Lectures TOC:

Lecture 1

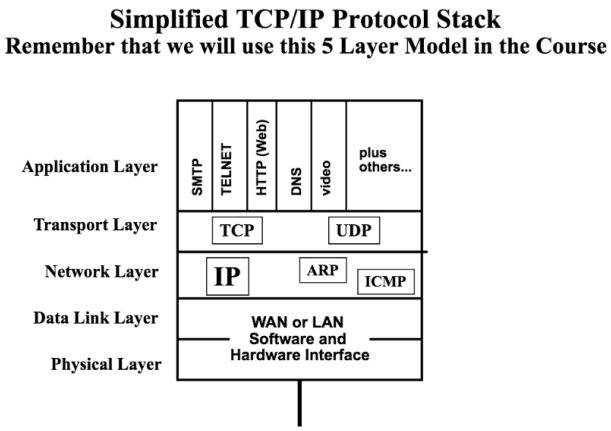
- Key Topics:
    - Bandwidth
    - Multiplexing
    - Switching
    - 5-Layer Model
    - Encapsulation
- Internet 2 (For research purposes)
- NOC (Network operations Center)
- PS (Packet Switch i.e. Router i.e. Gateway i.e. Layer 3 Switch)
- Evolution of Network Technology:

- Transmission:
  - "Sneaker Net" (Disk, USB, DVD etc.)
  - Electrical
  - Radio
  - Optical
- Multiplexing:
  - "Sharing a communication channel in a specific manner"
  - Three Forms:
    - Time division (Slotted & Statistical)
    - Frequency division
    - Optical division
- Switching:
  - Circuit Switching (think old school patch panel operator)
  - Packet Switching (Relies on Multiplexing to operate)
- Interfaces & Protocols
- Network Management
- Network Applications

- Error Rates:
  - Wireless < Copper < Fiber

- When are packet transmission errors indicated:
  - At the Software layer. The hardware layer may drop a packet but its not until the software sees the missing packet that an error can be indicated.

- Layering:
  - The internet model (5 Layers)
    - Used to be the OSI model
    - Different layers may repeat functionality (error detection, encapsulation, addressing etc.)
    - Layers are:
      - Application
      - Transport
      - Internet (Network)
      - Link (Network Interface/Data Link)
      - Physical



- Reliability:
  - **Error Detection vs error correction (big diff Prof made note!)**

Lecture 2

- Key Topics:
  - SP3
  - Packet:
    - General terminology, think: Ethernet Frame, IP Datagram, UDP Datagram, TCP Segment
  - Time Sequence Diagrams
  - Flow Control
  - Error Control:
    - Error Detection vs. Error Correction
- "Good news there's lots of standards, bad news there's lots of standards"
- Mesh Network:
  - Separating the host from the network
  - Each device in a network needs an address and its likely that they'll need more than one address
  - Local and global significance (think a cellphone: has a phone number and a serial number)
- Defining Protocols (SP3 approach):
- Service Definition (What do I want to accomplish) i.e. a reliable, sequenced transfer
- Purpose of the Protocol
- list the functionality that needs to be implemented (flow control, fragmentation, reassembly, error control)
- How do I handle this functionality in the header?
- Procedures
- PROBE packet example
- most complex part of a protocol design
- 80% of design is procedural and error handling
- Error Control is needed for daata transfer errors and procedural errors
- If a checksum is incorrect the entire "bundle of bits" is discarded
- Sophistication usually leads to efficiency

Lecture 3

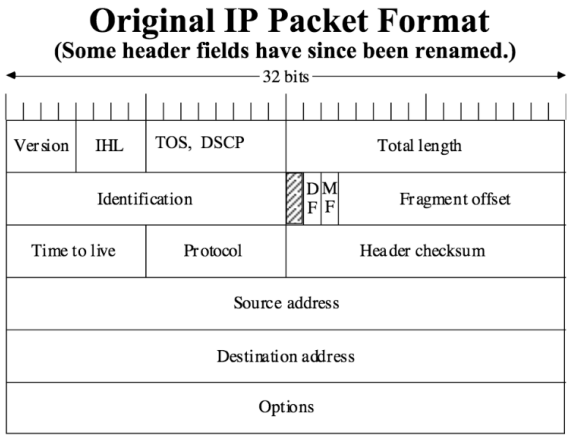
- Key Topics:
  - Ethernet
  - CSMA/CD:
    - The development and improvement of Ethernet to maximize the effective bandwidth of Ethernet in the "hubbed" era when there were many collisions
  - Hubs vs. Switches
  - Duplex
  - Switching
  - Switch Table
    - What do switch tables do?

    - Broadcast/Collision (Domains)
- 5 Layer Model is really important
- LAN: A network that takes up some limited geography
- A LAN is not a topology
- LAN vs WAN
- LAN: low delay, high bandwidth, broadcast
- WAN: pay for bandwidth by the month, relatively low bandwidth
- Ethernet has many different forms
- ALOHA Protocol from the 70's in Hawaii
    - "How do multiple users share a single resource?"
    - "How do you share a channel in a distributed decentralized fashion?"
- CSMACD
- Ethernet is an unreliable protocol
    - Error detection and not error correction
- Ethertype Field to aid the receiving system in what to do with the bundle of bits
- Ethernet vs 802.3
- **Expected to be able to draw an Ethernet frame!** 48:00 into lecture
- Multicast: A subset of devices on the network should be targeted (Target all printers)
- Vendor address at the Ethernet level (Ethernet address, MAC address) helps within the LAN for one device to find another quickly. No two devices should have the same address, but it has no routing information associated. ("Flat address")
- Layer 2 is also the Ethernet layer/MAC layer
- "Sharing at the Trunk" Sharing at the switch level
- Ethernet switch has a frame forwarding table just built off of layer 2 information
- POE utilizes the unused (4 middle) wires in the RJ45
- Inbound (left 2 wires) outbound (right 2 wires)
- Full Duplex: send and receive information at the same time over inbound and outbound wires
- Half Duplex: Send OR Receive (only one at a time) over these wires
- Duplexing software misconfigurations still happen today!!!
- MLAB: a network diagnostic tool
- 802.11
- Two Flavors (from a topology standpoint):
- Infrastructure network: Multiple devices talk to a wireless access point which then has an upstream connection

- AdHoc Network: Just amongst the devices
- QUIZ QUESTIONS:
- False: Wireless networks have better error performance than wired networks
- Different frame format: Includes management and control protocols (which 802.3 doesn't include)
- CSMACA: different protocol for wireless
- Each new iteration of wireless (g -> n -> a/c) has better speed and error performance
- They get better performance by using different modulation techniques or by using more bandwidth (i.e. a wider radio frequency)
- Higher freq. means shorter transmission distance

Lecture 4:

- Key Topics:
  - IPv4
    - How does it work?
  - ARP
    - How does it connect IPv4 to Ethernet?
  - ARP Table
    - How is it gradually filled out as IPv4 is talking to Ethernet?
  - Fragmentation
    - How does it allow for IP to make more types of networks equal by allowing IP Datagrams to pass over networks that have a smaller maximum transmit size (MTU)?
  - Classful (10. | 172.) / CIDR (Classless)
    - Should be able to do an example of "/" notation
  - ICMP
    - **Go back to lecture and review this!**
- ARP takes advantage of the broadcasting nature of Ethernet
- Some home networking all in one's devices encapsulate a router, switch, modem and firewall
- 20-30% Network utilization is good, once it creeps around 60% you may want to think about a redesign
- The bigger Ethernet network is a broadcast domain
- A cable uplink to a switch is a collision domain, and a switch port is a separate collision domain
- Ethernet futures:
- Most successful network
- Certain characteristics for Ethernet based off of industry
- for ISPs, Datacenters, Schools etc.
- If you are in a particular industry you will see specific Ethernet implementations than the norm (what we've learned so far)
- Ethernet for automotive environments is a good example of this (better connectors, less wires in cable etc.)
- IP:

- IP vs. Internet which came first? IP did! It was the building block that made the internet possible!
- Provides: addressing, routing, fragmentation, multiplexing (based on protocol type field in header, and QOS marking (lecture 4 1:09)
- CATENET: was the original name for the Internet topology
- QUIZ QUESTION: The Internet is a network of networks true or false? -> TRUE
- IP provides a limited number of common services but they are the same for every computer that connects to the internet
- Addressing and Routing
- Internet Exchange Points: FIX/MAE
- Once you say a protocol is reliable it gets complicated (Sequence numbers etc.) TCP takes care of this for IP. IP itself is not reliable
- Unreliable Protocol: Error detection but not error correction
- "Datagram" is synonomous with Unreliable
- IPV6 Does not add any reliability
- IP can run over WAN/LAN/Wireless basically over almost anything! (Carrier pigeons included :D)
- IP is a globally unique addressing scheme that allows for routing based on that addressing
- Nowadays we want to have the ability to separate the host portion from the network portion for any 32 bit IP address
- This is called CIDR notation. /15 means that the first 15 bits of the 32 are reserved for the host #
- SP3
- Service
- Purpose
- Packet
- Procedure
- 

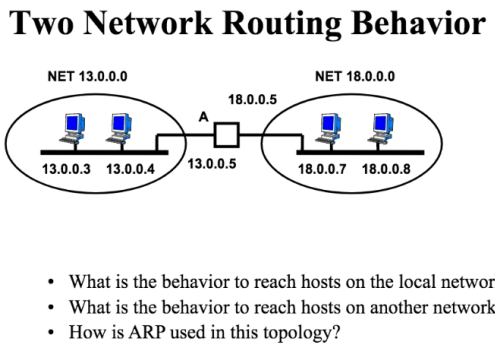
- IP packet header is of variable length so the IP packet has a header length field
- Every protocol that we study has a field in the header that identifies what the payload contains
- For IP this could be TCP, UDP, UCMP (This is an example of logical multiplexing)
- MTU 1518 bytes for the maximum size of an Ethernet packet

- IPv4 headers are 20 bytes long at a minimum, IPv6 is longer due to longer addresses
- Some specialized networks can support larger or smaller Ethernet packet sizes
- IP handles fragmentation and reassembly
- fragmentation splits packets into smaller frames and reassembly puts them back together
- There are advantages to using a MAC addres to delviering a packet to a host vs IP address
- ARP helps map IP addresses to MAC/Ethernet addresses
- ARP frame is sent out (broadcast) "Who has IP: xxx.xxx.xxx.xxx"
- Machine with said IP will respond with: "That's Me! Also, here is my MAC address"
- IP Datagram can then be sent to the proper machine
- `arp -a`
- IP Addressing:
- To the outside world the host portion of the address is known through /XX in CIDR
- The inside world (sub nets) hosts know through subnet masks
- IETF working documents for protocols are called IDs (Internet Drafts)
- If there is a lot of interest then it will move to an RFC
- This then moves to a Proposed Standard
- If it becomes widely adopted and well documented it can become an internet standard

Lecture 5:

- Key Topics:
  - IP Routing
  - Forwarding Table
    - How is this table populated?
    - ^^^ Know about RIP & BGP as they apply here
  - RIP
  - Count to Infinity Problem
  - OSPF
- Broadcasts do not pass router boundaries
- Routers:
- Operate at layer 3
- Multiple collision and broadcast domains
- Limit broadcasts to whichever side of the router that the broadcast is on
- End stations send packets to the router only if they need to leave the local network
- From an administrative standpoint Ethernet switching is easy compared to routing. "Switch when you can, route when you must"

- A router forwards packets based on some information about the "next best hop"
- All routers in a network talk to eachother and share routing table information
- **Two network routing behavior** (lecture 5 33:00)



  -

- How does the "Hop Count" work?
- After a packet enters a router, it exits with a new Ethernet header. Destination MAC and Source MAC will change (Source now being the MAC of the router) among some other fields.
- Router forwarding tables are really important
- RIPv2 (Distance Vector): A routing protocol that counts hops to measure efficiency of a given route. Populates routing table with the most efficient routes.
- Lacks efficiency when router/link hardware comes into play
- OSPF (Open Shortest Path First) (Link State): A routing protocol that takes into account routing hardware and link speed/bandwith (Complex cost function for links managed by network administrators w/ vendor input)
- A routing table for a given routing (at the beginning) sends out information about the networks/other routers it is directly connected to (lecture 5 1:18)
- VLANS:
- Users/Hosts are assigned to a specific IP Network regardless of the ethernet switch or ethernet switch port they are physically connected to
- Think of this as an administrative overlay to a given switch
- One switch can support multiple subnets
- QUIZ question (answer): The clients/users in a VLAN have no idea that they are in a VLAN vs. directly connected to a switch/router
- 802.1q VLAN Protocol
- Added the TAG field to the Ethernet frame
- VLANS are identified by a 12 bit VLAN id here
- One router port assigned to a VLAN (lecture 5 1:47:53)
- Another possibility would be to assign VLAN based on MAC address
- 802.1p Ethernet Priority Protocol

- 3 bits
- Not widely used
- In networking, there are only a handful of really good ideas: Muxing, VLANs.
- VXLAN really extends VLAN and provides Datacenter scale support (12 bits weren't enough)
- Some last questions about VLANS
- A VLAN can span multiple sites (buildings on a campus)
- There is no best way to assign users, but there are multiple ways (MAC, Router port, etc)
- A client machine has no clue its on a VLAN
- Users on different VLANS talk to each other through the routers

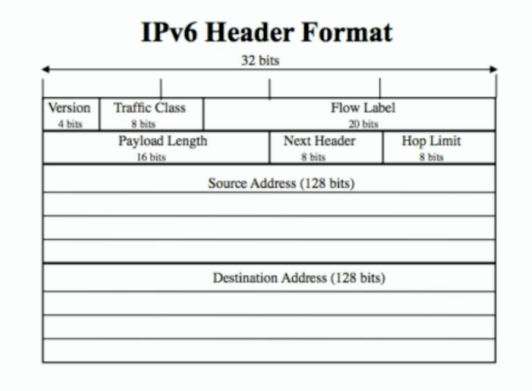
Lecture 6:

YOU MISSED THE BEGINING OF THIS LECTURE

- Key Topics:
  - ## VLANS
  - IPv6
    - How is it different than IPv4 for:
      - Addressing
      - Fragmentation
      - The changing of the headers
    - How is it similar to IPv4 for?
      - The changing of the headers
- Should be able to complete a basic RIP based routing table
- VLANS do at level 2 what previously would have had to been done at level 3
- IANA: Assigns IP addresses
- Keeps track of all of the IPv4 & IPv6 addresses
- ARIN
- `whois`
- Secondary market for ipv4 blocks
- CIDR alleviated some addressing concerns:
- Lack of address blocks
- Better routing tables
- Private IP addresses
- Autonomous Systems, ASNs, and routing

- 4 billion IP addresses out there currently and 800,000 (IPv4) network prefixes being used
- **Know how to fill in RIPv2 table**
- ASN (Autonomous System Number):
- Allows for grouping of networks
- ~ 70,000 out there
- AS (autonomous System) A group of routers managed by a single organization
- example: RCN has an ASN and hundreds of networks (Prefixes) that they manage
- OSPF and RIPv2 are used within a particular network
- IGP (interior gateway protocols) vs EGP (Exterior Gateway Protocols)
- IGP used to exchange routing information (populate routing tables) within an Autonomous System (AS)
- RIPv2 & OSPF are examples
- EGP are used to advertise and manage routes between AS
- BGPv4 (Border Gateway Protocol) is an example of this
- bgp.he.net
- A prefix identifies one network within an AS
- Internet Exchange Points:
- There are thousands of ISPs in the country. There would be a huge(r) spaghetti mess of networking without IEPs
- Companies today that offer internet service to ISPs
- Be able to explain IGP & EGP usage of an Edge Router
- RIPE NCC is another good site for tools around AS
- BOGON: Mis-advertised (non-allocated) AS prefixes
- IPV6: (because we ran out of IPv4 assignable blocks):
- IPv5 existed (research only)
- Not our first IP transition: NCP -> IPv4 in 1982
- 128 bit address vs 32 bit in ipv4
- Simplified header format
- Better support for options and extensions
- Capability for flow labelling is added
- Added authentication and privacy capabilities
- Good comparison at (lecture 6 1:23:11)
- leading `0` s in each 16 portion get collapsed

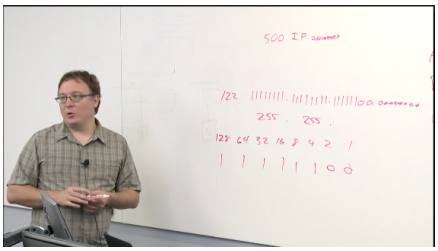
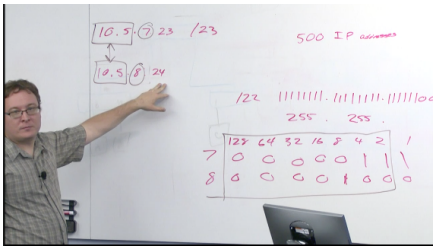
- IPv6 address dissection (lecture 6 1:28:21)



- "Private IP" -> Link Local in IPv6
- FE80:::: is a link local address
- IPv6 multicast address is much less invasive than IPv4 broadcasting
- can `whois` an IPv6 Address
- What does it mean for an org. to be on IPv6?
- Does a Ethernet switch need to support IPv6?
- Yes! Most devices need to! (See dual-stack)

Section 2:

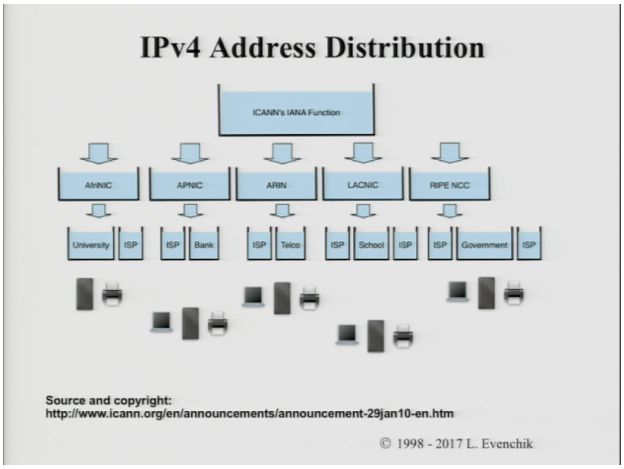
- Switch Table Failure Cases/ Switch Table diagram: Section 2 (5:35)
  - Ethernet broadcast/Multicast addresses
  - When a switch table is empty a broadcast will always be sent out to populate
  - Newly plugged in devices send out a gratuitous ARP to aid in invalidating other devices switch tables that it may have been connected to prior.
- Lecture 4 Key Topics:
  - IPv4:
  - Classful / CIDR addressing
  - ARP & ARP Table
  - Fragmentation
  - Routing
  - Ping / Traceroute
- Lecture 5 Key Topics:
  - IP Routing
  - Forwarding/ Routing Tables
  - RIP/OSPF
  - BGP
  - VLAN
- Lecture 6 Key Topics:
  - AS Numbers
  - Exterior Routing - BGP
  - IPv6
- Ethernet: Single broadcast domain, many collision domains
- VLANS:

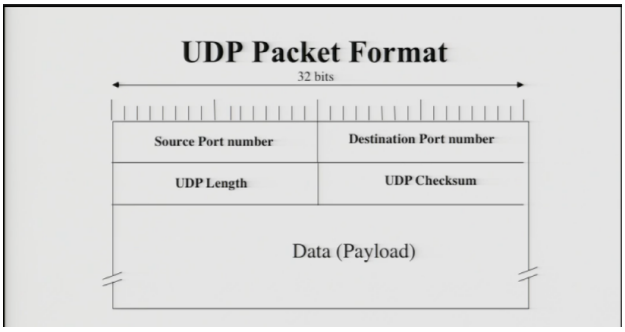
    - In a VLAN-ed network the "top" router acts as a switch, and you logically assign the lower switch ports to VLANs (VLAN Tags/Numbers)
    - "Lower" switches assign VLAN tag to Ethernet frames, and normal forwarding ensues, but only within the VLAN Tag/color/number
- IP:
    - TTL in IP packet stop IP Loops
    - "Make all networks equal"
    - Unreliable Datagram protocol
    - HDLC, PPP, Token Ring are old layer 2 networks that Ethernet extinct-i-fied
    - IP vs. Ethernet (Section 2 32:39)
    - Classful & Classless (CIDR):
    - Classful
    - Class A, Class B, Class C
    - 2^24 hosts per subnet, 2^16 hosts per subnet, 2^8 hosts per subnet
    - Led to the "Goldilocks Problem":
    - If you had a need for 500 IP addresses, you're forced to use class B, but its very wasteful.
    - Variable Netmasks: Solved this problem, but you can no longer calculate subnets based on IP alone (this leads to CIDR notation)
    - With the granularity of variable netmasks the internet was saved! And IP block could be allocated much less wastefully!
    - Subnet/IP math (Section 2 43:13)
    - 
    - **QUIZ Question: "Are these two IP addresses on the same subnet?"**
    - 
- ARP: Map Ethernet(MAC) addresses to IPv4 addresses
    - Samuel Clemens vs. Mark Twain
    - The magic of ARP is that it's asking a very important question: "Who has?"
    - Layer 2 Ethernet broadcast address: "Who has IP: xxx?"
- Fragmentation:
    - IP is trying to make all networks equal
    - Different networks had different MTUs (Maximum transmission units)
    - IP needed to handle this case.
    - An IP packet is able to divide itself into smaller chunks using the fragmentation fields in the packet (Id/Fragment offset/More Fragments)
    - Reassembly and then pass it up to layer 4
    - Fragmentation creates more opportunities for failure (Section 2 1:09)
    - In IPv6 only the originating host can fragment
- Routing:
    - Ping: very basic network debugging (ICMP echo request)
    - Nmap is a useful tool as well
    - All hosts have a Routing and ARP table (Section 2 1:13)

        - Routing tables have more pertinent fields
        - `netstat -rn`
    - ARP table: "How doI talk to my neighbors?"
    - Routing table (Forwarding Table): "How do I talk to the rest of the world?"
    - **Grand Unified Network Theory: Layer 3 & 2 working across networks:** (section 2 1:30)
        - Every Router port has a MAC address
    - `traceroute` is LIT -> Lets you see the hops (generally) to your specified destination
    - `ping` is like Sonar, `traceroute` is like google maps driving directions
    - Routing (forwarding) tables are built with:
    - Interior: RIP & OSPF
        - Within one company or org
    - Exterior: BGP
        - Connecting multiple companies/orgs to the internet backbone
    - Every large company has an AS #, and these numbers are how Exterior protocols create a network mapping of all of the Autonomous Systems in the world.
    - 1 AS # == Many Networks
    - ARIN allocates network blocks
    - You can tell by looking at an ip now where geographically it comes from
    - ARIN reached IPv4 depletion
    - The internet needs to be saved!
        - Here comes IPv6 and NAT-ing!!!

Lecture 7:

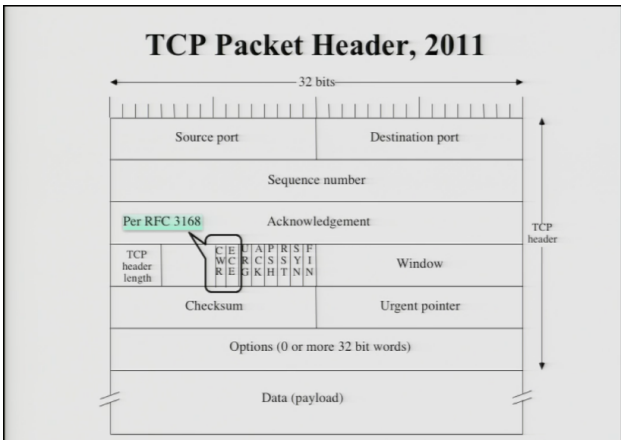
- Key Topics:
    - UDP
    - TCP
    - 3-Way Handshake
    - Sequencing
        - How a specific stream of bits is divided into multiple segments which are sent individually over the wire in different IP datagrams
    - Congestion Control
        - "Slow Start" is a common method
    - Connection Table
- Most if not all modern devices are running IPv6. But when you talk to something on the larger net, it will most likely be IPv4 due to lack of support from intermediary networks.
- Is IPv6 going to happen? Yes! 10% of traffic is currently!
- Cellular networks had the foresight to be IPv6
- IPv6:
    - Expanded addressing capabilities. 128 bits, improved auto configuration, anycast addresses etc.
    - Simplified header format
    - Better support for options and extensions
    - Capability for Flow Labelling is added
    - Added auth and privacy capabilities
    - Link local: (replaces private IP with the FE80)
    - How is multicast handled?

