.0
  + Class C:
    - 24 bit and 8 bit
    - Address range: 192.0.0.0 – 223.255.255.255
    - Network mask 255.255.255.0
* Subnetting – process of dividing large blocks of addresses into several contiguous subblocks to create smaller physical networks
  + Each subblock is a subnet
* Subnet mask is used ot allow sharing single network number among multiple hosts
  + All hosts on the same physical network share same subnet number
* Example:
  + Subnet 223.1.1.0/27
  + Subnet index: /27
    - Indicates that left 27 bits define subnet address
  + Subnet mask: 11111111 11111111 1111111 11100000
    - * 255.255.255.224
* Classless interdomain routing (CIDR) is a current internet addressing method
  + Generalizes notion of subnet addressing
    - Allows arbitrary length subnet address
  + Address bormat a.b.c.d/x
    - A,b,c,d – octets of IP address
    - /x – number of bits in subnet portion
      * Network prefix
  + Example
    - 200.23.26.0/23
    - Maximum number of hosts under subnet = 2^n – 2
* Architectures: Layering
  + Airline system
  + Computer network architectures are organized in layers
    - Each layer is composed of protocols
      * Each protocol belongs to a single layer
  + Protocols at each level serve two main functions:
    - Provide communication services between layers
    - Provide common functionality with the layer
      * Layers on different machines can process same message
  + No direct layer-to-layer communication
    - Protocol passes message down to the physical layer
    - Physical layer protocols can send messages directly.
  + Collectively protocols of various layers are called protocol stack
    - Also refereed as network model or network architecture
  + Main computer network protocol stacks
    - Internet (TCP/IP) architecture
    - Open systems interconnection (OSI) architecture
* Internet Model: Internet (TCP/IP) model layers:
  + Application layer:
    - Protocols: HTTP, SMTP, DNS, FTP
    - Unit of data: nessage
  + Transport layer. Protocols: TCP, UDP
    - Unit of data: segment
  + Network layer
    - Protocols: IP
    - Unit of data datagram
  + Link layer
    - Protocols: Ethernet, Wifi, DOCSIS
    - Unit of data: frame
  + Physical layer
    - Protocol: copper wire, fiber optics, radio waves
    - Units of data: bits
  + Procol graph view of internet model
  + IP protocol is a focal point
    - Defines a common method for exchanging packets among a wide collection of networks
* OSI model:
  + Open systems interconnection architecture
    - Pre internet 7 layer model
  + Two additional layers
    - Presentation layer
      * Interpret the meaning of the exchanged data
        + Compression, encryption
    - Session layer
      * Delimiting and synchronization of data exchange.

12/12/2022

* Application
  + Foundation
    - Architectures
    - Processes
    - Services
    - Protocols
  + World wide web
    - http protocol
    - connection types
    - http message format
    - cookies
    - caching
* intro
  + communication for a network application takes place between end systems at the application layer
* architecture
  + network application
    - client sever arch
  + peer to peer
    - clients host their own version of the app
  + hybrid
    - texting and messaging apps
* processes
  + application layer provides process to process communication by allowing message exchange
    - process – part of a program that is running within the end system
  + network app consists of pairs of process that send messages to each other
    - client – process that initiates the communication
    - server – process that wait to be contacted
  + two pieces of information beeded to identify a process:
    - Ip address of the host
    - port number of the process
      * identifier that specifies the process on the host.
  + Message sent from one process to another must go through underlying network
  + Socket – sotware interface that is used by process to send message into and receive message from the network
    - App programming interface (API) between the application and the network
* Services
  + Two transport layer protocols provide services to applications:
    - Transmission control protocol
      * Connection oriented
      * Reliable data transger
      * Flow control
      * Congestion control
    - User datagram protocol
      * Lightweight
      * Connectionless
      * Unreliable data transfer
* Protocols
  + Application layer protocols define
    - Type of messages exchanged
      * Request messages / response messages
    - Syntax of various messages
      * Fields in messages and how they are delineated
    - Semantics of the message fields
      * Meaning of the information in the fields
    - Rules on how process sends messages and responds to messages
  + Network applitaiton != app layer protocol
    - World wide web app vs http protocol
* Web
  + Overview of web components:
    - Web page (document) consists of objects
      * Objects are files
    - Web pages consist of base html file