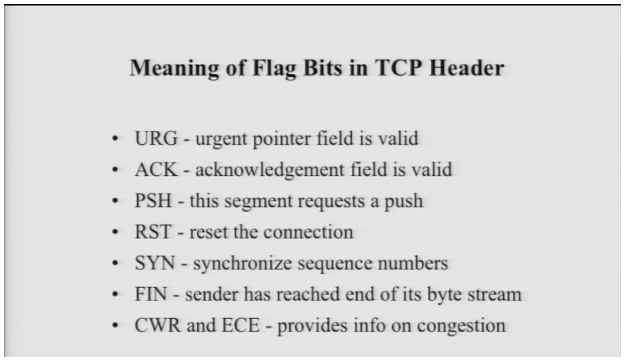
- **QUIZ**
  - Know what the IPv4/IPv6/Ethernet headers look like
- IPv4 `ping` vs IPv6 `ping`:
  - How does a "dual stack" devices know whether or not to handle an incoming frame with IPv4/IPv6? -> The EtherType (Protocol Type) field in the Ethernet header!
- Address Allocation:
  - 

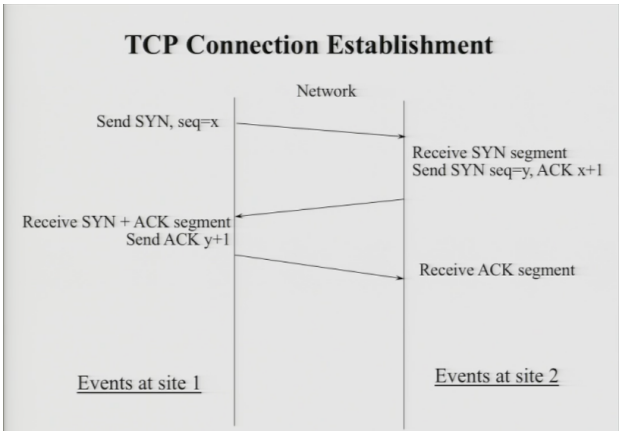
- Autonomous Systems, ASN, & Routing:
  - Routing Protocol Families:
    - Interior/Exterior
    - (OSPF/RIP)/BGP
  - Harvard has many Autonomous systems (since its been in the business for so long)
  - AS advertise network prefixes
  - BOGON: Somebody ha misconfigured their routing table and are announcing a prefix they don't own
  - BGP Hijacking: Advertising the wrong prefix could allow for someone else's traffic to be routed to you. Said traffic would get dropped, because there's no internal network, but the hijack could happen to snoop on the traffic and then re-route it back to its proper place.
- **Transport Layer:**
  - SP3 Protocol framework
    - Service
    - Purpose
    - Packets
    - Procedures
  - **UDP**:
    - 

    - Completely unreliable

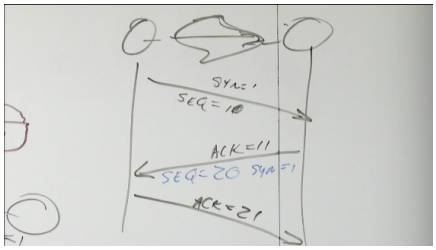
- Multiplexing
- Error detection but not error correction
- Datagram delivery service (no error correction)
- Different types of layer 5 protocols in its payload (Port # in the UDP header)
- The IP header has a Protocol Type in its header (Lecture 7 40:50)
- Checksums:
  - Ethernet was at the end of the frame
  - IP is in the header
  - UDP is a bit different. It can prevent the misdelivery (Lecture 7 46:00) of packets by creating a pseudo-header that is prefixed to the UDP header. As part of the checksum computation, it pulls in the originator's source and destination address, IP protocol field and the UDP length.

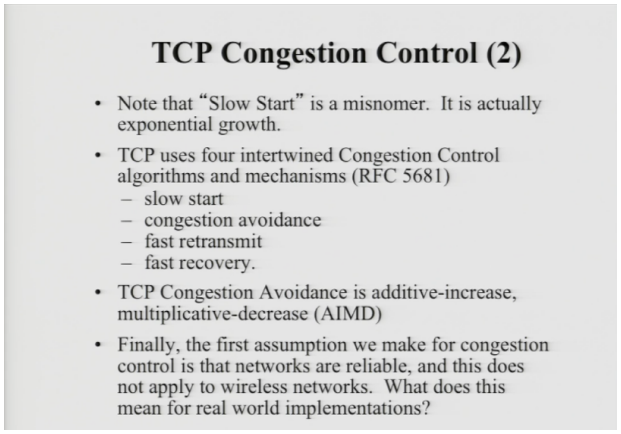
- **TCP**:
  - 

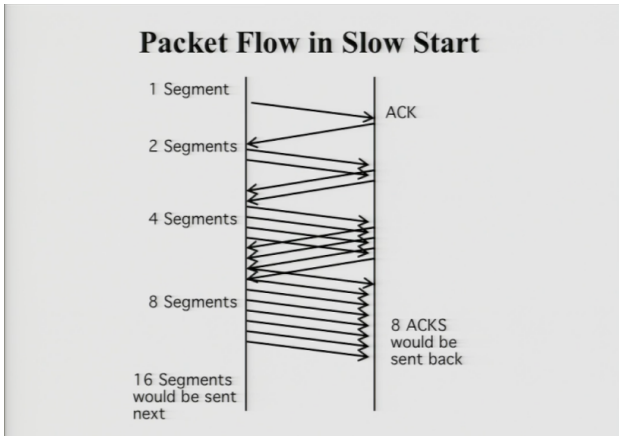
  - Reliable, connection-oriented, sequenced delivery of packets/payloads
  - TCP is quite complicated and implementation-specific
  - Window size is directly related to Flow Control
  - Was designed to be incredibly efficient (HELLO vs single bit):
    - 

  - 

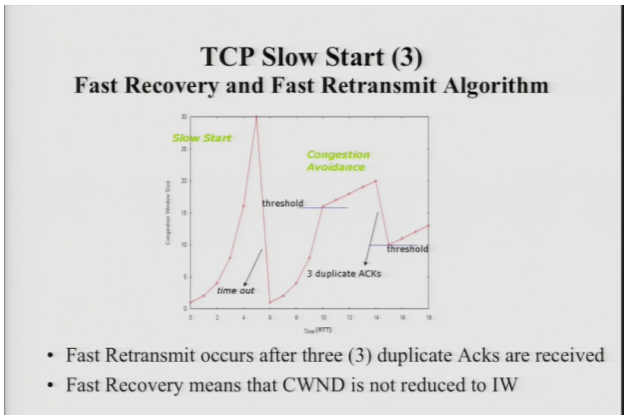
  - Sequence numbers are very important and they keep track of bytes not packets
  - Sequence numbers start with a pseudo-random value for security purposes (if everyone started at 0 it would be easier to spoof)

- **Full duplex**
- Three way handshake:
  - 

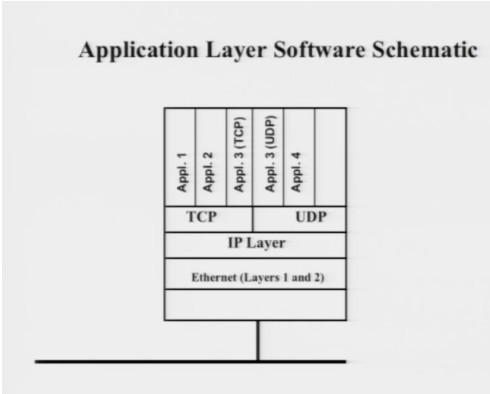
- The amount of data that can be sent between two devices is determined by the receiver
- Byte Stream functionality
- TCP sends data in segments (MSS max segment size) is negotiated during connection setup
- MSS is dependent on the size of the MTU
- **TCP segments can arrive out of order!**
  - The receiving application sees the segments in the proper order. It never knows if they're out of order. There is a buffer at receiver's TCP layer that segments are held in until order is ensured.
- TCP Congestion Control (Flow control):
  - "Slow start" (this is a misnomer) (Lecture 7 1:26:51)
    - 

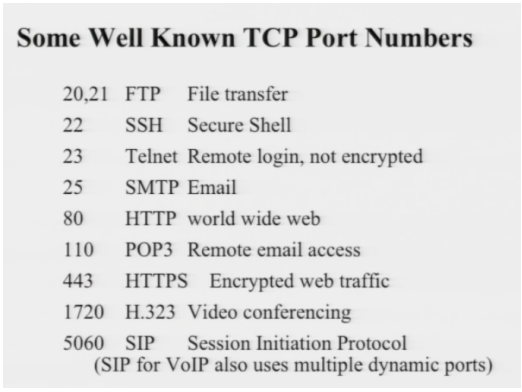
    - 

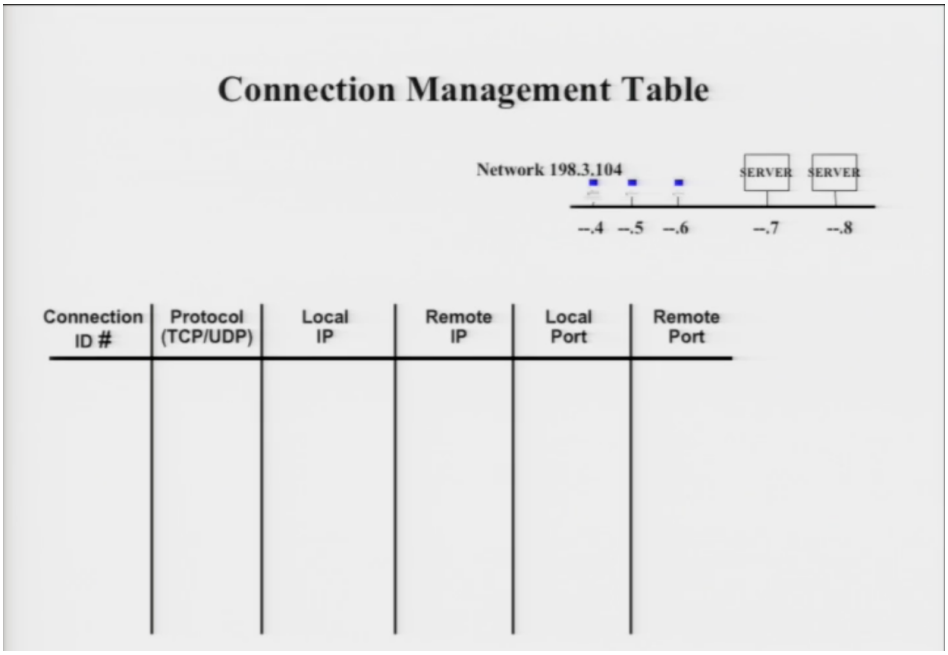
  - Additive increase: At a certain point one of the many constraints (Flow Control window size, Host says: "Maybe I'm sending too much") will be encroached upon so the exponential increase stops and growth of rate of segments being sent becomes linear.
  - If there is a loss, then the # of segments being sent drops and this growth starts again.



Lecture 8:

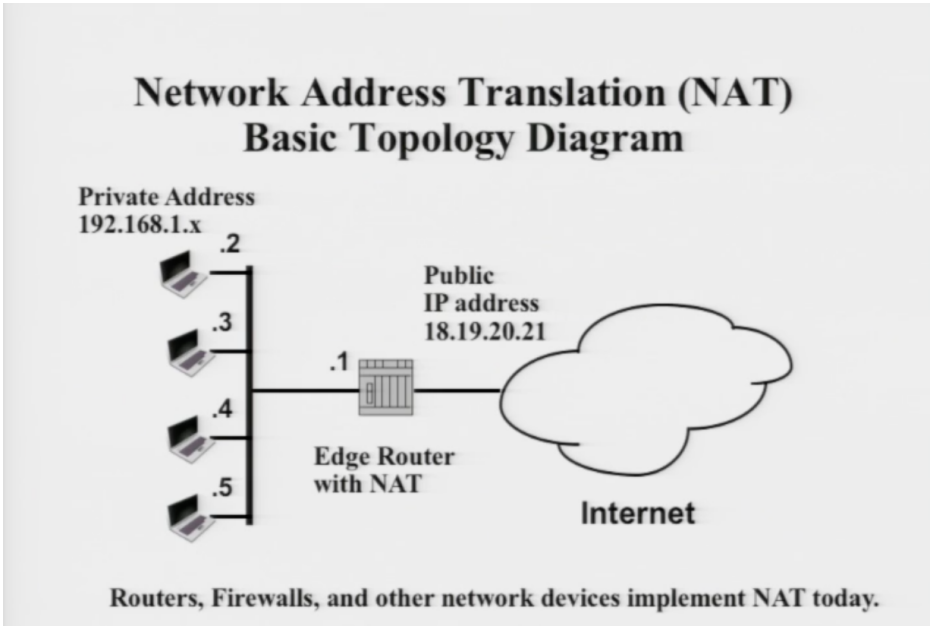
- Key Topics:
  - Private Addresses
  - NAT
  - NAT Table
    - 7-Tuple
- Connection Management & NAT (Network Address Translation)
- IXP (internet exchange points) a bunch of specialized locations where many ISPs get together and pass traffic around
- UDP Pseudo header (What its there for, why its there, and how it works and prevents mis-delivery)
- TCP: **Difference between Flow Control and Congestion control**
  - Flow Control:
    - End to End: A receiving computer isn't able to sustain the input from a sending host
  - Congestion Control:
    - Network issue: The network itself is not able to handle the traffic and the network is telling the host to slow transmission
    - **"Slow Start"** Slow only relative to not doing anything at all
    - Doubles segments sent every time it receives the full set of ACKs until packet loss, then increase in segments is additive
- **Connection Management:**
  - How does a device uniquely identify its connections to remote machines? (all application layer connections through TCP)
  - **IP delegates to TCP or UDP based on "Protocol Type" in IP packet header field**
  - TCP knows which Application layer (Layer 5) application to send to by the destination port
  - 

- 

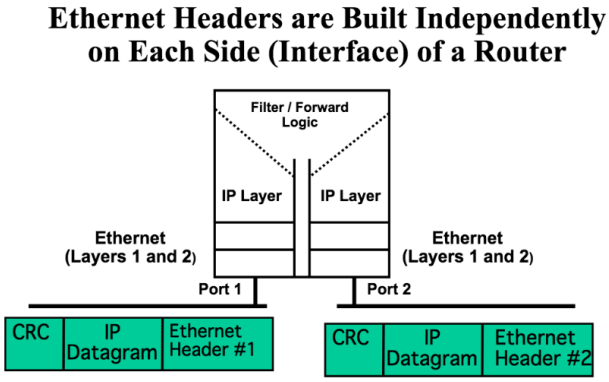
- 

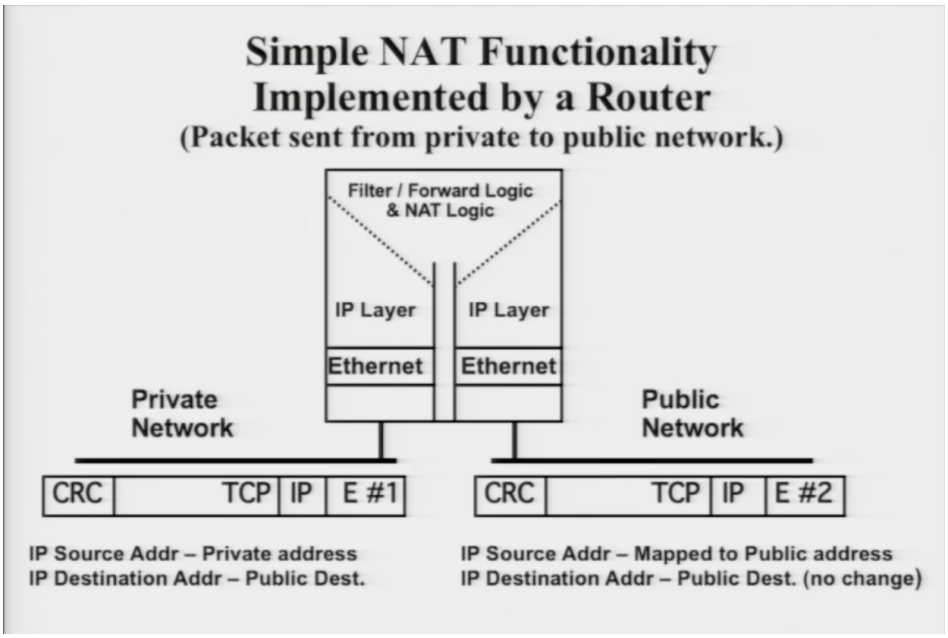
  - The receiving system has no control over what the source port (from the client) will be, and its perfectly valid for many clients to have the same source port
  - All 5 pieces (local/remote IP, local/remote Port, Protocol) are used to uniquely identify a connection **This is the 5-tuple mentioned earlier!**
  - Prior connection management tables are local to the client machines!
  - This whole table is just so the machine can determine which applications are using which port(s)
  - `netstat -an | more`
  - `whois <ip address>`
  - Operating system processes will use inter-process communication over TCP (through the local network interface

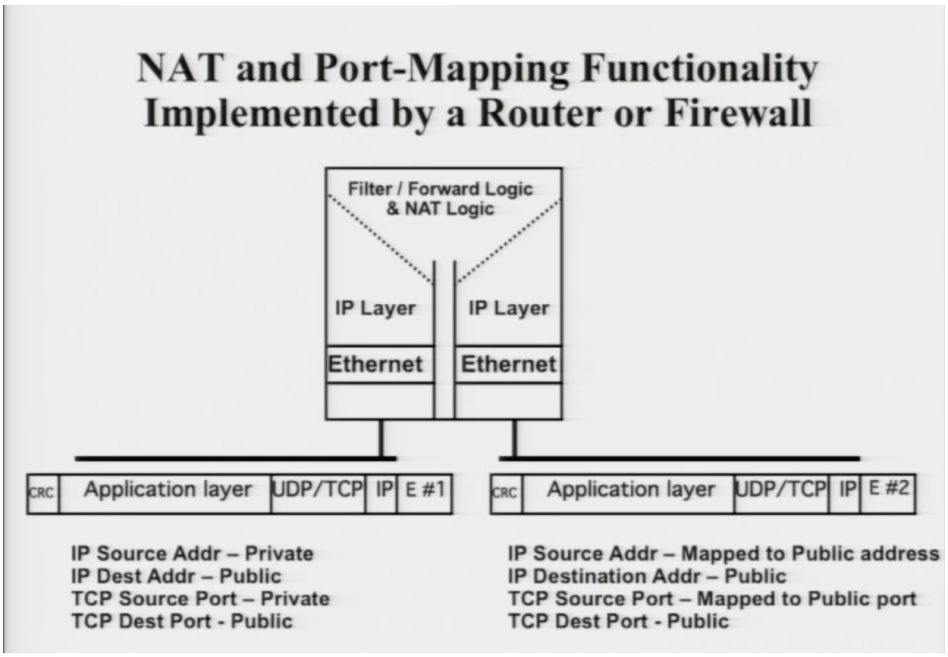
- **NAT:**

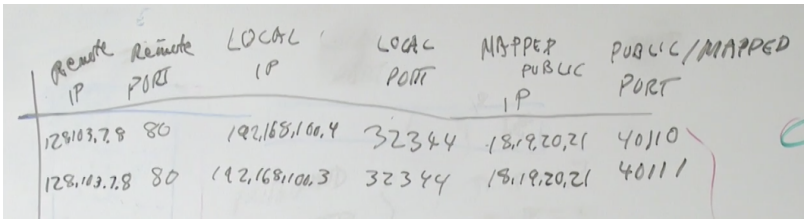
- 

- Different types of NAT:
  - 1-to-1 address mapping
  - N-to-N address mapping
  - Network address and port mapping (NAPT or PAT)
- **Remember that Ethernet headers are built on each side of a router!** (Lecture 8 1:05:07)



- 

- 

- The edge device doing the NAT logic assigns public/mapped ports to be able to route traffic to its private ips on the other side
- 

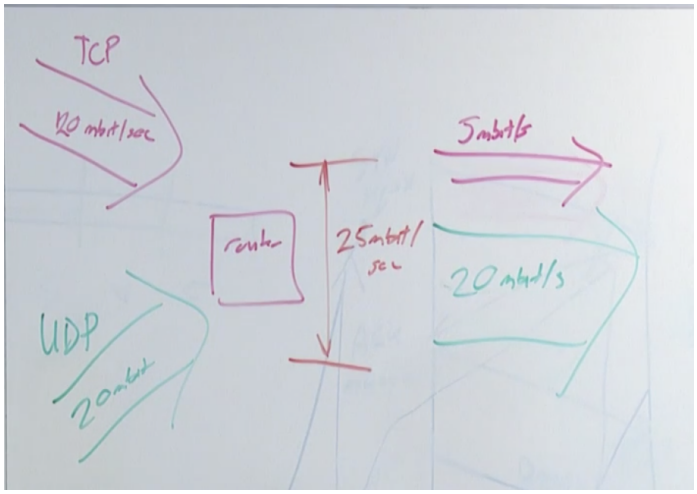
- NAT handles incoming connections by the use of port forwarding
- **NAT is not a security solution** it only obscures internal private addresses
- It makes network management and debugging much more difficult
- Its being used as part of the transition to IPv6
- **QUIZ: Does a client know that its operating through a NAT box?** NO!

- Multiple NAT-ing devices can be configured on the same network

- **STUN & ICE:**
  - **STUN: OS protocol to tell whether or not NAT is present**
  - Some applications on the net need to know if NAT is being used or not.
  - VOIP for example needs to know if NAT is being used
  - ICE: a way to if NAT devices are present and what/how they are operating

Section 3:

- Lecture 7 Key topics:
  - Streams vs. Packets (UDP vs TCP)
  - UDP & TCP Source and destination ports
  - UDP: pseudo header
  - TCP: 3 way handshake, sequencing, congestion control, slow start, connection table, NAT
  - Connection table (5-tuple)
  - NAT (7-tuple)
  - STUN/TURN/ICE
  - IPv6:
    - ARP/ Fragmentation works differently
    - Use double colon to mean "all zeroes" Ex. f000🆎:1eab
    - Can trim leading zeroes
    - 2000::/3 Global unicast
    - FE80::/10 Link Local Unicast
    - DNS A vs AAAA records for ipv6 (4 times larger that an ipv4 address, also a sick joke! :))
    - "Next Header" for encapsulation concept vs Russian doll nesting
    - IPv6 is expensive to implement!
    - `dig -T AAAA facebook.com -> 2a03:2880:f112:83:face:b00c::25de` :D
  - TCP & UDP:
    - UDP:
      - "User Datagram Protocol"
      - Unreliable
    - TCP:
      - Connection-oriented Protocol
      - Reliable
      - bidirectional communication mechanism
    - Port numbers are "new" concept when these protocols came along
    - Destination Ports == "What service do I want?"
    - Source Ports == Identification for the connection

- `cat /etc/services` See well known ports
- TCP/UDP Pseudo-header:
  - Brings in information from IP header: (source/destination address, Protocol, UDP length)
  - Specifically designed so that layer 4 software can tell if a UDP Datagram has been misdelivered
  - There's no other mechanism in UDP to detect misdelivery (TCP has the three-way handshake as well)
  - Calculating the entirety of the UDP Datagram plus the above-mentioned fields
  - Is a very basic checksum
- TCP: 3-Way Handshake
  - Agree on initial sequence numbers and window sizes
  - Confirms that the sender and receiver are "real"
  - SYN -> SYN/ACK -> ACK
- Send data with TCP segments after connection is established:
  - Size constrained by the MTU
  - Keep getting ACKS back from receiver
  - Can be received out of order
  - Selective repeat, won't resend entire payload
- Flow Control:
  - Think window size
  - So that the sender can't overwhelm the receiver
- Congestion Control:
  - The network can be saturated even if the endpoints are not!
  - TCP slow start to the rescue!
  - Name is a misnomer (slow is relative to sender)
  - Make the window size exponentially larger until loss or full transmission happens the chop back down to some threshold and do additive increase afterwards
  - Window size can constantly be negotiated by both ends of the connection
  - How is this like the development of Ethernet?
  - UDP has no congestion control (Section 3 1:05:00)!
    - Just keeps blasting!
    - 

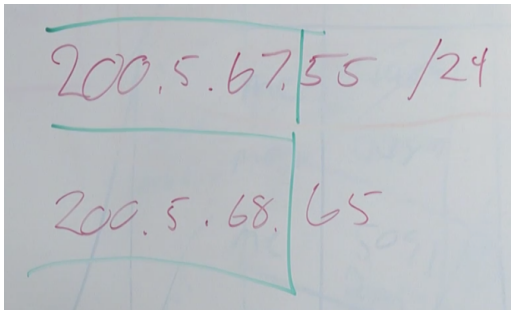
    - TCP goes through a back off and oscillation phase
- Connection Table:

        - 5-tuple used by each host to uniquely identify connections
            - Protocol, Source/Destination IP, Source/Destination Port
        - Each tuple is globally unique! WOW!
    - NAT:
        - To aid in IP address exhaustion
        - Allows you to share a very small number of public ips with your friends and neighbors
        - Now we have public and private addresses!
        - Three private ranges:
            - j. 172.16., 192.168
        - Not allowed to pass over the internet
        - Passes messages between public and private
        - Not the same as a firewall, its primary function is not security
        - Think "Port Forwarding" & "Bigger on the inside"
        - Limitations:
            - Layer 5 protocols won't NAT very well
        - NAT table:
            - Sits on NAT device
            - 7-tuple
            - Adds two fields: (Mapped IP & Mapped Port)
        - NAT device rewrites IP header and TCP header & calculate checksums -> and then in reverse on the way back
        - Completely transparent to end users
        - A Hack that has broken the end-to-end principal of the internet
        - A NAT can map public IPs to other Public IPs
    - STUN/TURN/ICE: "Am I behind a NAT? If, so what kind?/Relaying/BEst path to communicate"

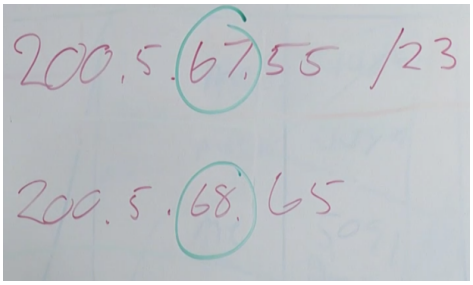
Midterm Exam Review:

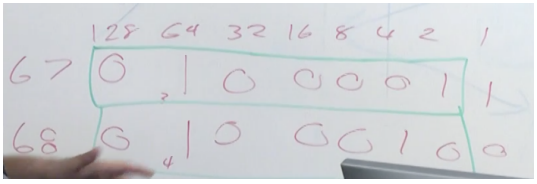
- Know detailed layout of Ethernet header
- Basic Theories, Technologies and Definitions:
    - Protocol Layering
    - Encapsulation
    - Muxing
    - Packet Switching
    - Packet Loss
- Tools:
    - Time Sequence Diagrams
    - State Diagrams
    - 5-tuple/7-tuple
- Protocol Framework:
    - Protocol Functionality: Error Detection vs. Error Correction
    - SP3
- Architectures and Topology:
    - Bus
    - Point to Point

    - Ring
    - Star
    - Mesh
- Building Blocks:
    - Switches
    - Routers
    - NAT
- Protocols
- Design Tradeoffs and performance issues
- **At this point I'm going back to update my notes from prior lectures as to not duplicate useful info.**
- Types of problems on the midterm:
    - Math Problems:
        - Bandwidth Calculation from HW1
        - Basic IP Subnetting and Binary Arithmetic
        - Example Questions:
            - How many hosts in a `/24`, `/25`, `/23` ?
        - 32 total bits - the host bits (in the case of 24 -> 32 - 24 (network bits) = 8 (host bits))
            - 2^8 = 256
            - 256 - 2 (Network address (.0) & Broadcast address (.255)) = **254!**
        - `/23` has 510 potential host for comparison
        - Are `200.5.67.55/24` and `200.5.68.65` in the same network?
        - YES!



        - Same question, but `/23` ?
        - "Break" is in the third octet
        -



- Nope! Green boxes don't match! (7 boxed from that octet because 16 (first two octets) + 7 = 23)
- 

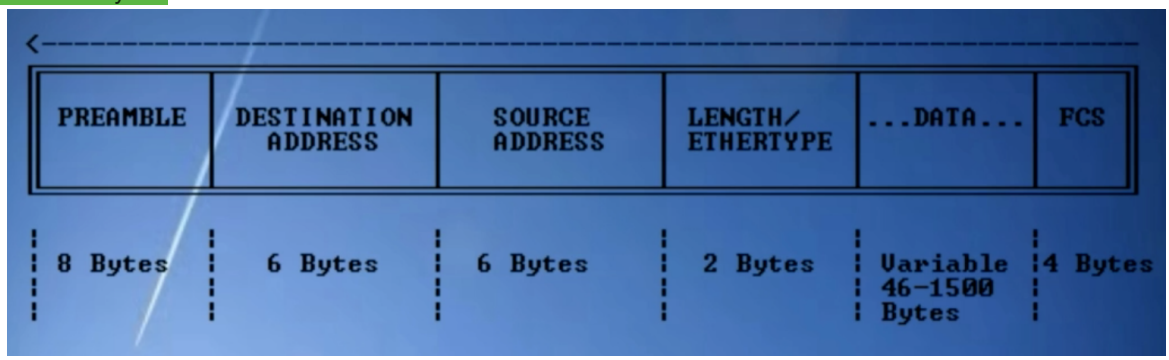
- Lots o' tables:
  - How is each populated?
  - What protocols are used by each?
  - Switch Table
    - Listens to the Ethernet frames that are going over the wire, and looking at the source address fields of each frame.
    - "Do I know that this source address comes from this port? No? Cool, I'll add those to my table."
  - ARP Table
    - Used by IP
    - Map IP to Ethernet (MAC) addresses
    - Populated by sending an ARP message (broadcast)
    - Will then get an ARP Reply with the destination IP's Ethernet (MAC) address
  - Can "zero" these ^^^ two tables at any time and the software will rebuild them. Kind of Transient
  - Routing/Forwarding Table:
    - Table on a router
    - Populated by: RIP, OSPF, or BGP
  - Connection Table
    - Describes the data that is passing over a machine
    - Each host has one
  - NAT Table
    - Describes the data that is passing over a machine
    - Only the NAT device can know all 7 fields

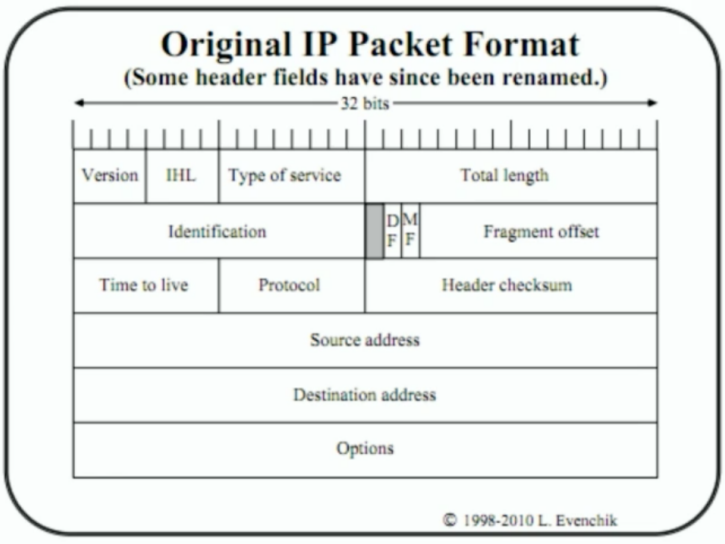
- You run a webserver (ports 80 and 443) and recieve connections from hosts A, B and C. A is connecting twice. Draw one possible connection table to describe this.
Host	Local Port	Remote Port	Local IP Address	Remote IP Address	Protocol
128.103.104.105	Probably something between 1024–49151	80 (http)	128.103.104.105	18.19.20.21	TCP
128.103.104.105	Probably something between 1024–49151	25 (smtp)	128.103.104.105	18.19.20.21	TCP

- Explain how a link level protocol that uses a window size of 127 could be more efficient than a protocol that uses a window size of 7. Include in your answer how the link's end-to-end delay and the link's bandwidth affect the link's performance.
  - The larger the window size, the more effective bandwidth you can consume because there is more data that can be on the wire.
  - The 127 one is more efficient because up to 127 "blocks"/frames could be sent before an acknowledgement (ACK) has to be sent.
  - A link with a long delay (high latency) will perform very badly with a small window size.
  - A large(r) window size could effectively take up the entire link's bandwidth

- Is it correct to say that one Ethernet frame on the wire (1500 bytes in length) can carry only one encapsulated IP datagram? How about the reverse?
  - The former is TRUE! The Ethernet protocol only allows for the encapsulation of a single IP datagram.
  - The Reverse: You can divide a single IP datagram across multiple Ethernet frames through the use of fragmentation.

- Know Your Core Protocols:

- Ethernet -> Layer 2
  - 

  - Frame Check Sequence (Basic Checksum) at the end!!!
  - Preamble doesn't matter the much
- IPv4 -> Layer 3
  - 

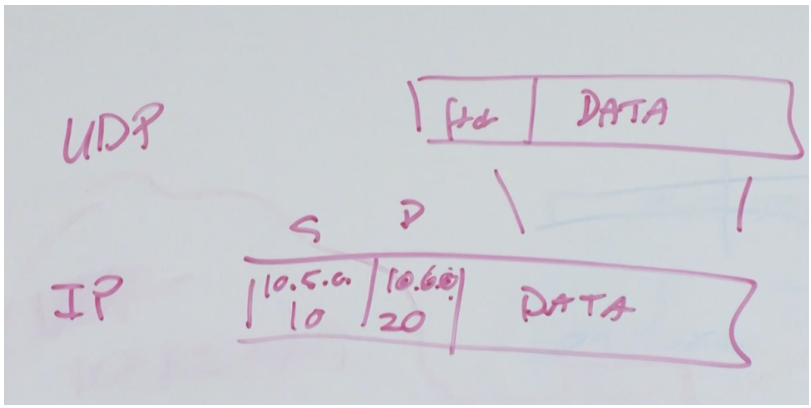
- TCP -> Layer 4
  - TCP Header

Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Source port																Destination port															
4	32	Sequence number																															
8	64	Acknowledgment number (if ACK set)																															
12	96	Data offset				Reserved 0 0 0				NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN	Window Size														
16	128	Checksum																Urgent pointer (if URG set)															
20	160	Options (if *data offset* > 5. Padded at the end with "0" bytes if necessary.)																															
...	...	...																															

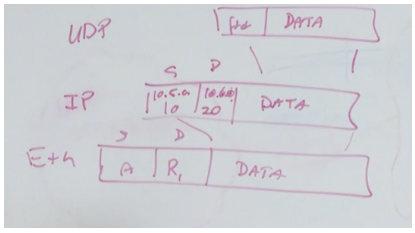
  - SYN & ACK flags really matter!

UDP -> Layer 4

- You Have a computer at IP address `10.5.0.10/24` with a default route of 10.5.0.1. Describe what happens when you send a UDP message to `10.6.0.20`. What tables are used on the host?
  - The host's (`10.5.0.10/24`) routing table will at least have entries fro local delivery (10.5.0.0/24) and the default route mentioned (10.5.0.1)
  - We create an UDP Datagram (layer-4) encapsulated within an IP Datagram (layer-3):

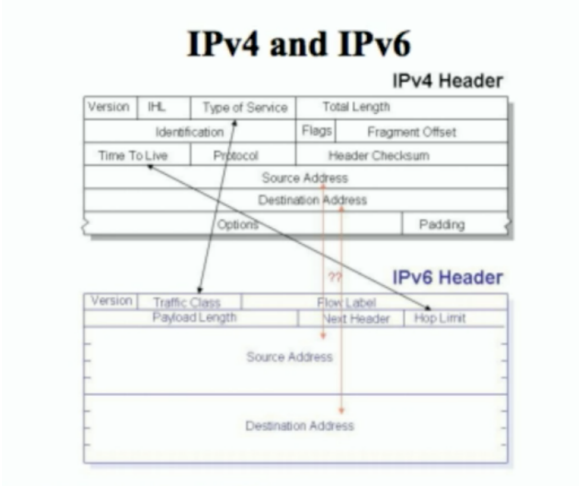


- Before anything goes down at layer two, we first consult the aforementioned routing table
  - "Is `10.6.0.20` in my subnet?" -> No! So let's use the default route
- Host now consults its ARP table... and doesn't see anything there
  - Host sends out an ARP message: "Who has 10.5.0.1?"
  - Router responds with ARP Reply containing MAC address of its network interface.
- With this new information, we can now move on to layer 2 and construct an Ethernet Frame!



- Include the connection table of the host as well!

- Describe two differences between an IPv4 and an IPv6 header:
  - Source/destination address sizes
  - TTL -> Hop limit



- Describe the mechanism that TCP uses that makes transmitting layer 3 Datagrams "feel" like a connection.
  - It is taking a block of data, and dividing said block into individual segments
  - It is numbering those segments (starting with a pseudo-random # to avoid leaking knowledge of cryptographical importance)
  - It sends these segments on the wire
  - It will "feel" like a connection because TCP is adding in reliability
    - Ensures that if any segments arrive out of order they get reordered
    - If any segments are lost in transmission they're sent again

    - The combination of these facets allows for TCP: a sequenced unreliable protocol to operate over networks that are unsequenced and unreliable

- Describe what Ethernet, IP, and TCP do when they identify corrupted data

    - They all have header checksums of one variety or another
    - Ethernet has the Frame Check Sequence that is stored at the end
    - IP has an IP header checksum
    - TCP has a header checksum based on a pseudo header
    - They will all assert that the given checksum matches what is expected
    - Ethernet and IP will discard the data if error is detected
    - Ethernet and TCP will notice an error anywhere in the frame/segment since their checksums on the frame/segment in its entirety
    - TCP will also discard the data, but it will also enable its error correction techniques to ensure that said data is retransmitted

- Token ring networks (an older form of LAN) and Ethernet networks have different maximum frame sizes (approximately 1,500 vs. 4,500 bytes). Given that Ethernet switches can be used to interconnect these two different types of networks, should these switches implement fragmentation and reassembly? Why or why not?

    - No way! You should not be implementing fragmentation and reassembly because its supported as part of the IP protocol at layer 3

- Describe what happens to an Ethernet frame when it traverses a wiring hub, an Ethernet switch and a router. In each case, describe what protocol fields (if any) change at both layer 2 and layer 3.

    - Hub:
        - When a frame traverses a hub, it gets "sucked in" and "spit out"
        - Does not speak Ethernet or check Ethernet Frame Check Sequences
        - Very simple piece of hardware
        - No protocol fields change
    - Switch:
        - Still gets "sucked in" and "spit out"
        - A switch will check the Ethernet Frame check sequence and do error detection by discarding frames
        - It could consult its stich table and say "I'm going to send this frame out port X"
        - Or "I'm going to push this out every port except for the port it came in on"
        - No protocol fields change
    - Router:
        - When an Ethernet frame hits a layer 3 router, depending on your point of view either the whole frame gets discarded (pulling off the frame on the left side and completely rebuilding it on the right side)
        - On the "right side" the Ethernet frame will have different source/destination addresses, different Frame Check Sequence (would recalculate it)
        - The IP Datagram in the Ethernet frame have a decrement happen to it's TTL field
        - Because the TTL changes we have to recalculate the IP header checksum as well

- Define time-division multiplexing and frequency division multiplexing. What techniques are used by a cable or DSL connection?

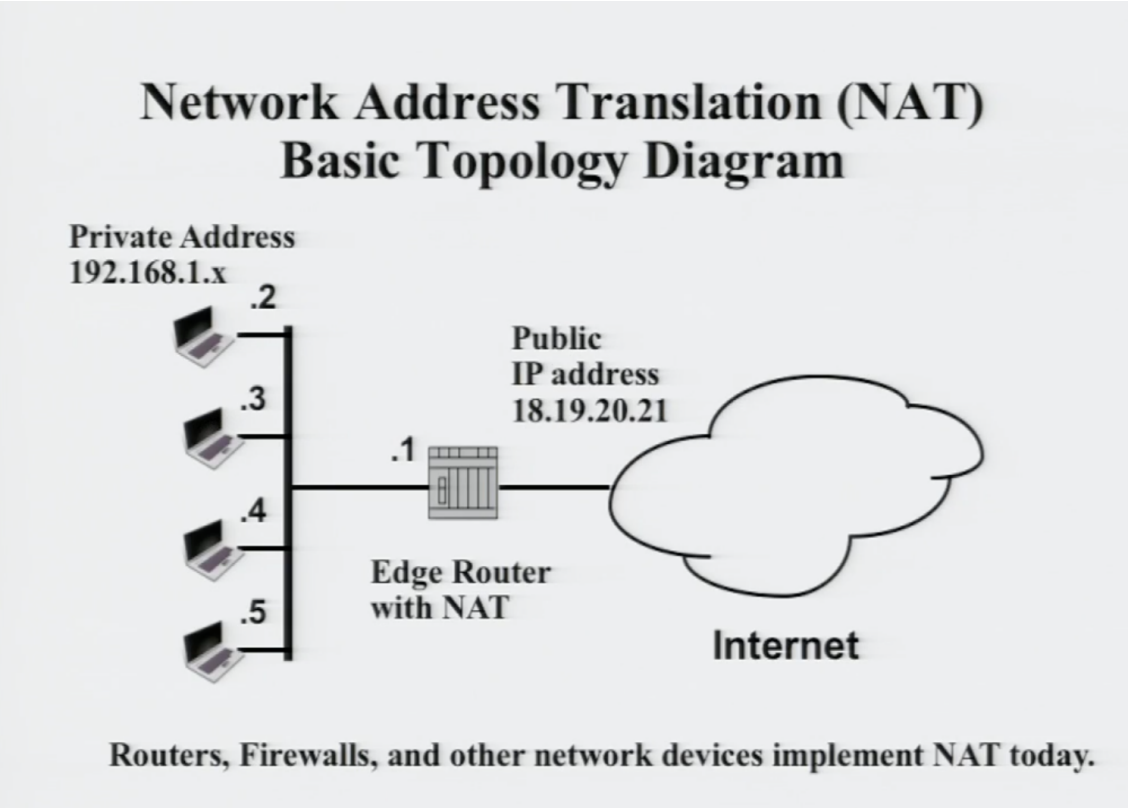
    - Time Division:
        - Taking a single piece of media and saying: "at one moment we use this flow, and at another moment we use another flow"
    - Frequency Division:
        - Dividing a piece of media using the Electro-Magnetic Spectrum into multiple channels
        - Think 2.4Ghz & 5Ghz or Radio stations

        - Cable and DSL connections use both!
            - A cable modem divides your coax into multiple channels: (actual TV cable, incoming bandwidth, outgoing bandwidth)
            - A DSL connection divides your RJ-11 into multiple channels: (analog phone system, range for sending, range for receiving)
            - If you looked at these wires at any given moment a different IP Datagram could be passing over it (Time Division)
    - Describe your current internet connection briefly. Are you behind a NAT? Do you have a public or private IP address? Describe how you came to these conclusions.
- Questions from the class:
    - "I know that routers have ARP tables. Do hosts and switches have ARP tables or must a packet with an IP destination be sent to the router?"
        - Every device on a wired IP network using Ethernet has an ARP table.
        - Two machines on the same wired network do not have to talk to the router to communicate because they are on the same subnet. (Machines like this will be in each other's ARP tables (after they've spoken))
        - Wireless Ethernet has no guarantee since the ranges may not overlap, so all communication has to go through the wireless router
        - Switches don't have ARP tables

Lecture 9:

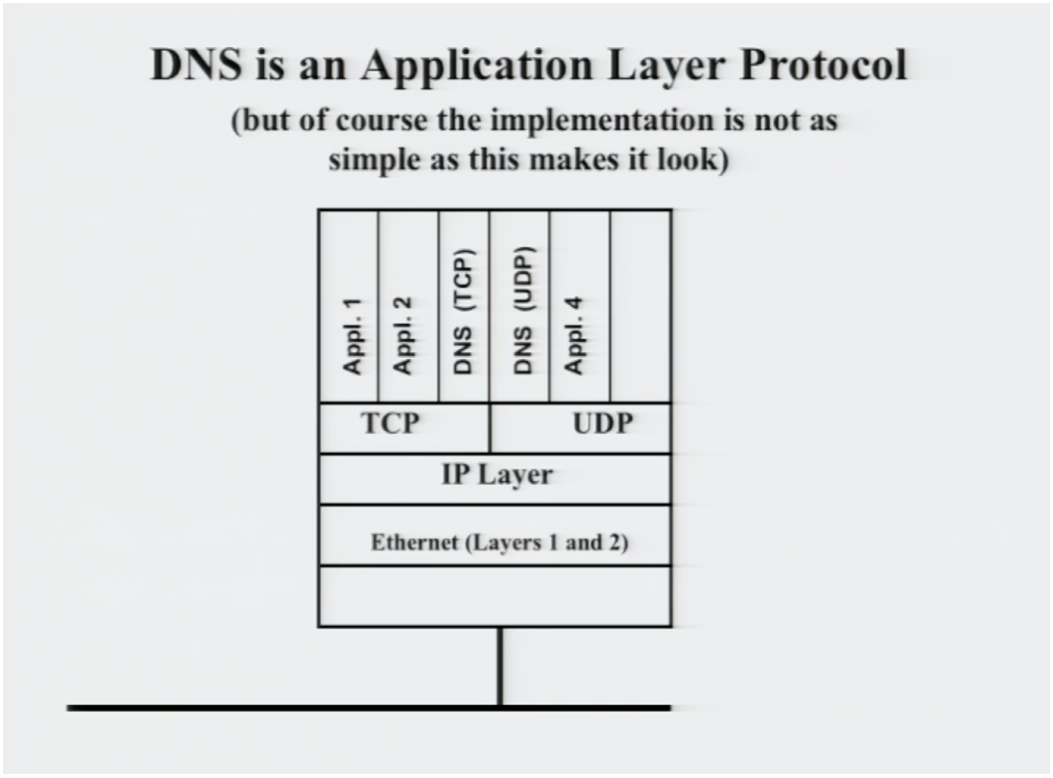
- DNS
- Application Layer Protocols
- EMAIL Protocols (SMTP) and Architecture
- Connection management:
    - Network Security & Firewalling
    - 5-tuple
    - Connection Management Table **Final***
- NAT:
    - 7-tuple
        - Map Source IP/Ports

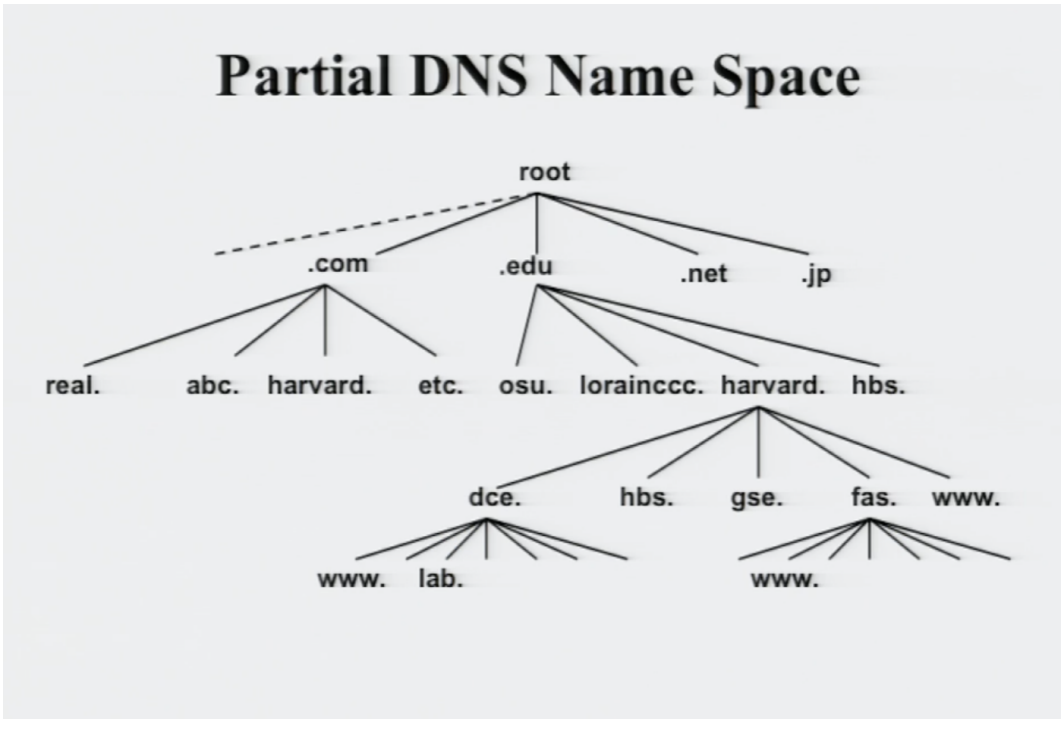
- The Muxing is done on the source port number -



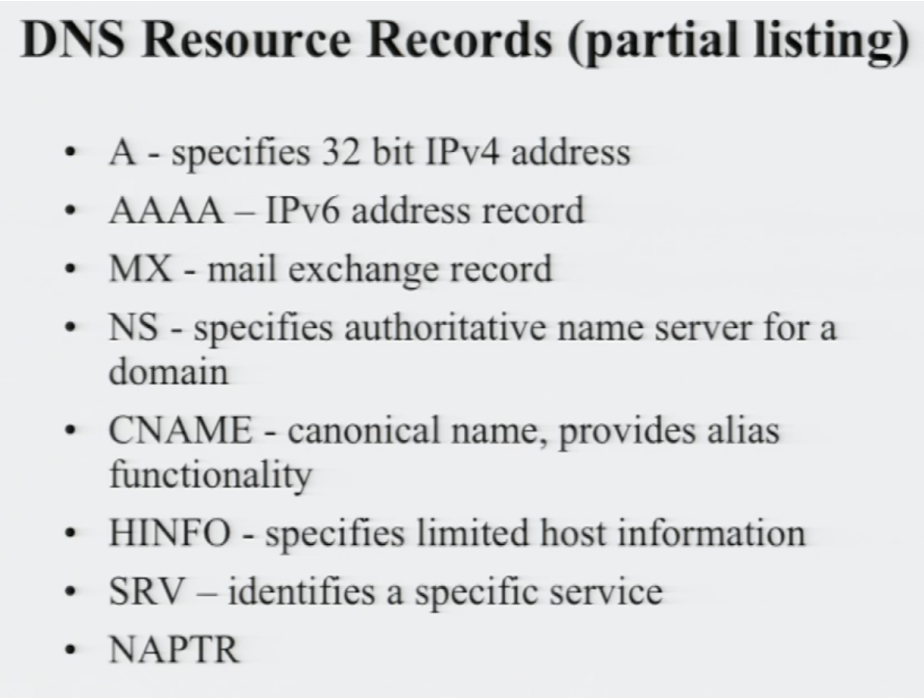
  - If a NAT box dies all connections will be broken
  - If a router crashes, it's not going to reset any state information

- STUN/TURN/ICE:
  - Know what these do (Lecture 9 7:05)
- DNS:
  - Application Layer Protocol
  - Rudimentary form initially (Emails and IPs mailed out in book)
  - Hierarchical name space to preserve naming uniqueness
  - Decentralized Distributed Database to hold all records