12/14/22

* Internet is 30 years old
* Webpage consists of base HTML-files, which include several referenced objects
* Web browsers request data
* Web server store the data
* HTTP (HyperText Transfer Protocol) – Web’s application- layer protocol
* Web browser – client side of HTTP
* Web server – server side of HTTP
* Http runs on TcP transport-layer protocol
  + Client initiates TCP connection to the server through socket
    - Port number 80
  + Server accepts tcp connection from client
  + http messages exchanged between browser and server
  + tcp connection closed
* http is stateless protocol
  + server maintains no information about past client requests
* http provides two trypes of connectinos
  + nonpersistent
    - at most one object sent over tcp connection
    - pings 5 times per object, making 10 objects take 50 exchanges
  + persistent
    - multiple objects can be sent at a time
    - used by default
* two types of http messages
  + request message
    - contains request and header lines
  + response messages
    - contains status, header, and entity body (data) lines
* 200 OK
  + Request succeeded
* 301 Moved Permanently
  + Requested object moved, new location specified
* 400 Bad Request
  + Request message is not understood by server
* 404 Not Found
  + Requested doc does not exist on this server
* 505 HTTP Version error
* Cookies are used by websites to keep track of users
* Four main components
  + Cookie header line in response message
  + Cookie header line in request
  + Cookie file on client
  + Back-end database at website
* Cookies are controversial since they can potentially invade privacy

12/16/22: DNS

* Domain Name System
  + DNS services, operation, caching
  + Resource records
  + Messages
  + Directory service that translates hostnames into IP addresses
    - Latech.edu -> 138.47.18.212
    - Only responsible for this process
  + DNS consists of two parts:
    - Distributed database implemented in a hierarchy of DNS servers
    - Application-layer protocol that allows hosts to query distributed database
  + DNS protocol runs over UDP and uses port 53
  + DNS commonly employed by other application-layer protocols
  + Example
    - HTTP client to request [www.someschool.edu/index.html](http://www.someschool.edu/index.html) is must obtain IP addresss of someschool.edu to send an HTTP request message
  + Additional DNS services
    - Host aliasing
      * Maps additional hostnames to canonical hostname
    - Mail server aliasing
      * Provides canonical hostname for supplying alias hostnames in mail application
    - Load distribution
      * Distributes the load among replicated web servers
      * A set of IP addresses is associated with one canonical hostname
      * IP addresses are rotated within each reply
  + DNS is implemented as a distributed hierarchical database
    - Centralized design would not scale to the needs of modern internet
      * Single point of failure
      * Traffic volume
      * Distant centralized database
      * Maintenance
  + Three\* classes of DNS servers
    - Root DNS servers
      * Top of the server hierarchy
      * 13 root DNS severs (A-M)
      * Each server is a network of replicated servers (security and reliability)
    - Top-level domain (TLD) DNS servers
      * Responsible for top-level domains (com, gov, net, org…)
    - Authoritative DNS servers
    - Organization’s own DNS servers that provide authoritative hostname to IP mappings for organization’s named hosts
    - Local\* DNS servers
      * Not a part of the hierarchy
      * Have local cache of recent name-to-address translation pairs
    - B
  + Operation
    - Two Approaches
      * Iterated query
        + Contacted server replies with name of server to contact
      * Recursive query
        + Puts burden of name resolution on contacted name server
  + Caching
    - Server cache hostname-address mapping that they receive
      * Cache entries timeout after predefined time (TTL)
      * TLD servers typically cached in local servers
        + Root servers are less often visited
    - Caches entries may be out-of-date
      * Change of host IP address may not be known until its entry expire
        + Best effort name-to-address translation
  + Resource records
    - DNS distributes database stores resource records (RR) that provide hostname-to-IP address mapping
    - RR format: (Name, Value, type, TTL)
      * Time to live determines when the resource should be removed from cache
      * Meaning of name and value depends on type
        + Type = A

Name = hostname, value = IP address /

* + - * + Type = NS

Name = domain name, Value = hostname

* + - * + Type = CNAME

Name = hostname (alias), value hostname (canonical)

* + - * + Type = MX

Name = hostname (alias), value = name of mail server

* + Messages
    - Two ttypes of DNS messages: query and reply
      * Both have the same format

12/19/22

Socket programming

* Typical network application consists of two programs (residing on different systems)
  + Client program
  + Server program
* When programs are executed a client process and a se
* TCP client/server
  + Client-server application using TCP
    - Client must contact server
      * Server process must be running
      * Server must have created socket that welcomes client’s contact
    - Client contacts server by:
      * Creating TCP socket and
      * Specifying IP address and port numner of server process
    - When client creates socket, client TCP can establish connection with server TCP
    - When server is contacted by client, TCP creates a new socket for server process to communicate with that particular client