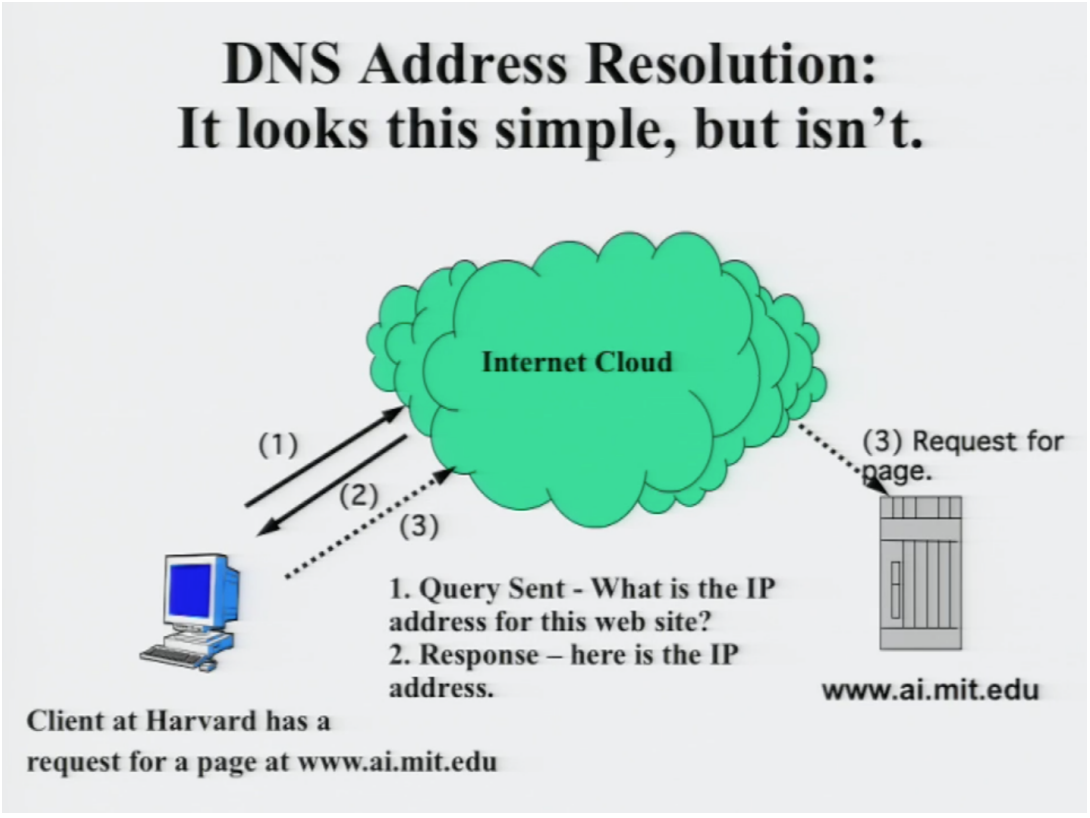
- 

- 

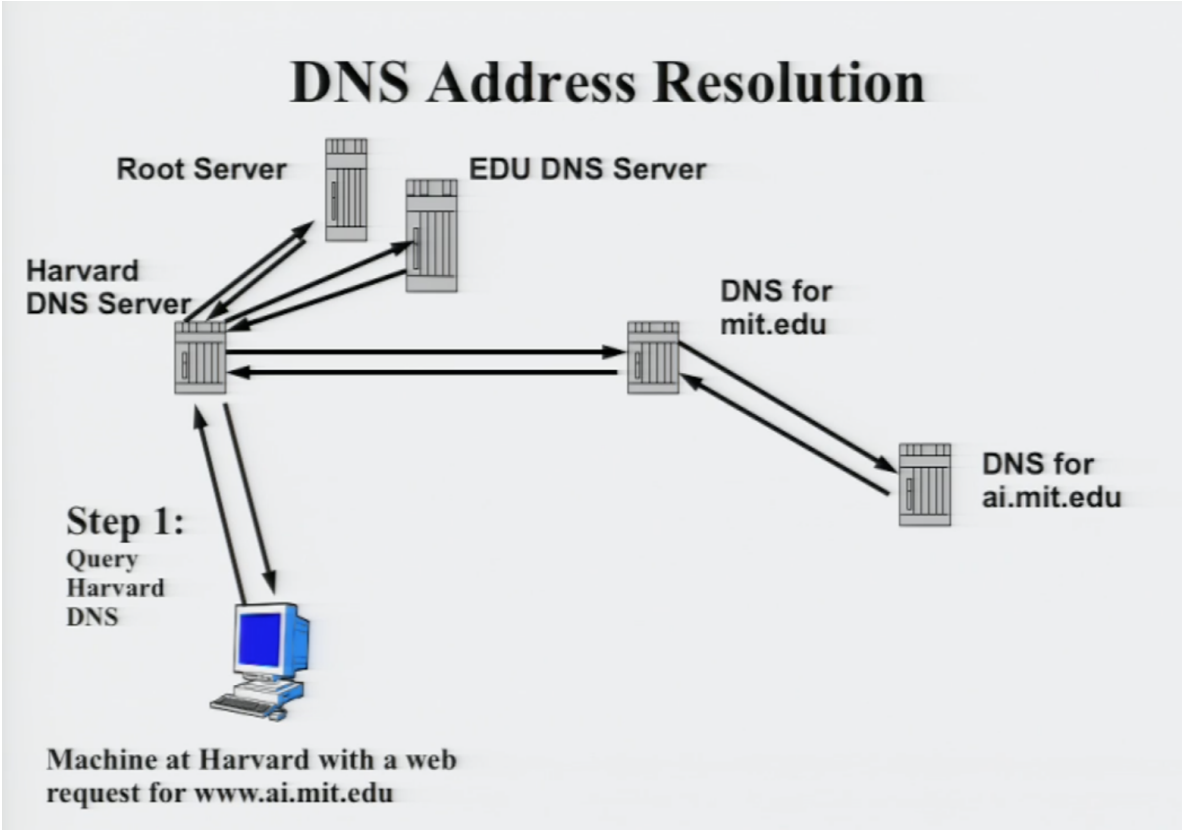
- TLD (Top-Level Domain)
- Each branch in the above diagram is a "zone" a.k.a. A unique portion of the namespace
- 1500 TLDs now
- DNS Servers map between domain names and IP addresses
- DNSSEC is an update on DNS that provides security and authentication
- `dig` will tell you what your local DNS knows about a given domain
- AAAA record is for an IPv6 Address (4 times larger than IPv4, a sick joke hahaha)
- DNS resource records:



- DNS helps with load sharing
- There are certain times where we don't want to have to know the "name" of the machine we want to talk to
  - MX records! We want the Mail Server for Harvard.edu we can get back a name that maps to an ip without knowing the full "name" of that machine
  - `dig yale.edu mx` Yale has silly tea names for subdomains of their mail servers
- Address Resolution:
  - It seems simple, but it isn't

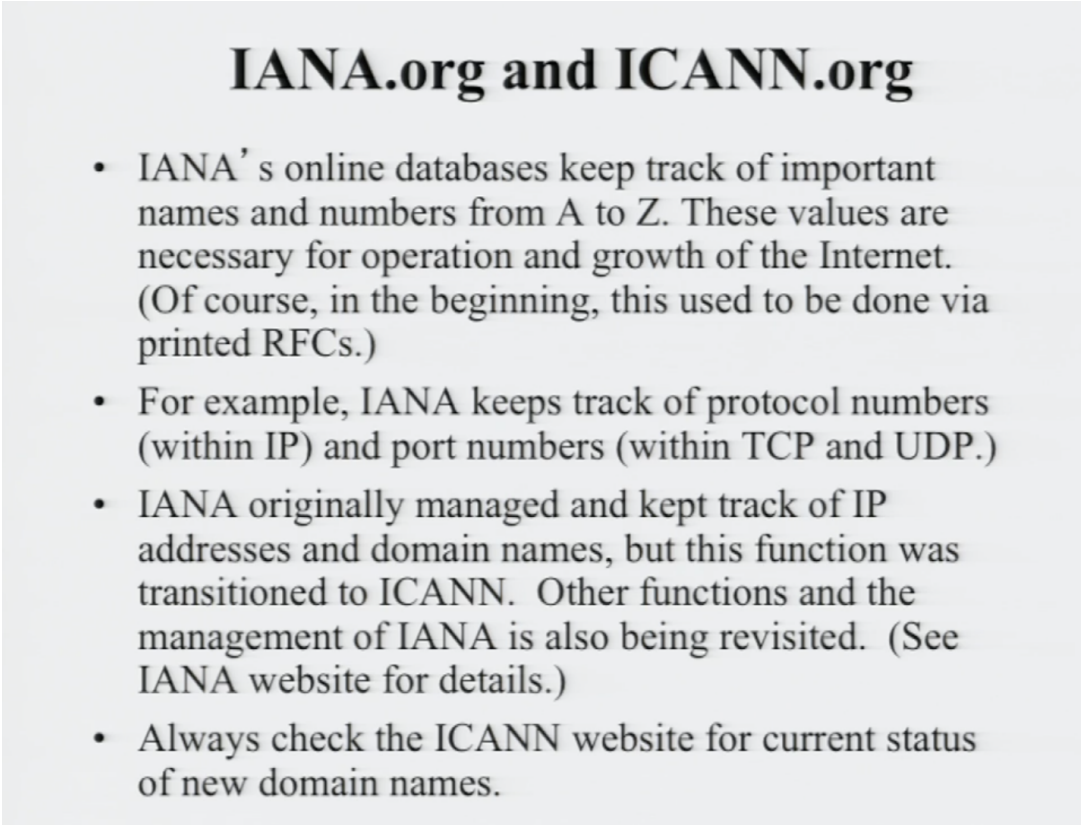


  - The client only needs to know the IP of the DNS server it has been configured to use
  - A DNS server is called a resolver
  - Uses TCP & UDP for different things
  - In order to answer a query for a host outside of its own zone, it must ask other DNS servers for that information



- `dig +norec www.csail.mit.edu +trace`

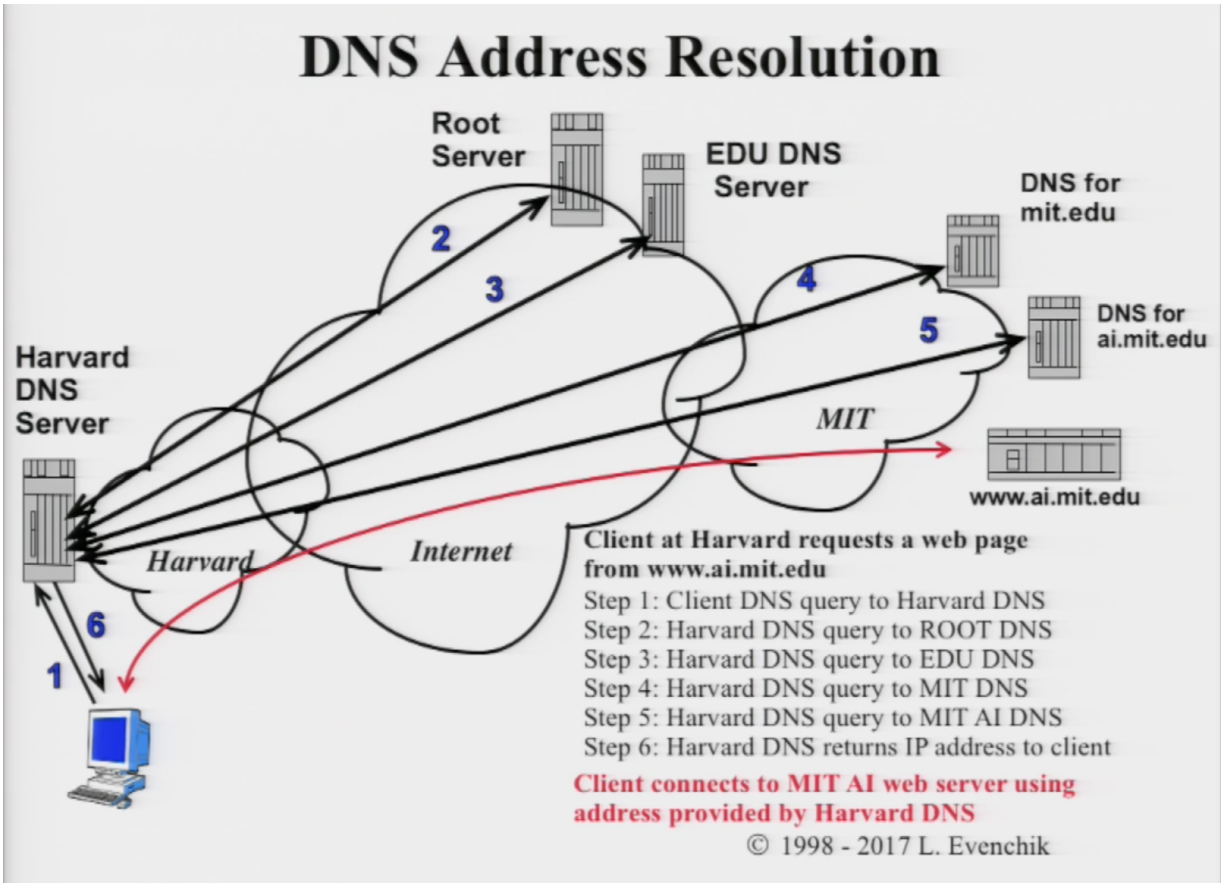
- Root servers only know about TLD IPs
- TLDs only know about the next level down in their namespace
- An organization's DNS servers must know of at least one root-level domain to begin using DNS
- www.root-servers.org
- 13 named root servers but hundreds of machines
- Root Zone File
- IANA vs ICANN:



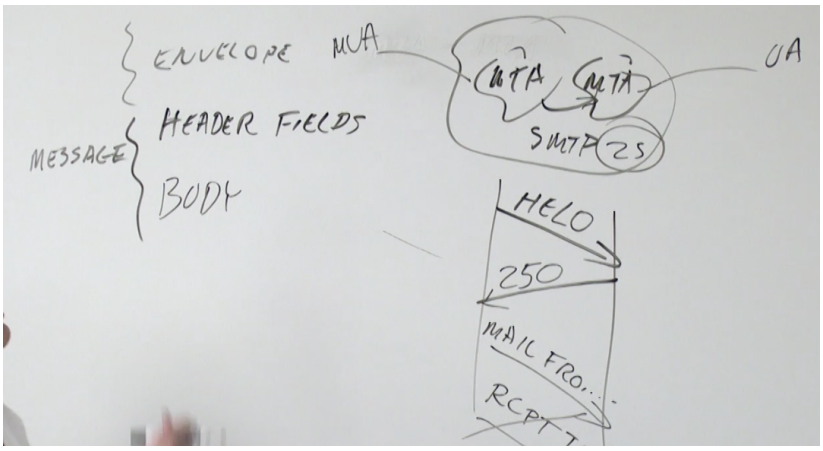
- ICANN expanded TLDs in 2000
- gTLD in 2008
  - .me .cool etc.

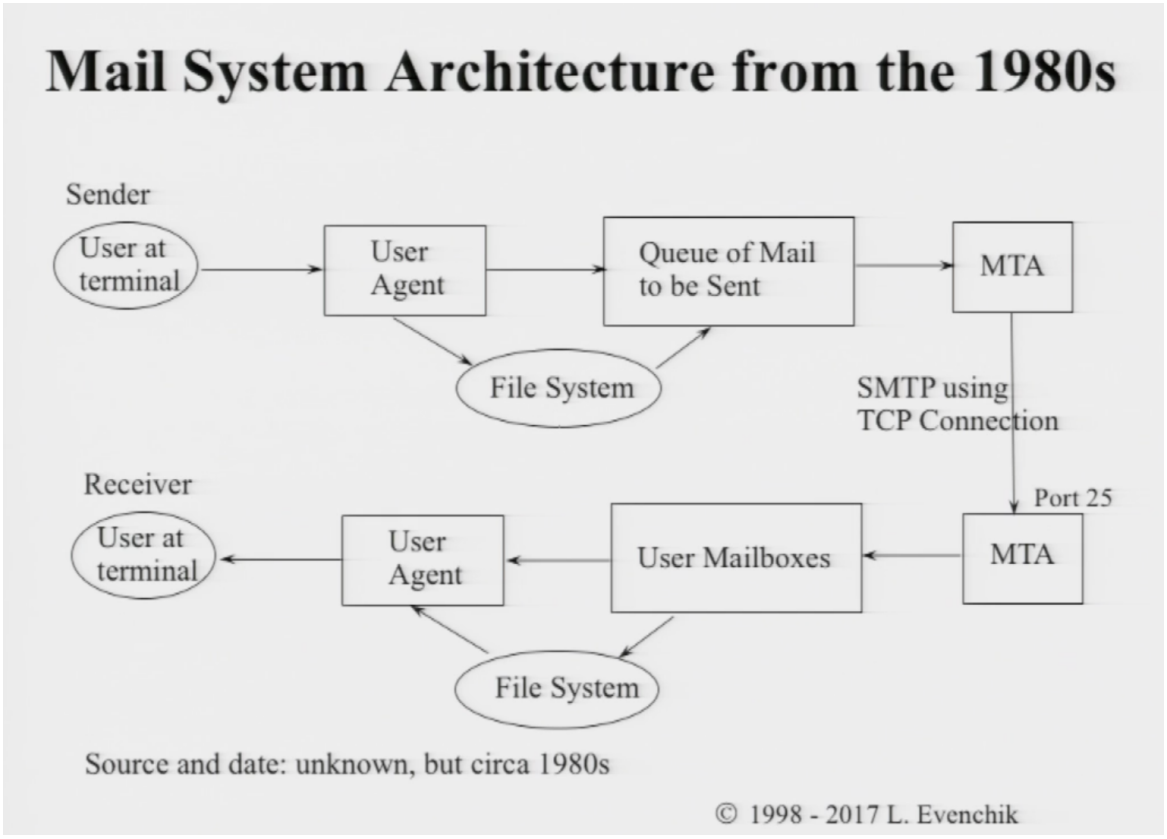
Lecture 10:

- DNS Zones
- DNS Address Resolution

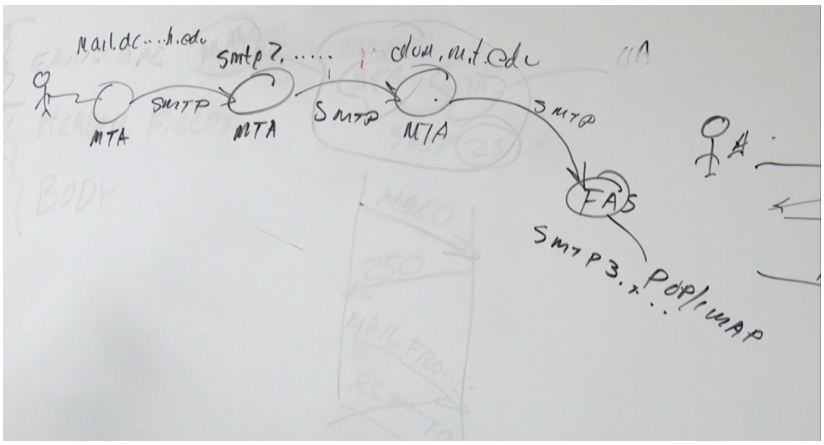
- 

- DNS TTL: How long something will be cached at the local resolver
- SMTP (Electronic Mail):
  - RFC 5321 & 5322
  - Application Layer protocol
  - TCP port 25
  - A specific protocol for the delivery of email messages
  - There is also a set of protocols for formatting those email messages
  - Internet came after email
  - MTA (Mail transfer agent) Email is sent between MTAs using SMTP
  - 3 parts:
    - Envelope
    - Headers
    - Body
  - The headers and the body make up the actual email message
  - SMTP mail servers are discovered using MX records in DNS
  - Is a very simple protocol

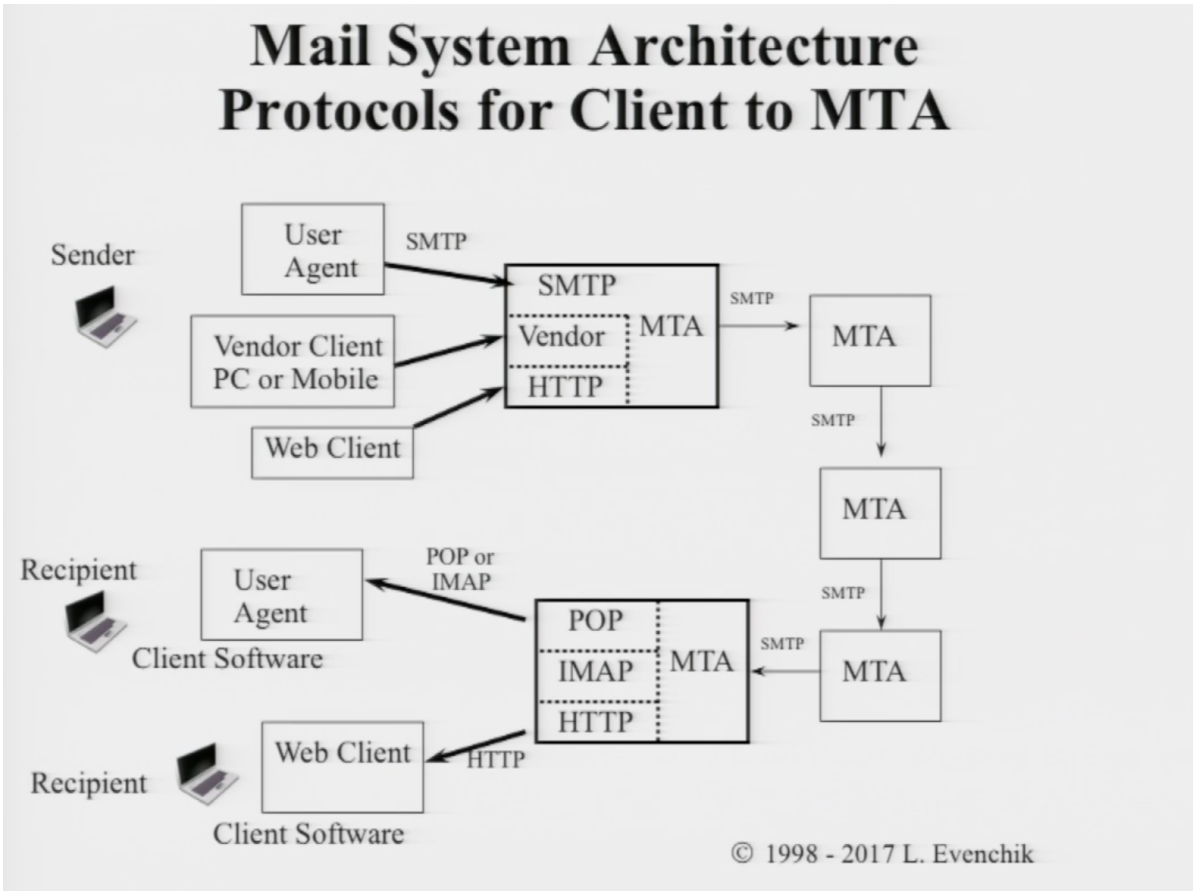
- 

- 

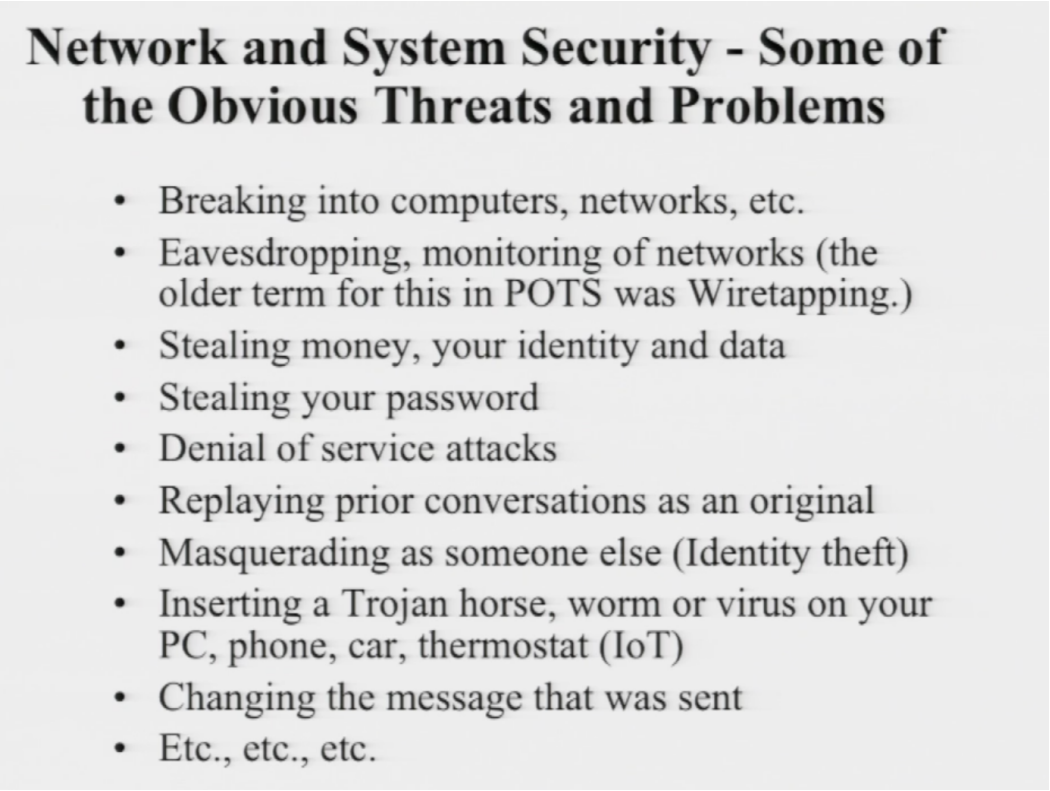
- Simplified SMTP Procedure (Lecture 10 23:00)
- 3 digit response codes (Idea has been carried over to things like HTTP)
- TELNET:
  - Very simple application layer protocol to which you can setup a port and open a connection to any device on the network that is configured to accept it
- `telnet mail.dcd.harvard.edu 25`
- Email delivery problems:
  - Its up to the mail system to alert the user or keep the info to itself
  - The sending MTA will try periodically to redeliver on failure
- MTA forwarding:
  - Lecture 10 50:00



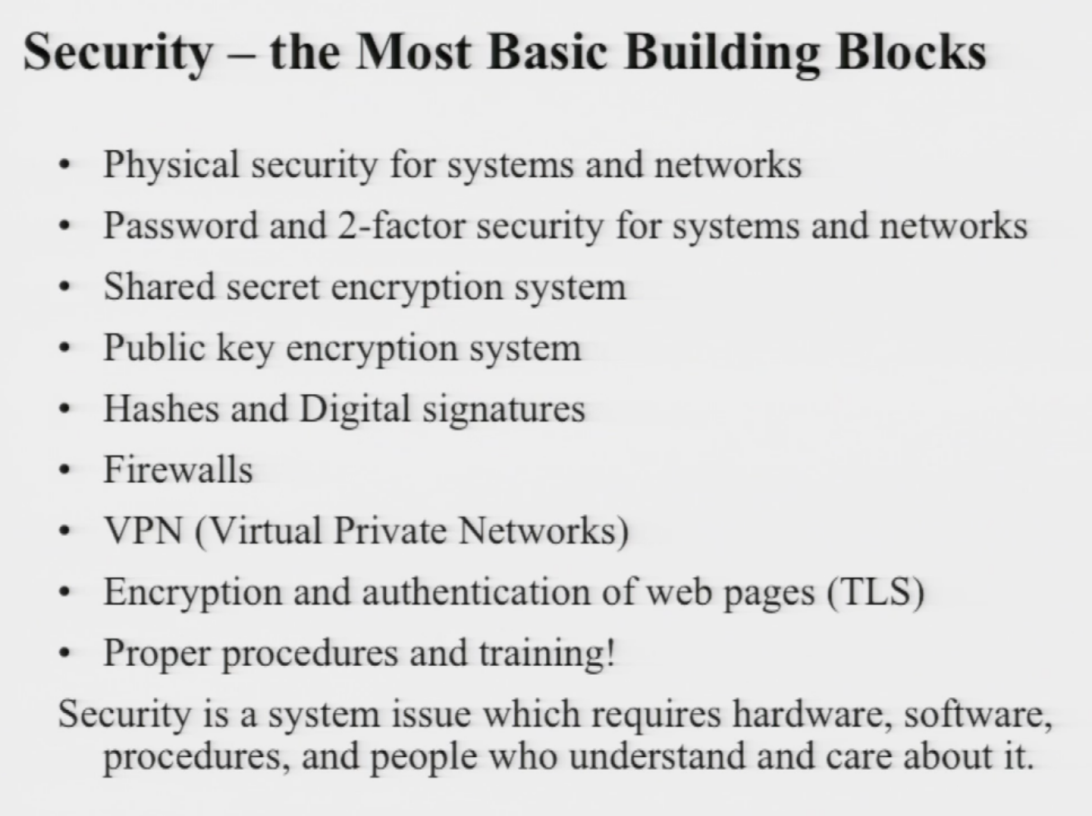
- Email is anything but reliable
- Message Disposition Header Field:
  - Used for Email delivery notification
  - Handled by the receiving mail client
- MIME (Multipurpose internet mail extensions):
  - Defines additional message headers in email (and now utilized in the web)
  - Content-Type
  - Content-Transfer Encoding
  - Is really just a way to deliver non- text information
  - IANA
- Ways to send mail to an MTA:



- Network And System Security:
  - Hard area of work to understand
  - Security has to be considered everywhere in the network!
  - Some Obvious types of threats:



- A very common approach to security is known as the InfoSec triangle
  - Confidentiality
  - Integrity
    - The data hasn't been altered
  - Availability
- Basic Security building blocks:

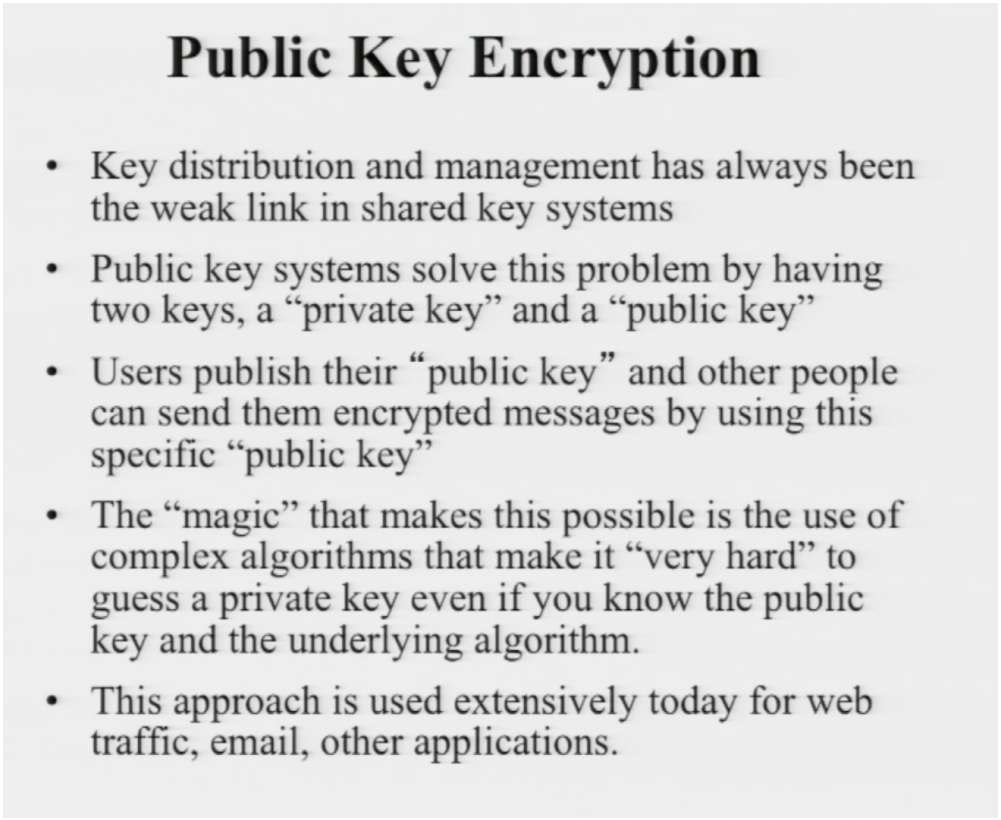


- Security Requires:
  - Hardware

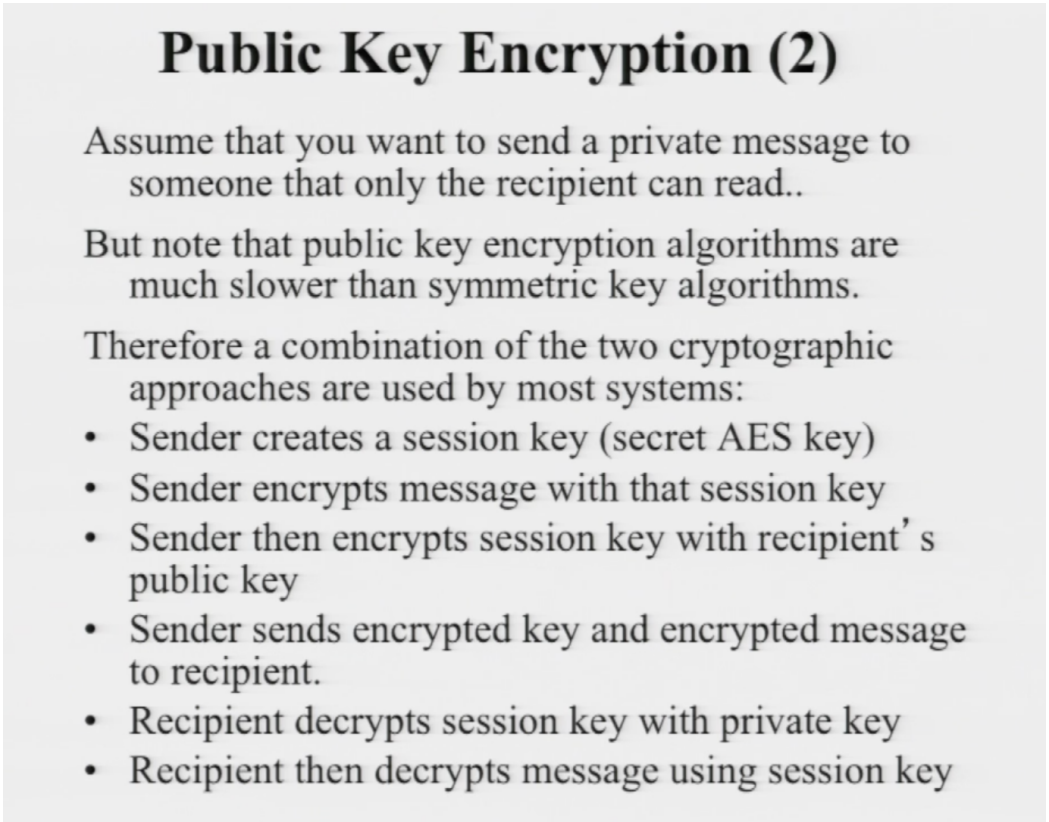
        - Software
        - People
        - Processes
        - People Educated on what security means and how to do it
    - Five important elements of structured security:
        - Privacy and confidentiality
        - Authentication
            - Are you who you say you are?
        - Authorization
            - Are you allowed to do a certain action
        - Integrity
        - Nonrepudiation
            - Can you prove that someone did or didn't do an action?
    - Cryptography:
        - Two common approaches
        - Symmetric:
            - Same key is used to encrypt and decrypt
            - AES, IDEA
        - Asymmetric:
            - Key pairs, one public one private
            - Data encrypted by one key must be decrypted by the other key
            - Diffie Hellman & RSA
        - Key length determines the strength of the encryption
        - Public/Private key encryption:
            - `ssh-keygen -t rsa`
            - How do you use it to send a secure message?
            - How can you use it to authenticate someone in the conversation?

Lecture 11:

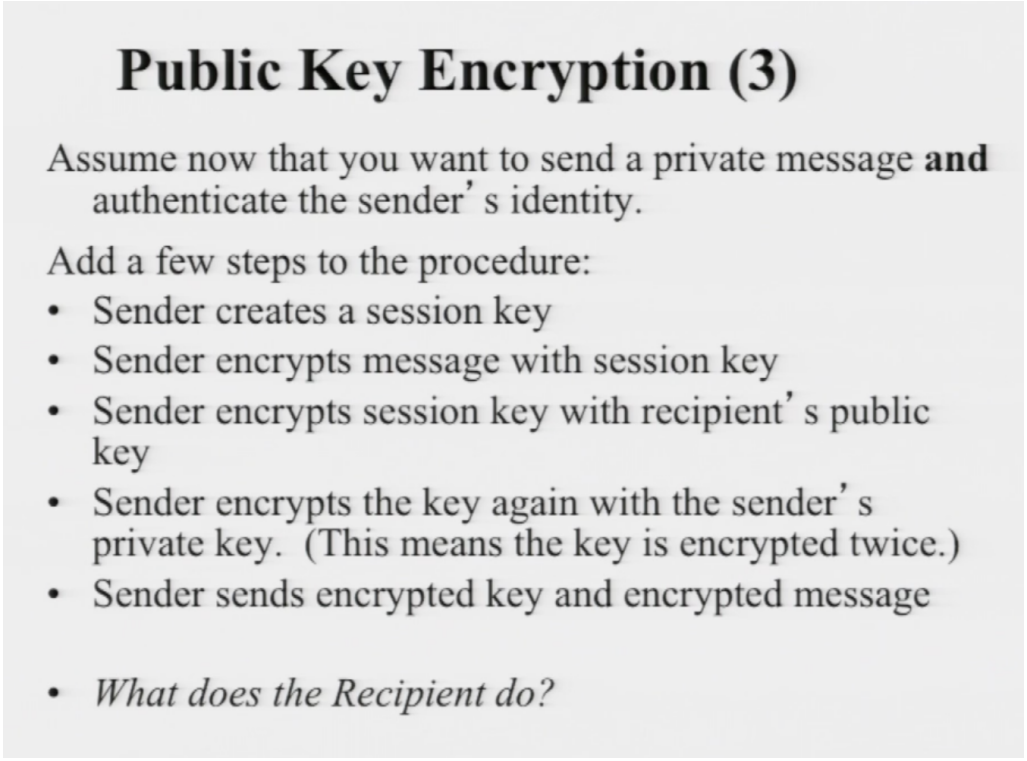
- Understand the difference between how email can be sent between MTAs
- Understand the envelope and the content (body & headers)
- Understand POP3 and IMAP
- Symmetric vs Asymmetric cryptography
    - Symmetric is much faster
- AES:
    - The algorithm is not the secret! The key is!

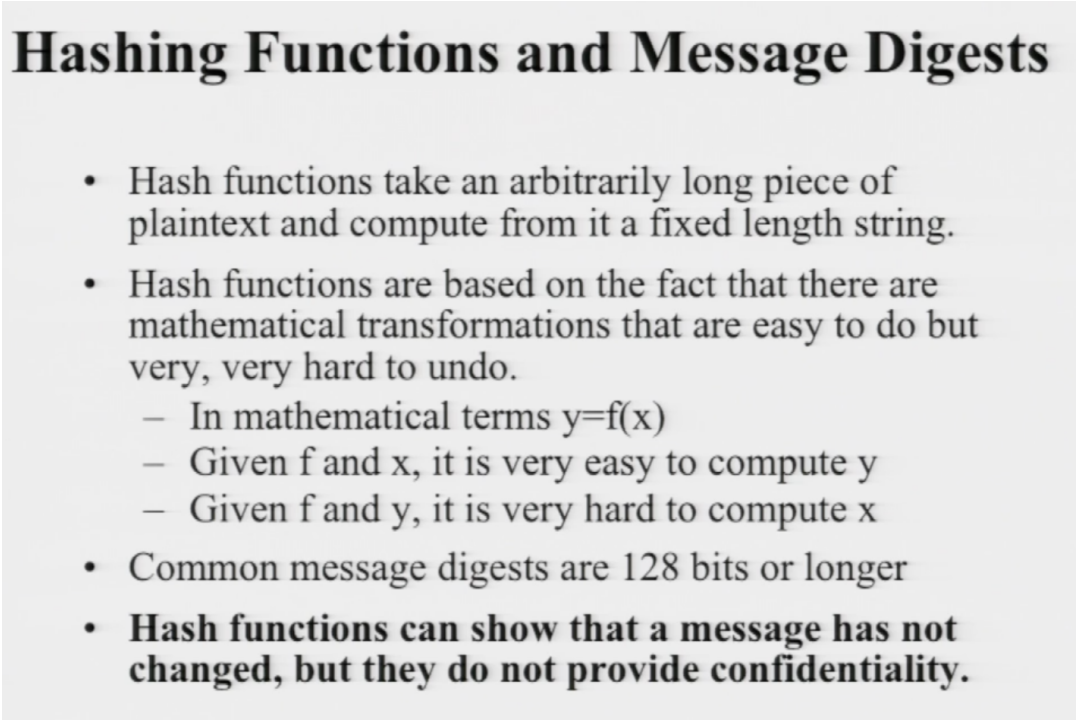


- Session keys Save a ton of time units!!!
  - Encrypt with whoever's public key that you're trying to talk today
  - Receiver can then decrypt the session key and then decrypt the larger message with the session key
- This approach is because asymmetric approach is slow!

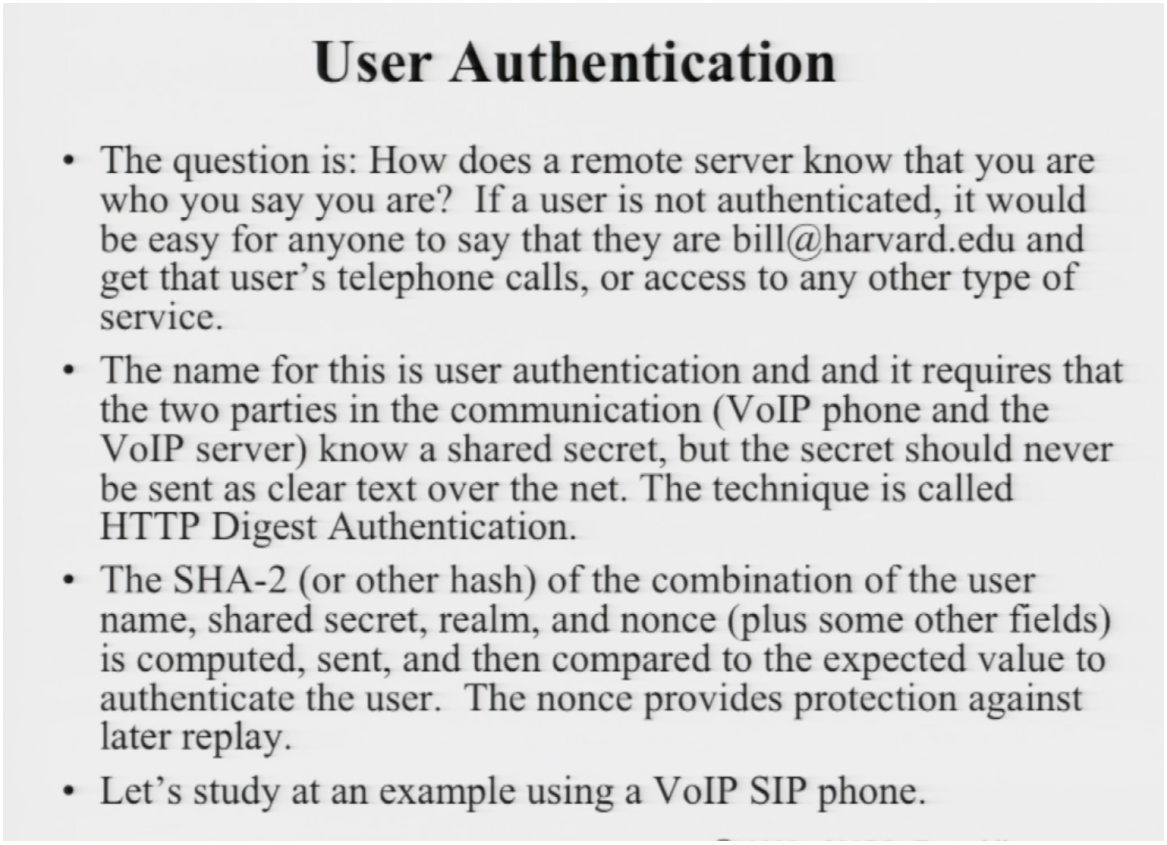


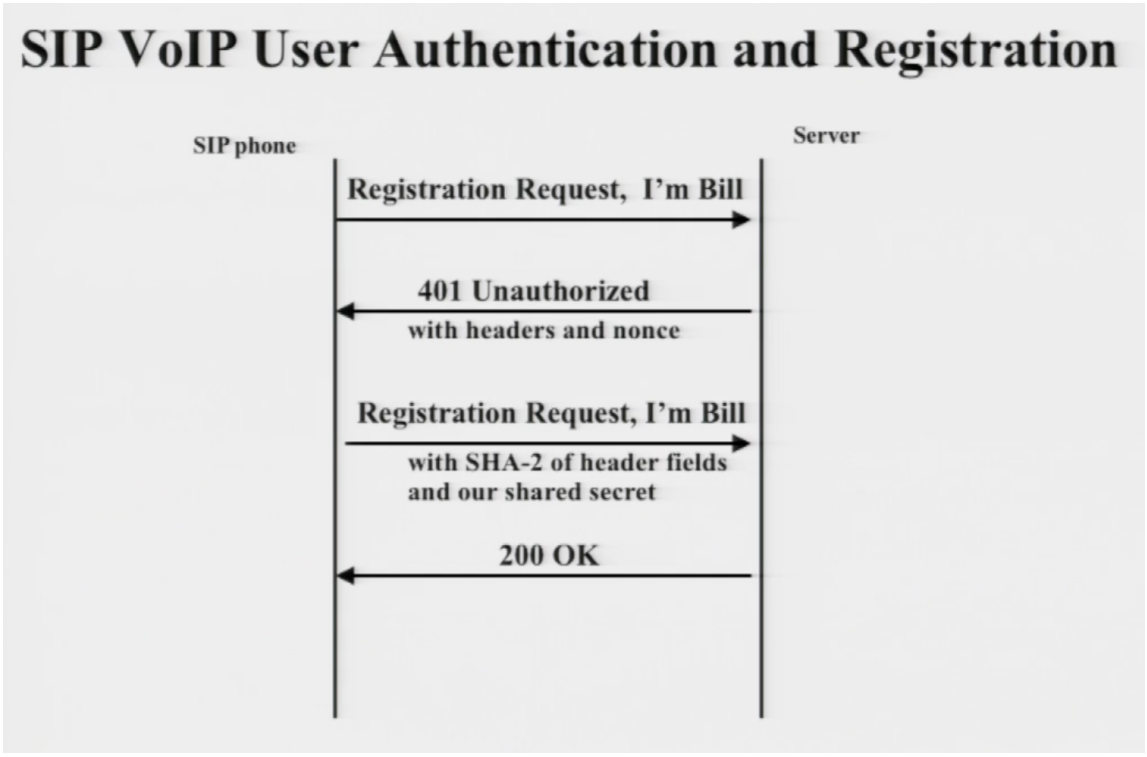
- Public key encryption does not provide authentication by default!
- Encrypt session key with my private key, encrypt again with friends public key. Friend comes along and decrypts with his private key. He then decrypts with my public key to get the session key! He can then go ahead and decrypt the large file.



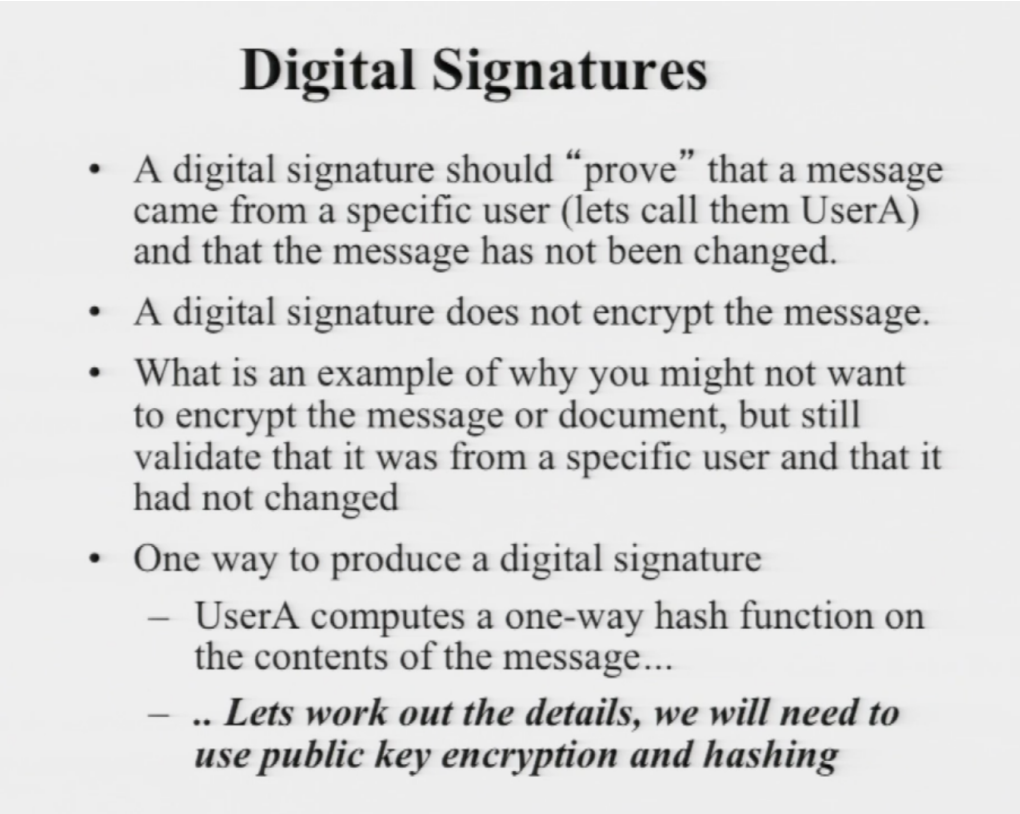
- Hash Functions:
  

  - a one way mathematical function
  - MD5 etc.
  - Produce a "message digest" much smaller than the original data
  - Provide integrity check, but does not provide confidentiality
    - Lecture 11 42:00
  - Reverse hashing
  - Rainbow tables
  - Do not store the user password hashes! Common password hashes are known!
  - Add something extra instead and then hash
- User authentication (Lecture 11 1:06:40)

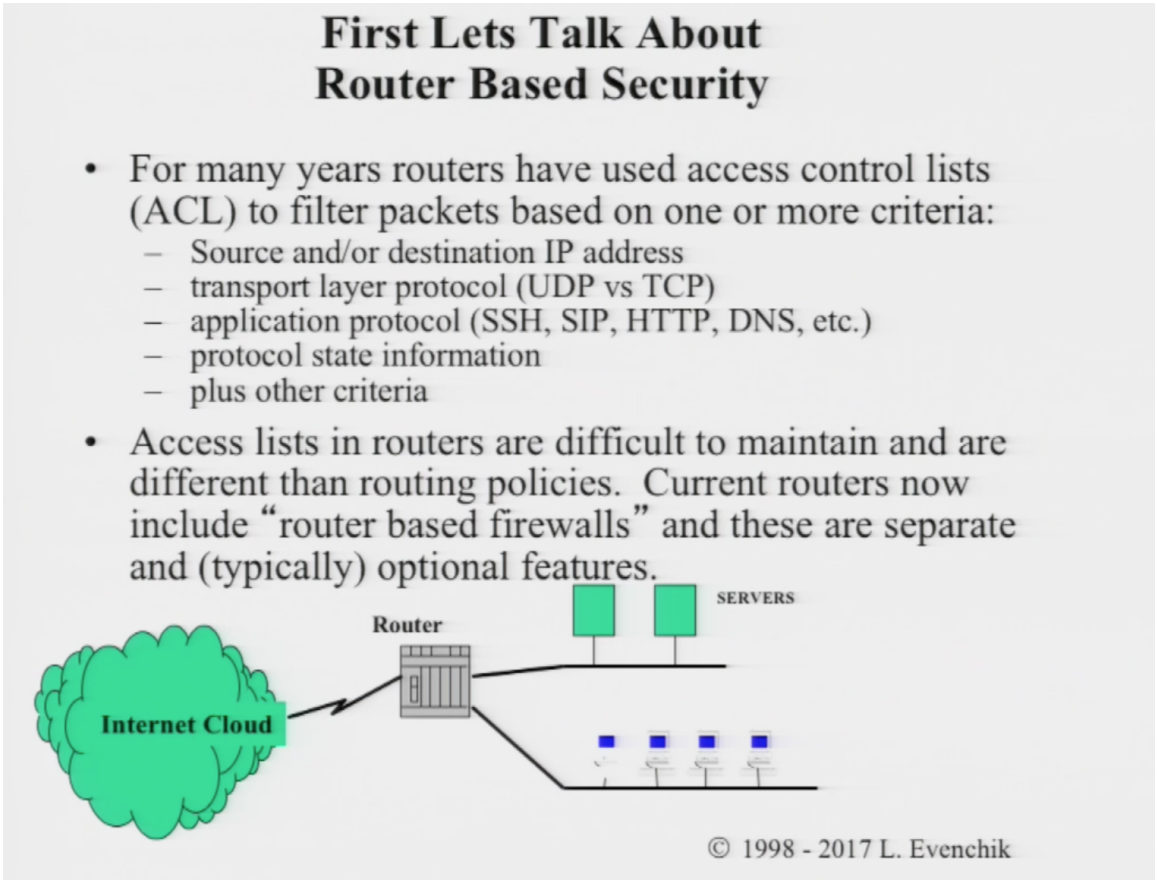
- 

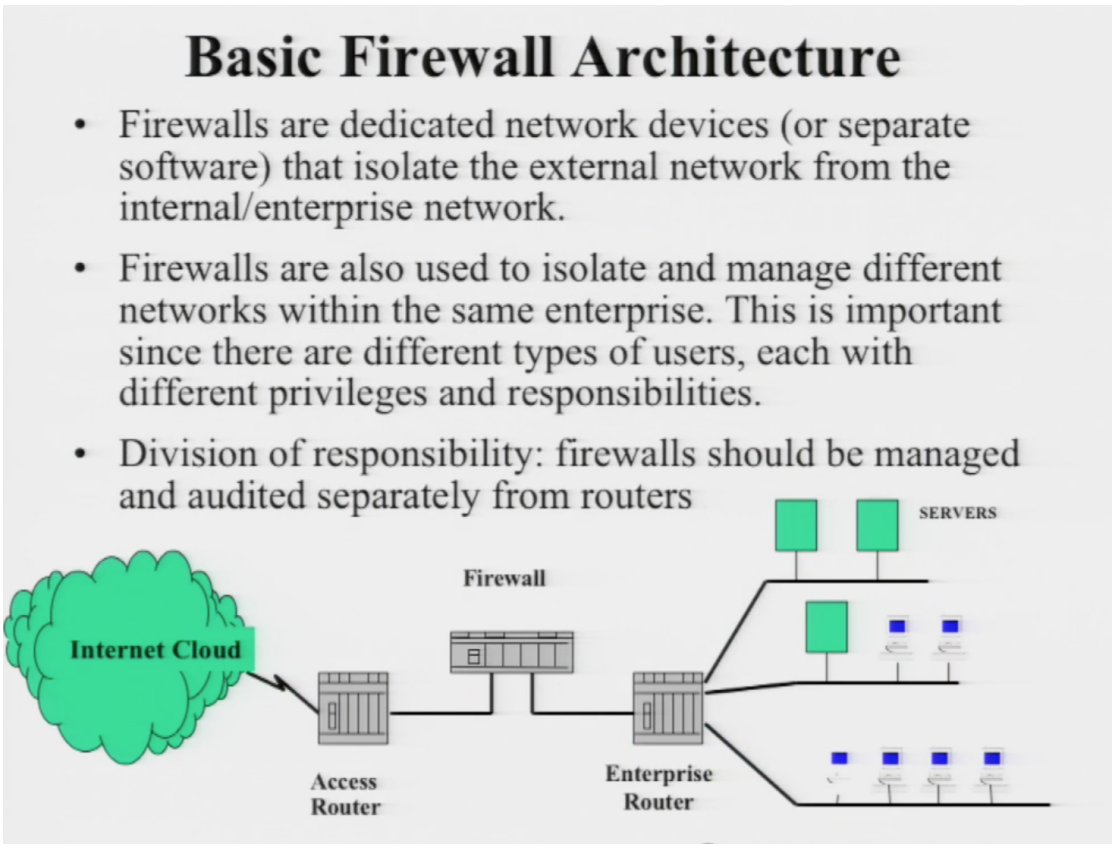
- 

- Authorization with response to challenge (Lecture 11 1:09)
- Why do we need a nonce in this example?
  - A bad actor could capture the sender and secret and "replay" the message later on. The random number from the authenticating server is crucial! The bad actor's "replay" attempt would have a different nonce!
- Digital signatures:
  - Should prove that a message should come from a specific user (authentication) and that it has remained unchanged (integrity)

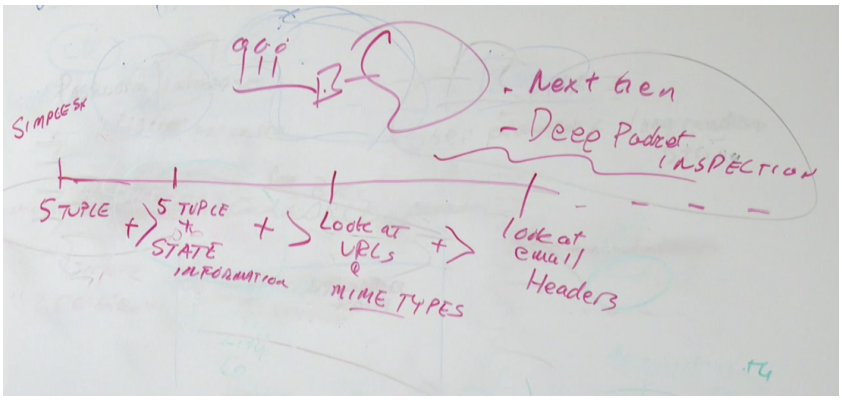
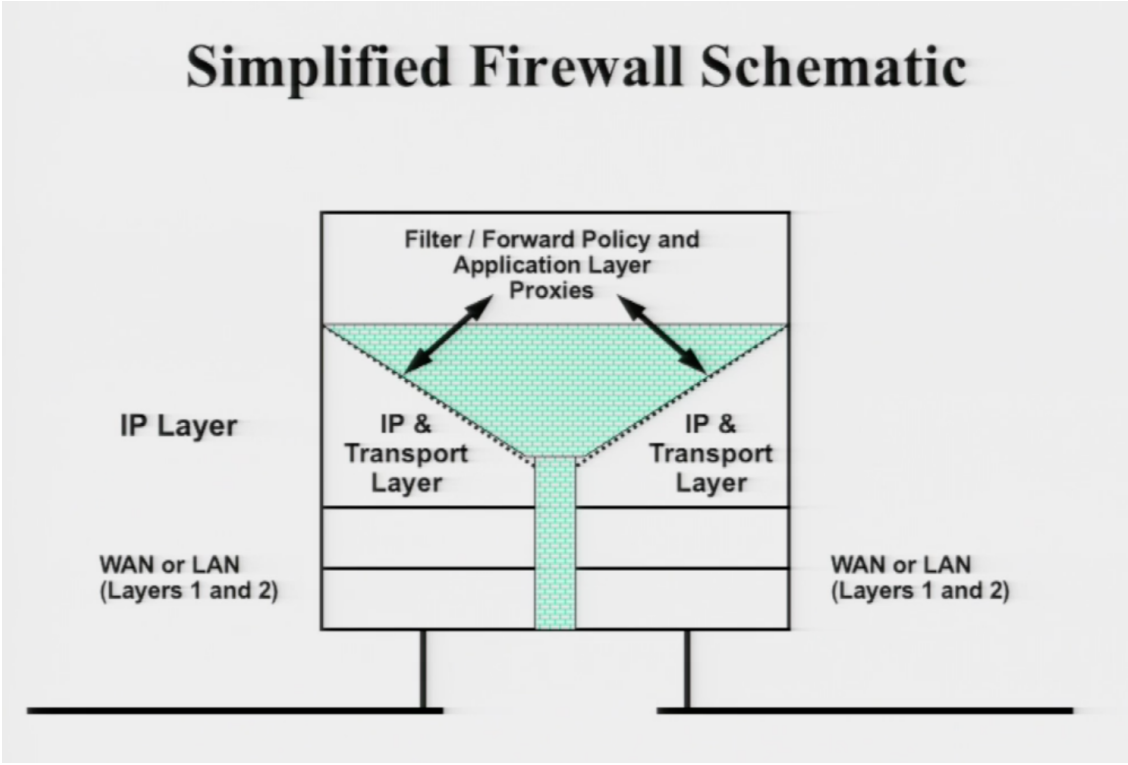


- Security provided by routers and firewalls
  - ACL (based on the 5-tuple)
  - Router filters and forwards packets
  - Could simply state in config that you don't want a specific type of traffic (web, email, etc.)
  - Firewalls look at a packet and make a decision to let through based on many metrics
    - 5 tuple
    - MIME types
    - Email headers

- 

- 

- When you think of Firewalls think of 5-tuples!
- Firewall functionality (Lecture 11 1:29)
- Inbound/Outbound rules
- Can be simple or complex:

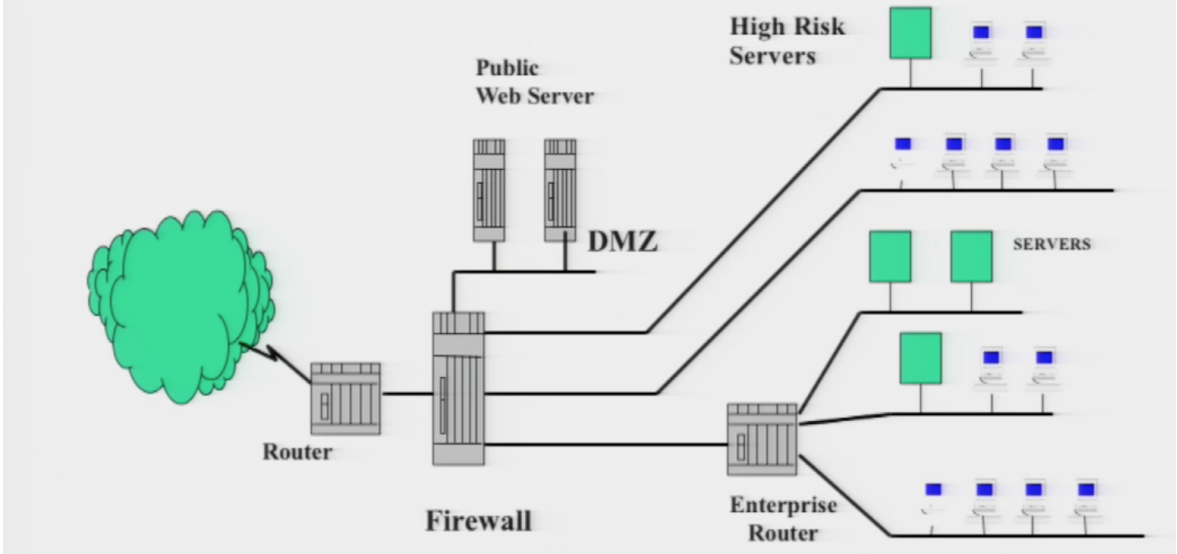
- 
SIMPLE SK
5 TUPLE
5 TUPLE
STATE
INFORMATION
Look at
URLs
MIME TYPES
look at
email
Headers
- Next Gen
- Deep Packet
INSPECTION
- 
Simplified Firewall Schematic
Filter / Forward Policy and
Application Layer
Proxies
IP Layer
IP &
Transport
Layer
IP &
Transport
Layer
WAN or LAN
(Layers 1 and 2)
WAN or LAN
(Layers 1 and 2)

Firewall Functionality (part three)

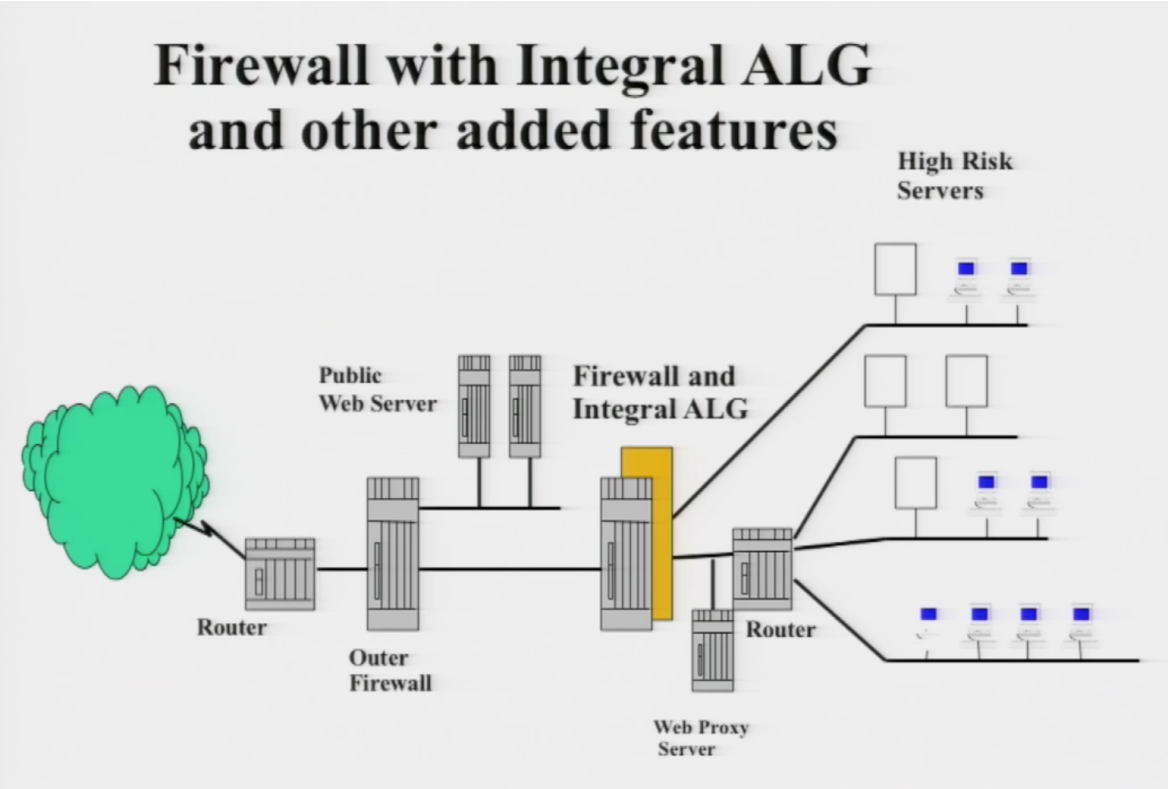
- Firewalls also provide NAT and service redirection
- Application Layer Gateways (ALG) can terminate application layer sessions for individual users and then create new sessions, depending upon security policy.
- Firewalls can provide secure tunnels with encryption for remote users (aka VPNs)
- Firewalls should provide extensive logging, reporting, management and alarms
- Firewalls features are being added constantly. For example: virus checking, spam filters, QoS, use of AI, etc.
- Firewalls can also be located at the ISP or in the Cloud. Managed security is also an option.
- Firewalls are not a complete security solution

- "Protect the edge of my network"
- SNORT Open Source Firewall software

Simple Firewall Architecture - option 2



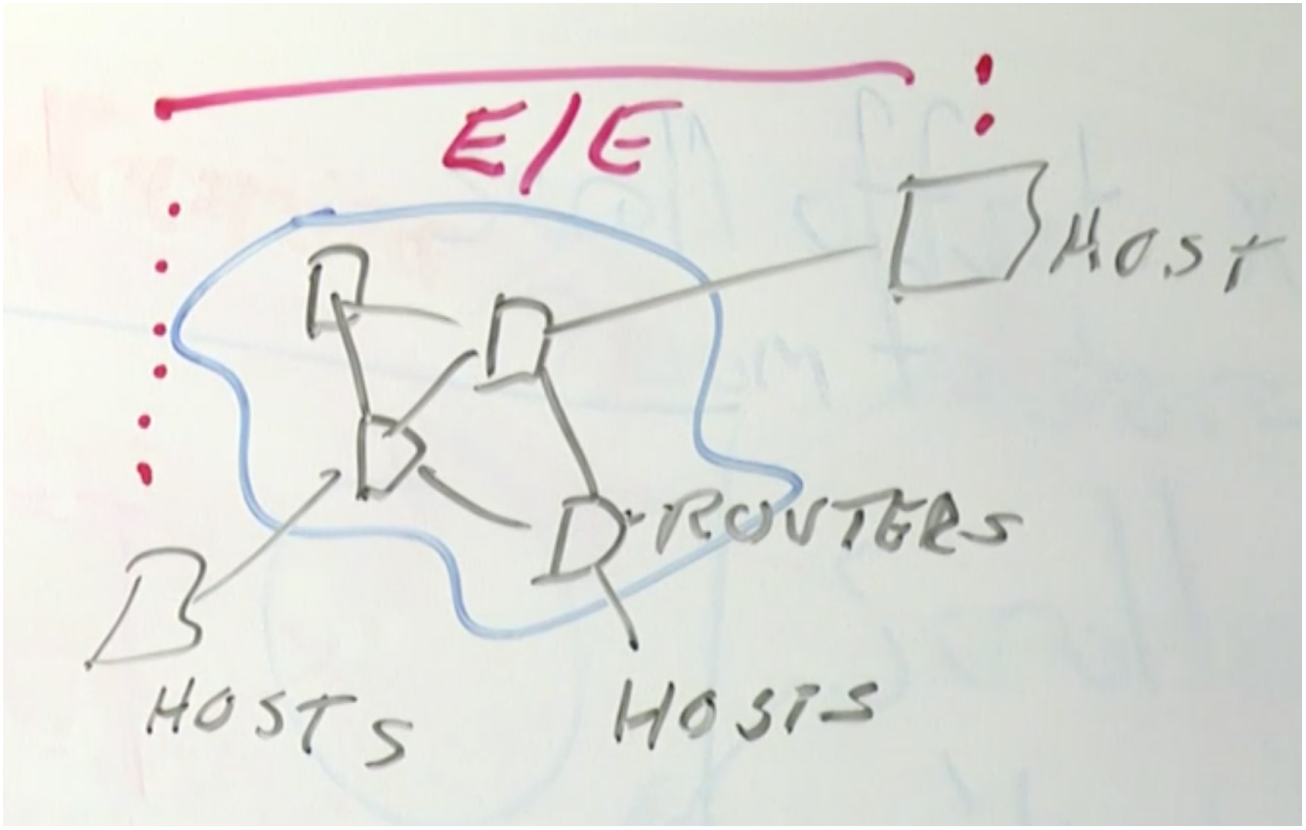
- Recommended to have 2 firewalls: an inner and an outer and that they come from different vendors
- The hope id that any security holes on one vendor would be addressed before the 2nd one is breached
- Firewall with ALG (application layer gateway):



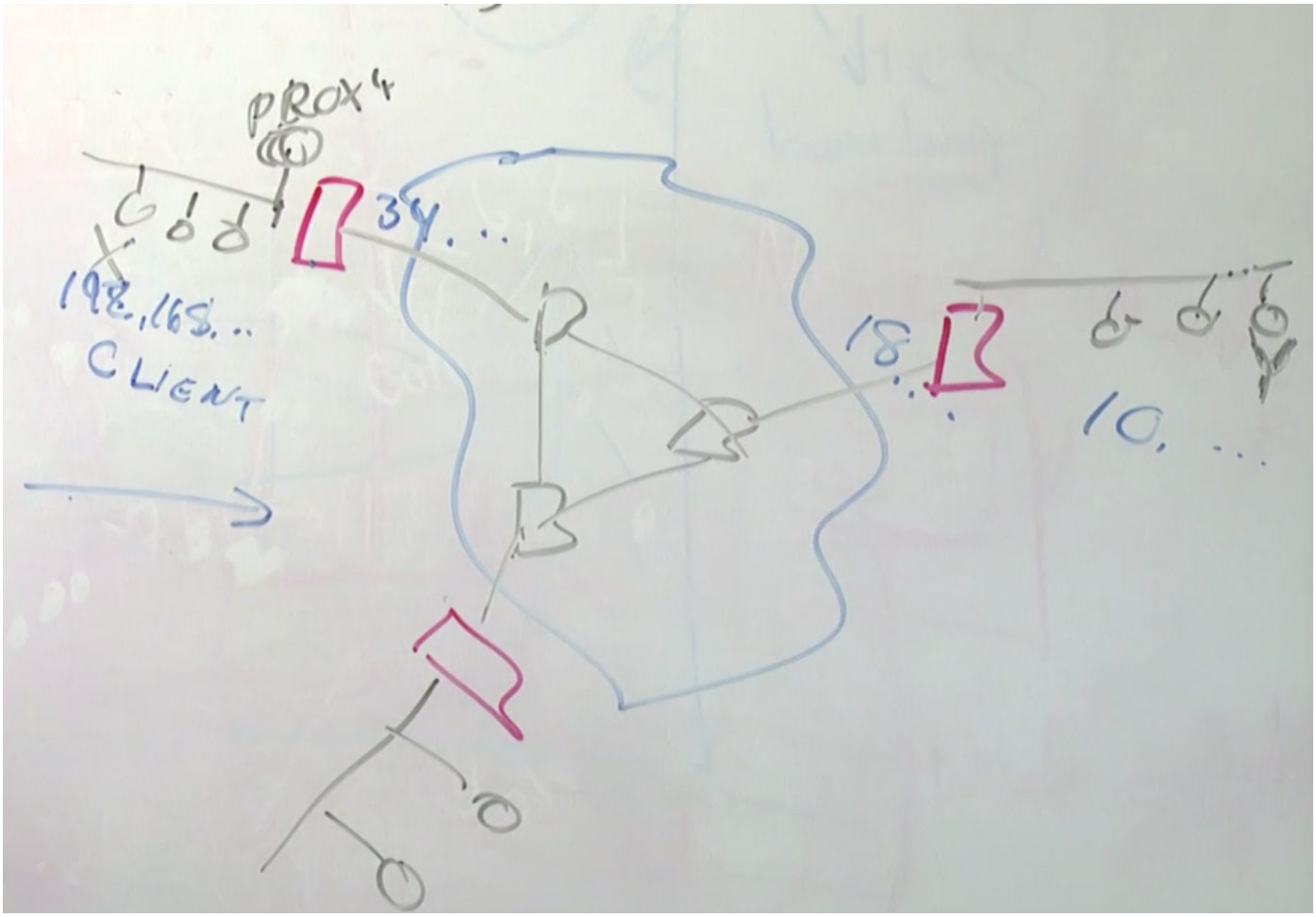
- IDS (intrusion detection system)
- A proxy can cache pages!
- VPNs and IPSec!

Section 4:

- Host vs Client vs Server etc.
- OS/ Net Stack/ Real World implementations
- TCP Congestion Control
- Default Host Config
- Routing Issues & Problems
- IP Functionality - Fragmentation
- End-to-end vs. Middlde Boxes (NAT, Firewall)
- Encapsulation
- Host/Client/Server used to be used distinctly
  - Devices in the inner network were very simple and had a single responsibility
  - Nowadays these terms can more or less be used interchangeably

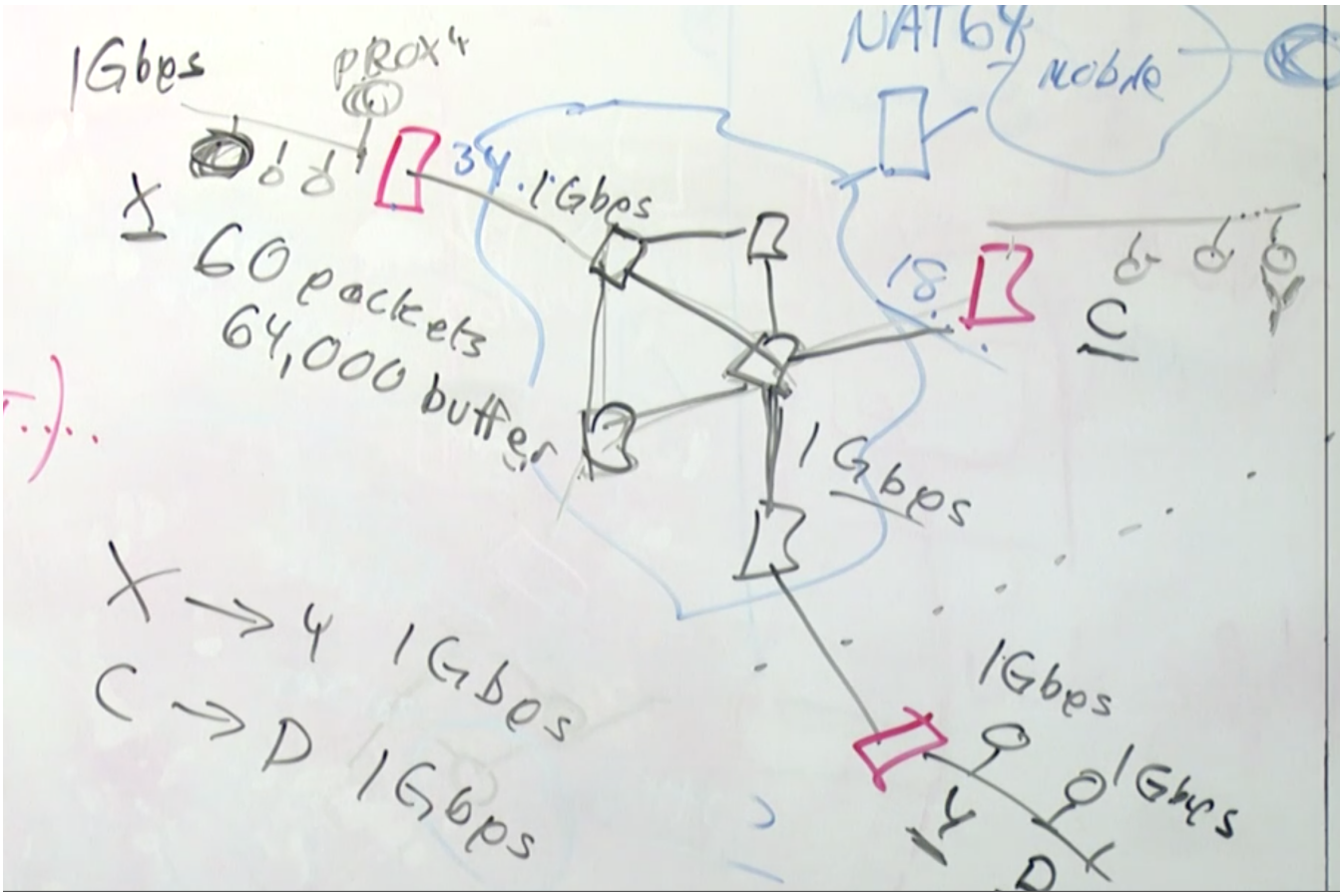


- End-to-end: connection directly from one IP to another
- A "Middle box" can break end-to-end connections:
  - Lets say a NAT box goes down and loses its NAT Table. This could cause a break in the end-to-end connection because a NAT box then won't know how to forward a request to private IP on it's other side

- 

- NAT64: NAPT for IPv6 devices
  - Mobile phones etc.
  - Appreciate the impact that could happen if one of these went down!
- IPv6 allows for auto configuration due to the fact that the address space is massive and the chance of a collision is minimal
- IPv6 only has "slash" notation no netmask like IPv4
- DHCP:
  - Client sends a broadcast out: "Hey someone give me an IP!"
  - DHCP Server sets that IP and send a responding packet to the client
    - Also provides the default gateway/netmask

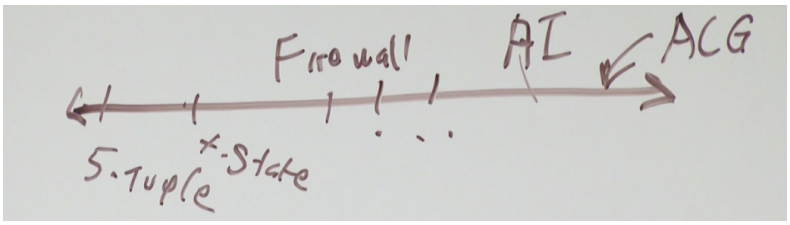
- TCP segment lifecycle and its interaction with the OS (Section 4 47:00)
- Fragmentation
  - In a perfect world, frames would be of infinite size, but this is not the case
  - This is not the case
  - Its unrealistic for a designer to assume that all applications will know what the MTU (Max transmit units) will be for different networks
  - Fragmentation will rarely happen at the host (section 4 53:00)
  - Large Frame Network vs Small Frame Network
  - IPv6 sends out PROBE packets that are smart enough to check the intervening paths of the network and not create any packets that couldn't be handled
    - This is the responsibility of the Receiving Host
- TCP Congestion Control
  - Flow Control: The receiving system controls the sending rate (Think TCP connection window size) Is end-to-end
  - Two Hosts overwhelming the available bandwidth of the inner network:

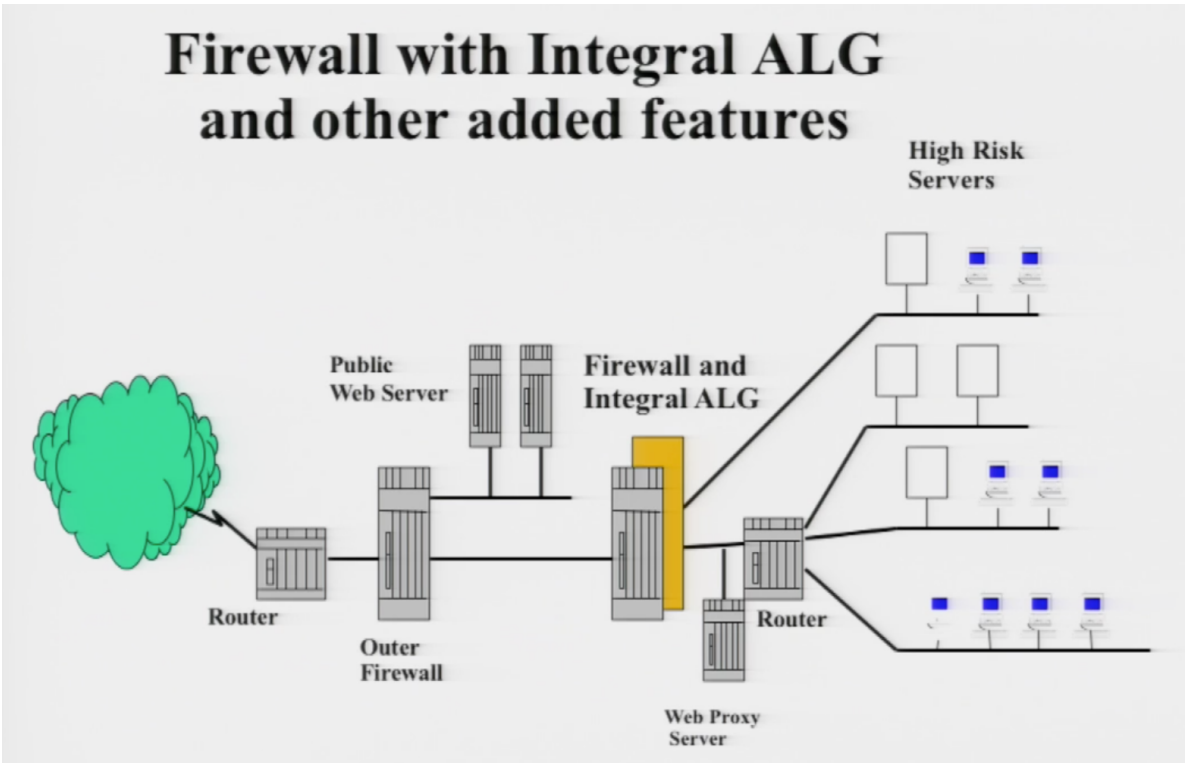


- "Slow Start"

Lecture 12:

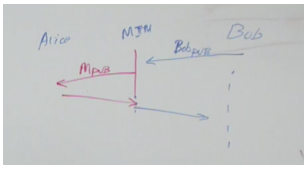
- Digital signatures
- Website Security Certificates and TLS
- Firewalls and VPN
- VoIP
- Session Initiation Protocol (SIP)
- CIA Triad
  - Hardware
  - Software
  - Written Procedures and processes
  - Educated people who care
- Five Important Elements of Security (ON FINAL EXAM):
  - Privacy and Confidentiality
  - Authentication
  - Authorization
  - Integrity
  - Nonrepudiation
- Firewalls

- Homework question about ACLs (the underpinning for the Firewall GUIs from lecture)
- Can reside on prem or be a managed service (ISP, third part etc.)
- Firewalls can be very simple to vey complex
  - 

- ALG (Application-Layer Gateway):
  - An extreme form of deep packet inspection
  - Basically used because you don't trust anything about the client or end system so you parse all traffic at any level you want
  - Terminates requests and creates them anew with some visage of the original information inbound and outbound
  - Usually sits at edge of network w/ Firewall or part of Firewall:
    - 

- Certificates and Website Security
  - X.509 Certificate
  - A culmination of Asymmetric Cryptography (public/private keys), Hashing (think integrity), and digital signatures
  - Digital signatures ensure integrity in data being sent as well as assuring that the sender is who they say they are (authentication)
    - Producing a digital Signature:

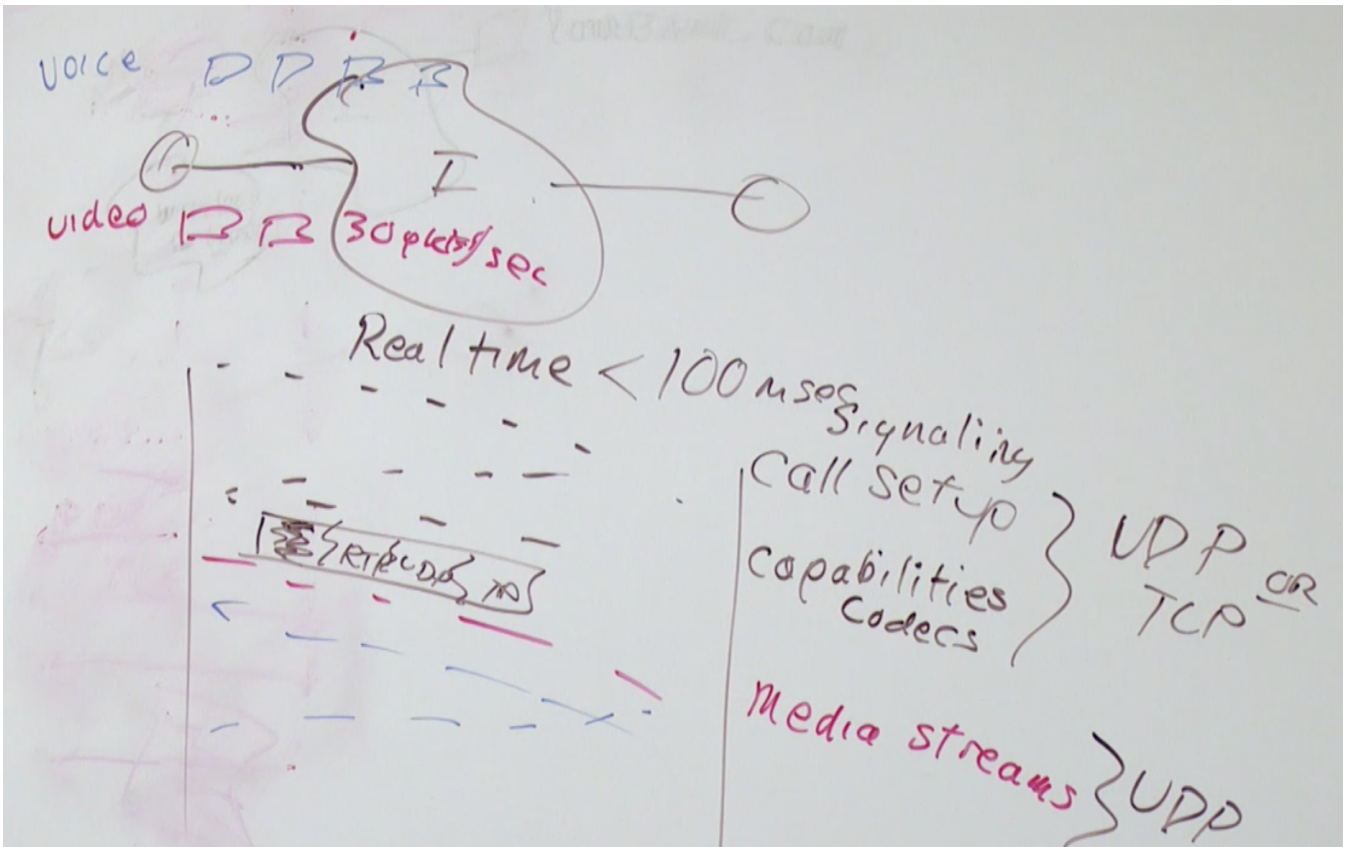
- One way to produce a digital signature
  - UserA computes a one-way hash function on the contents of the message
  - UserA encrypts the hash code using their private key
  - The encrypted hash code is appended to the message and the combination is sent to UserB
  - UserB computes the same hash function on the contents of the message
  - UserB then decrypts the received hash code with UserA's public key
  - If the hash codes match, the message came from UserA and the message was not changed in transit

- We want to accomplish Authentication, Integrity, and Confidentiality
- Man in the middle (MITM):
  - 

- Avoiding a MITM attack:
  - Using a CA (Certificate authority)
  - Bob and Alice both trust the CA and have its public key loaded on their computers
  - Bob creates a key pair and sends his public key to the CA in a secure manner
  - The CA confirms that the public key is coming from Bob and then digitally signs a Certificate that includes Bob's public key and some extra info
  - The CA sends this certificate back to Bob
  - Alice wants to send a secret message to Bob now
  - Bob sends his certificate which has his public key and has been digitally signed by the CA

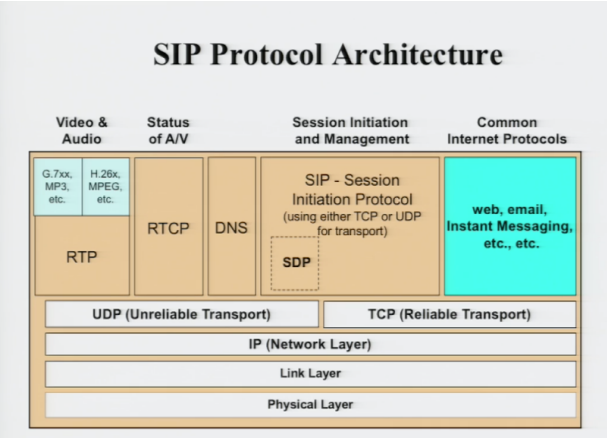
Lecture 13:

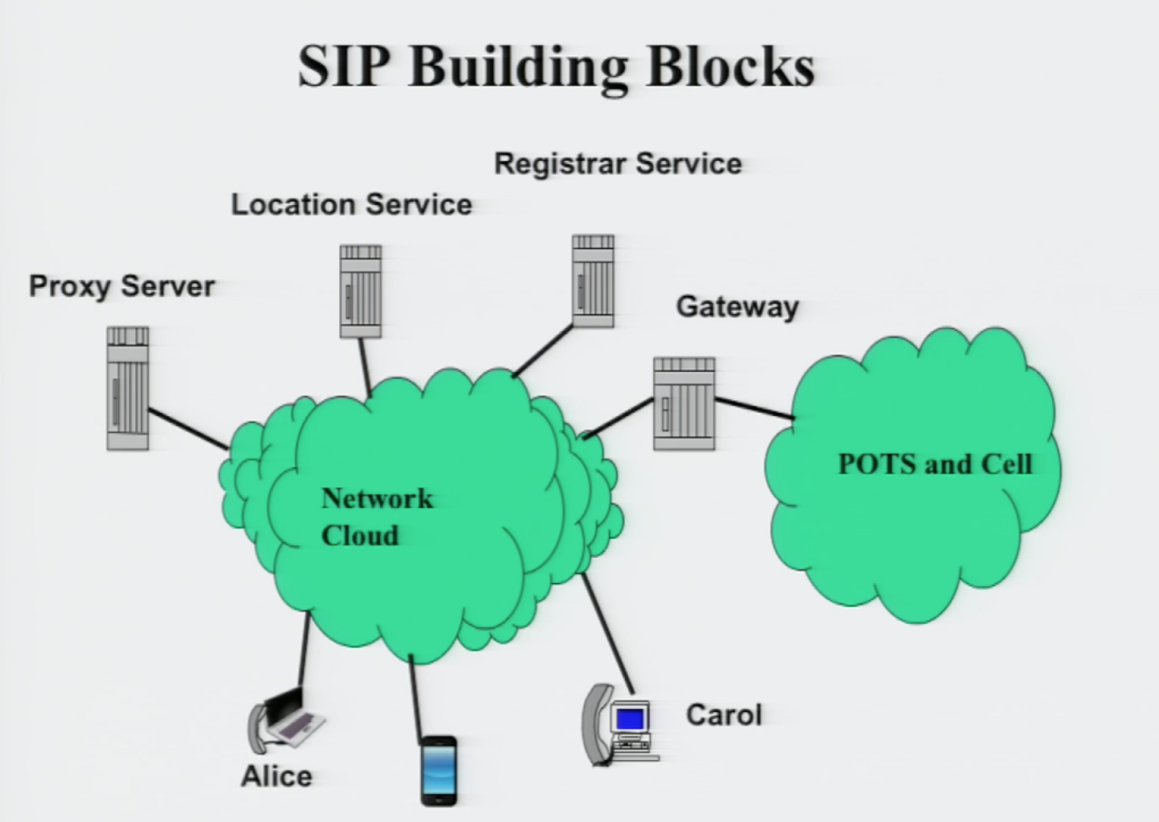
- Certificate Authorities/DNS
- SIP
- SDN (Software Defined Networks)
- NFV (Network Function Virtualization)
- **Remember the Five Important Elements of Security**
- Digital Signatures came up again (provide authentication & integrity)
- CA certificate is basically a digitally signed doc reflecting a website's identity and public key
- Non-trivial example of DNS hijacking (Lecture 13 8:30)
  - "YOURRBANK.COM"
- Voice and Video over IP (VoIP)
  - RTP Packet flow for a Video or Voice call

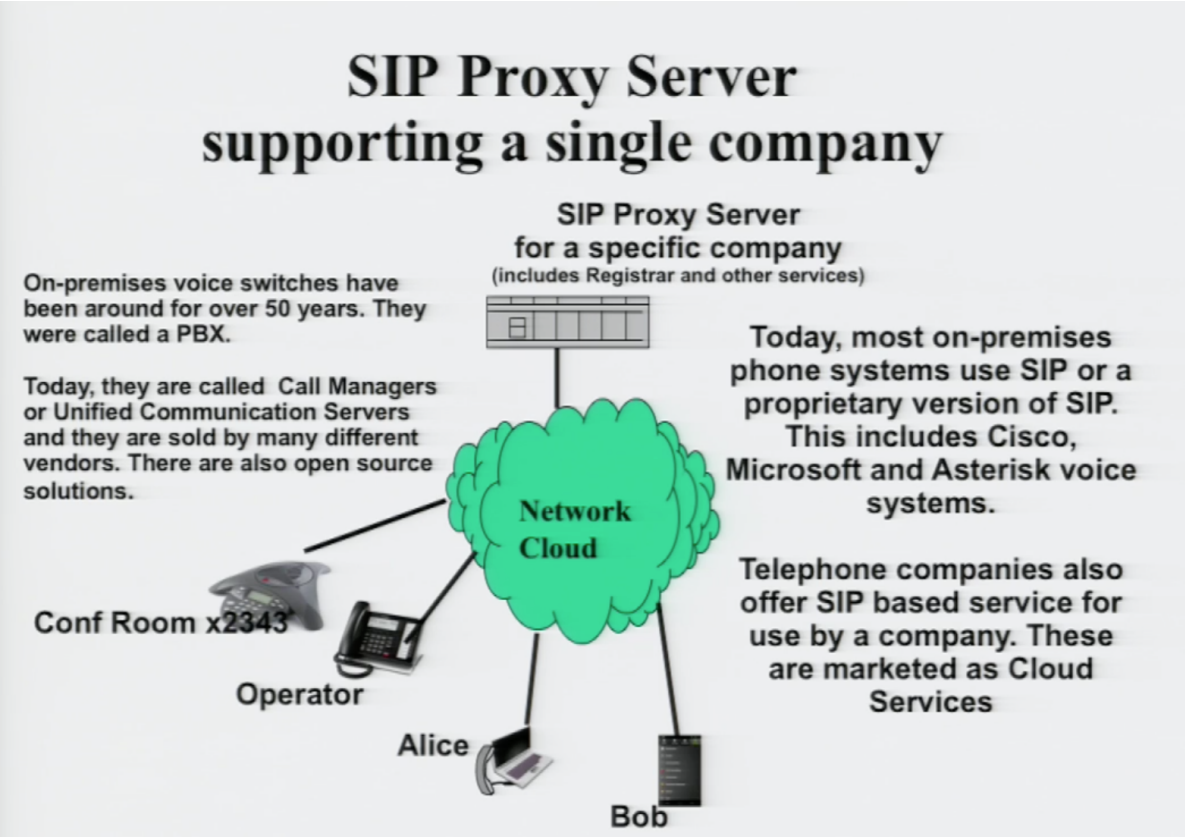
- Voice packets are distinct from video packets
- There are distinct "media streams"
  - 

  - When you hear RTP (Real-Time Protocol) think: two-way interactive communication
  - This is different than video streaming!
    - Video streams (and separate Voice streams) are generally sent over TCP
  - The receiving system know how to distinguish between video and voice packets at the transport layer by the port number
  - These port numbers are assigned during the capabilities exchange

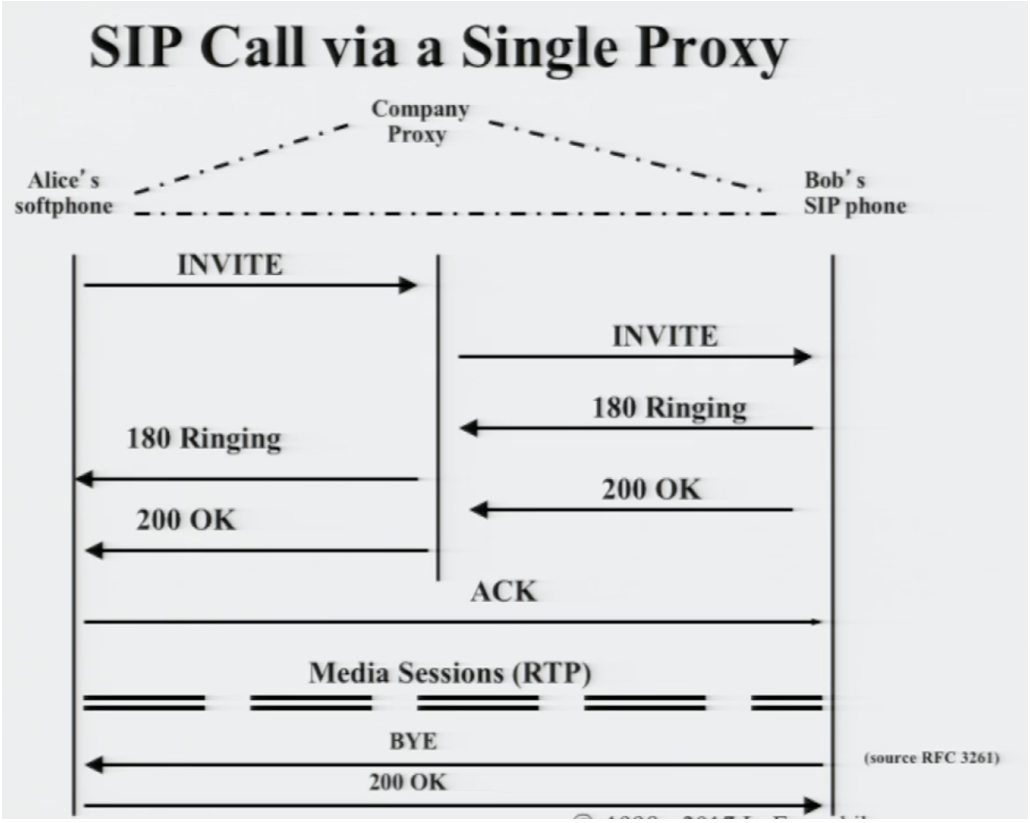
- SIP:
  - Establishes a session
  - A session is generally a "call" but could be Instant messaging or something else similar
  - **Remember again the three main concepts provided by SIP**:
    - Signaling (Call Setup)
    - Capabilities
    - Media Exchange
  - Appreciate the SIP Protocol architecture allowing for UDP and TCP transport:

- 

  - Real-Time applications (skype, Facetime, etc.) will use UDP always
  - SIP Port numbers: 5060 or 5061
    - IANA assigns these!
  - Think of SIP as the suite or family name Because there are a multitude of protocols underneath it to get it where it is today
    - Things got so confusing that RFC 5411 was written ("A hitchhiker's guide to the session initiation protocol")
    - 25 pages of mentions of RFCs that are necessary to implement SIP properly
  - SIP building blocks
- 

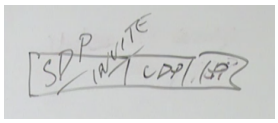
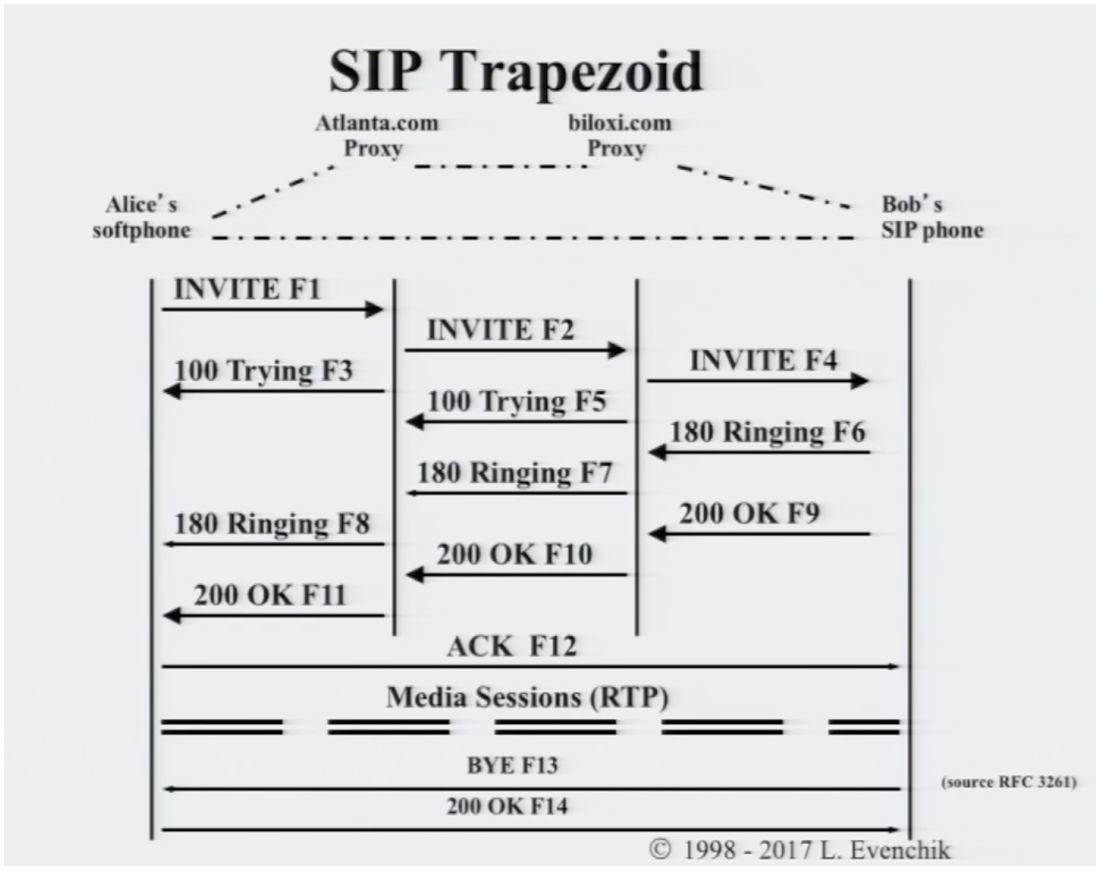


- User Agents
- Proxies
  - Have the Registrar and other related services
  - Also called Call Managers or Unified Communication Servers nowadays
  - Handles all of the call processing
  - Has a very specific functionality for SIP
- Asterisk is a popular open source solution
- SIP call through a single proxy
  - RFC 3261
  - Again, appreciate the three phases!



- Appreciate the similarities to EMAIL (HELO -> 250 vs. INVITE -> 180)
- Regarding the above diagram, its not immediately apparent where the capabilities exchange is happening.
  - Be aware that this could simply piggy back on the call setup portion (Within the same packet)

- Learning from the past: What prevents loops in IP routing networks?
  - TTL!
  - What if the SIP Proxy we're dealing with is buggy and sends the SIP INVITE to another SIP Proxy and another ..... etc.?
  - That SIP Proxy is UDP and the message is going from server to server
  - The TTL at the IP layer is never going to catch that its in a bad situation
  - So there is a Max-forwards in the SIP INVITE message to account for this

        - When designing protocols always try to account form something that could go wrong
    - SIP Reply codes look very similar to Email HTTP etc. Good ideas from the past prevail!
    - The capabilities are included within the SIP INVITE message
    - Following the SIP INVITE is the SDP (Session Description Protocol) which includes the capabilities exchange information
    - 
    - SIP Trapezoid:
        - When multiple SIP Proxies are involved:
        - 

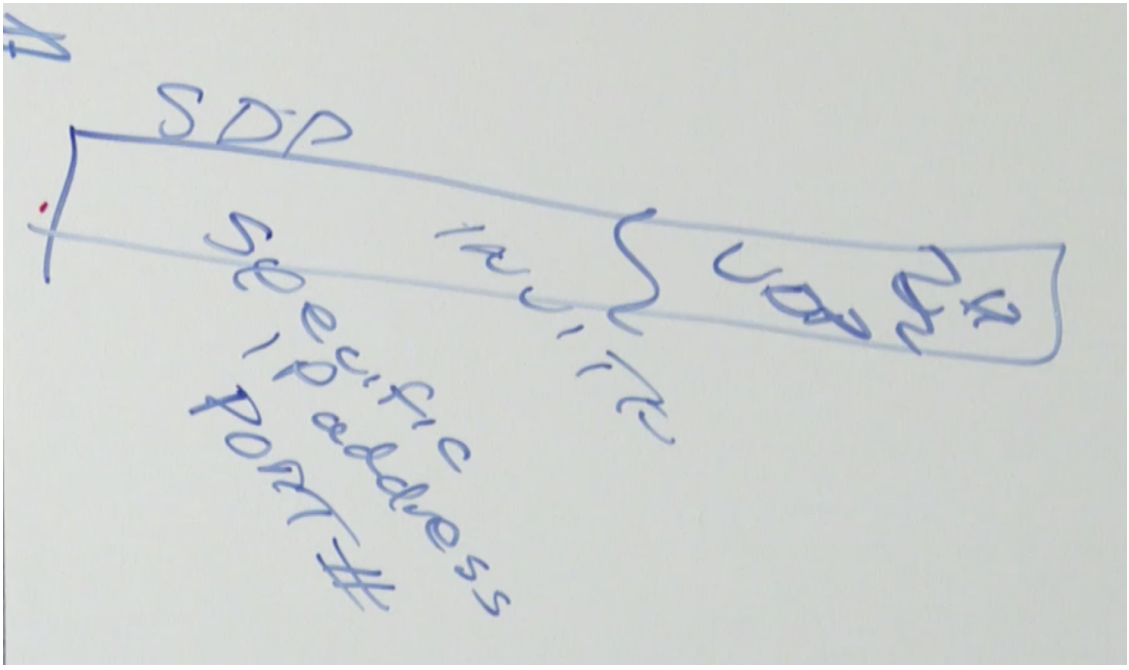
        - Again, this is super similar to MTA forwarding!!!
    - **How is SIP similar to EMAIL?**
    - Note that a **HUGE** difference in this scenario is that the media exchange is done end-to-end and not through the SIP proxies!
        - Real time applications are performance wary and shouldn't go through these extra hops
        - This is a big **DIFFERENCE** from EMAIL/MTA forwarding
    - Important to keep track of the path that the INVITE messages take through the network
        - Via fields: just breadcrumbs to show how packets flow from one place to the other
    - DNS & SIP how do they fit together?
        - Think about EMAIL again!
        - An MX record is to Email as an SRV or NAPTR (Name authority pointer) is to SIP!
    - `dig sip.mit.edu SRV`

- How you tell an inbound SIP call from an inbound Email from an inbound Web?
    - The port information!

- SDP (Session Description Protocol):
    - A mechanism to allow for direction of media stream to different places

- Specifying the types of media streams
- Imagine a scenario where you want to send audio to the nice speakers in the back (one IP) and the video to the machine with the nice screen (another IP)
- Gives you the ability to specify the types of Codecs, the IP addresses to send specific media streams to, and the port numbers to which the streams should be delivered
- We could spend days on SDP but this is the most important thing to appreciate

◦ SIP and NAT and Firewalls (oh my!):

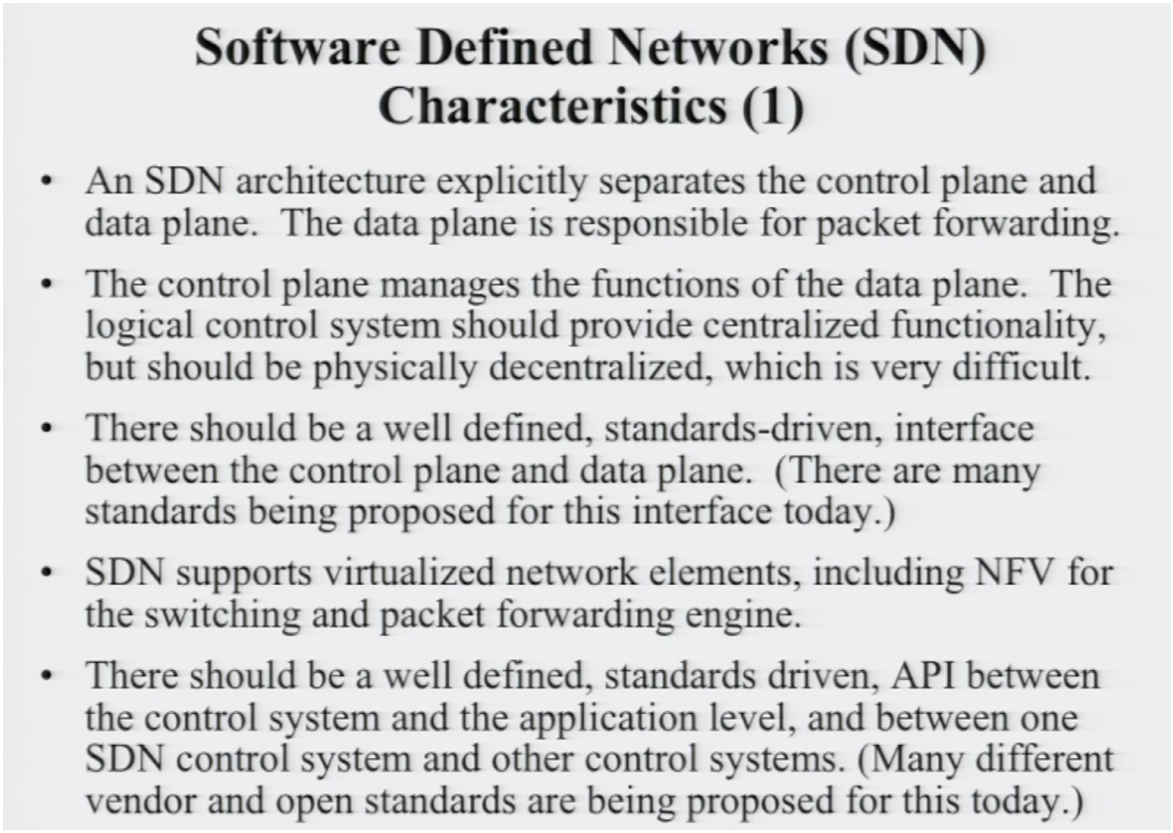
- Supporting VoIP through NAT/FW is not easy!
- The normal case for dealing with NAT forwarding info (address & port) is done at the UDP/IP levels
- In the case of SIP there is also information that needs to be mapped and forwarded (dest IP and port) and this information is set by SDP which lives inside the UDP payload
- 

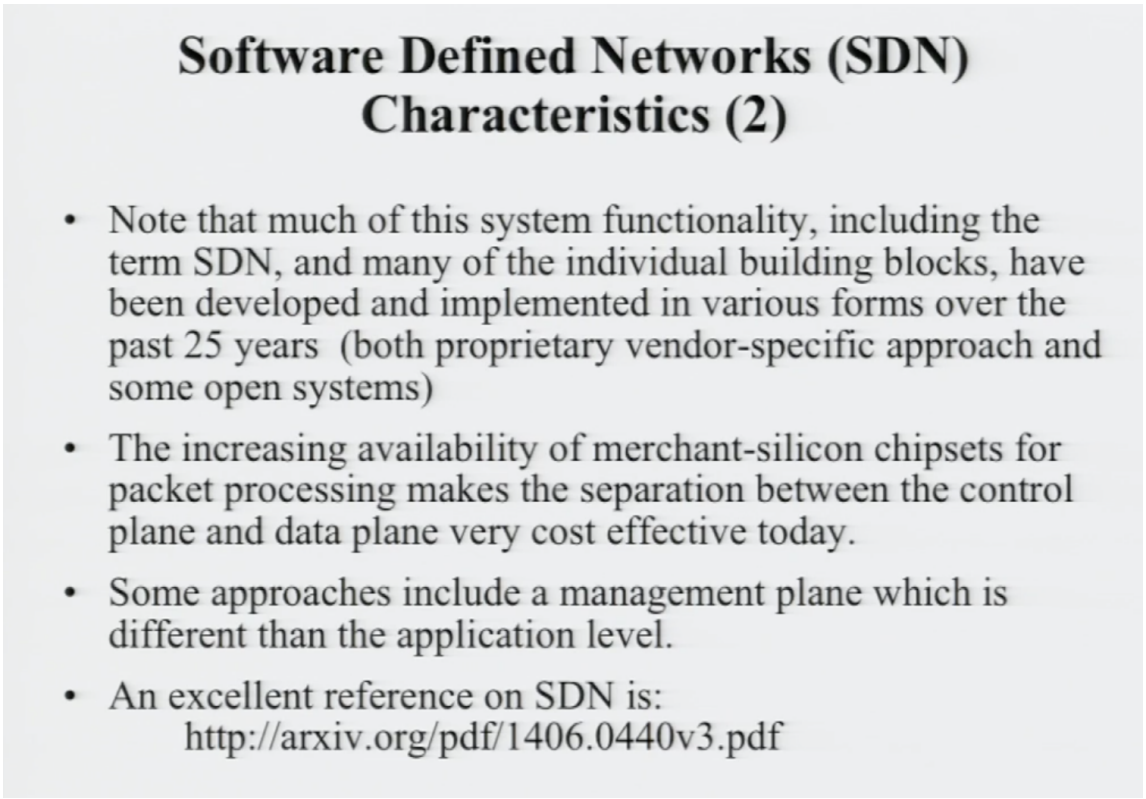
- SIP uses STUN, TURN, and ICE!
- **SIP Use of STUN, TURN and ICE**
  - SIP clients and proxies use STUN, TURN and ICE to overcome the problems caused by NAT. Proprietary protocols are also used by many systems.
  - STUN – protocol used by a client to determine the presence and type of NAT
  - TURN – protocol for working with a media relay located on the Internet. A TURN relay replaces the need for an inbound call through at NAT. All of the clients place outbound calls to the TURN server instead.
  - ICE – complex protocol for managing NAT traversal in protocols such as SIP (for VoIP) that use the offer/answer model.

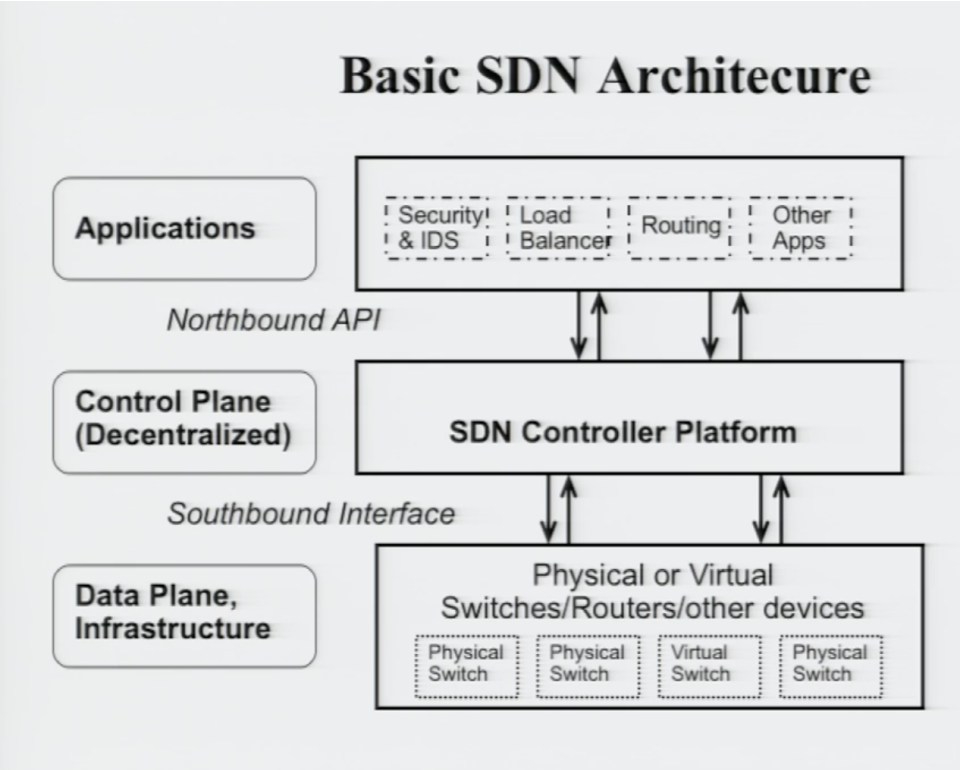
◦ RTCWEB/WEBRTC

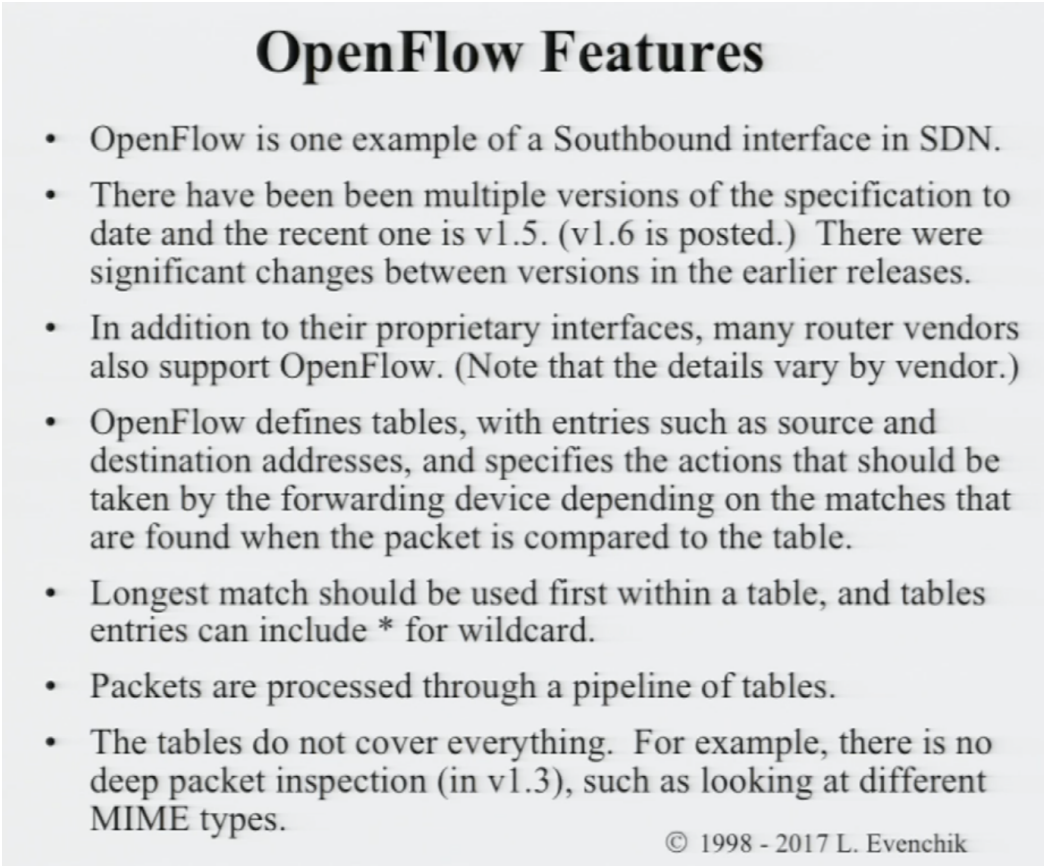
- WebRTC is W3C work, RTCWeb is IETF work
- The idea is that: "Lets allow the web to directly use VoIP"
- No need to install 3rd party plugins or software
- functionality for signaling (call setup), capabilities exchange, and media transfer is done within the browser
  - HTML5 Code to start a call "one click"
- But then there were Firewalls, and different codecs etc.
- This complexity continued...

- SRTP for media exchange
- ICE/TURN/STUN for NAT traversal
- "Click to talk customer service" is a proposed use case

- SDN (Software defined networking):
  - Separates the control plane and the data plane **For routers**
  - Data plane: Handling of packets
  - Control plane: OSPF or RIPv2 (manages the forwarding)
  - 

  - 

- This idea is really nice because "above the line" in the control plane the problem is really tuned for a software based solution and "below" the line in the data plane its really a speed and hardware solution, but there still needs to be an interface (software) for the two to communicate
- 

- Homework question on OpenFlow:
- 

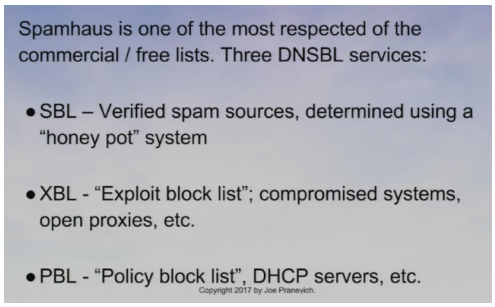
- See Lecture 13 1:36:00 for more details on OpenFlow
- OpenFlow can "look at everything together" where everything is all of the information in a usual frame (Ethernet, IP, TCP/UDP info)

- NFV (Network Function Virtualization)
  - In the old days one had to buy specialized hardware to move packets around
  - Nowadays general hardware is fast enough!
  - Specific software can implement networking functionality on non-specialized hardware
  - Switching, routing, firewalls etc.
- Models and standards groups:
  - IETF, Open Network Foundation etc.

Spam and Email Filtering:

- Spam is the single greatest challenge to email providers and users
- Without constant anti-spam efforts Email would be useless
- At one point 98% of email was spam, and the infrastructure to support anti-spam efforts cost more than the infra. to support the email itself
- Spam has decreased significantly in recent years
- 60% of email is currently spam
- Tragedy of the commons
  - "If everybody brings two goats"
  - "If everyone became a spammer"
- Spam is unsolicited and in bulk
- 1966 first networked email
- 1978 first spam mail was sent
- 

- #s are huge with spam (10s of billions messages)
- Where does spam come from?
  - Commercial email senders
  - Abused ISPs
  - Botnets (compromised systems)
- Botnets are the big evil
- Should ISPs block outbound port 25? what about the internet's end to end principle?
- Costs of Spam:
  - Network Bandwidth

    - Disk Space
    - CPU Cycles
    - Lost Time
    - Almost $22 billion lost annually to spam (in 2004)
- No silver bullet to preventing spam
- Adopt a layered approach
- First lines of defense are key
- The "deeper" into your network spam gets the worse off you'll be
- Where should you block spam?
    - Everywhere!
    - At the sender
    - At the network (block known spammers from communicating at all)
        - ACLs
            - Traditional Block lists
        - Real-time block lists (DNSBLs)
            - Block lists distributed through the DNS protocol
            - Cooperative approach
            - List adapts in real time based on who is sending spam "now"
            - 

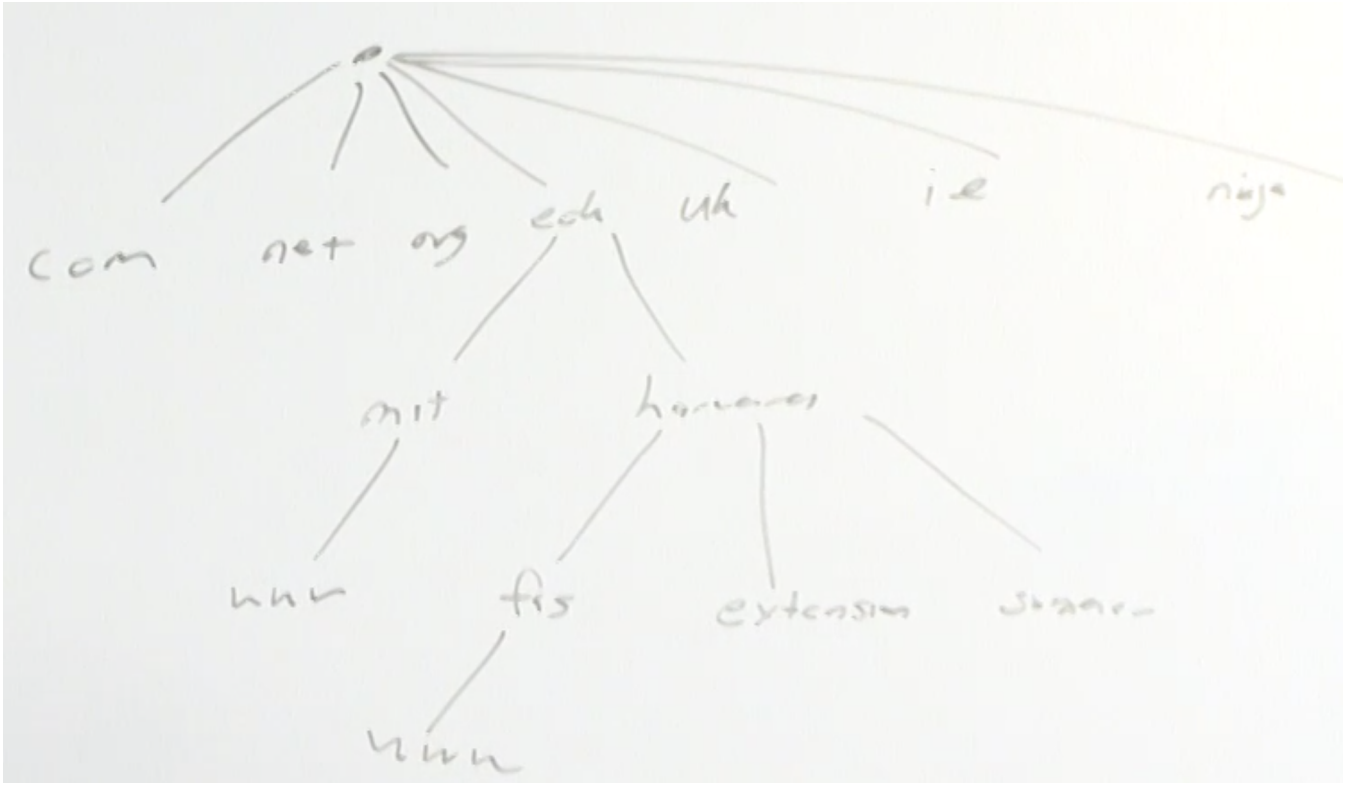
            - Honey pots: Place a never used email address on a webpage. Any email that arrives would most likely be spam
            - Blocking things before you ever see the mail so it could be risky
        - Greylisting
            - Not a black or a white list but something in-between
            - Send back SMTP 450 (try again later)
            - "Spammers do not queue" it would be very unlikely for a spammer to resend their messages
            - SPF (Sender Policy Framework) (38:00)
        - DomainKeys (DKIM)
            - Emails are signed with the private key of the sending domains
    - At the MTA
        - Rules-based systems
            - Identify spam based on content and headers
            - URL in email blocking
        - Bayesian filters
        - Big companies running neural networks and crowd-sourcing
    - At the User Agent
        - CAPTCHA
        - Outbound filtering
    - Future of email?

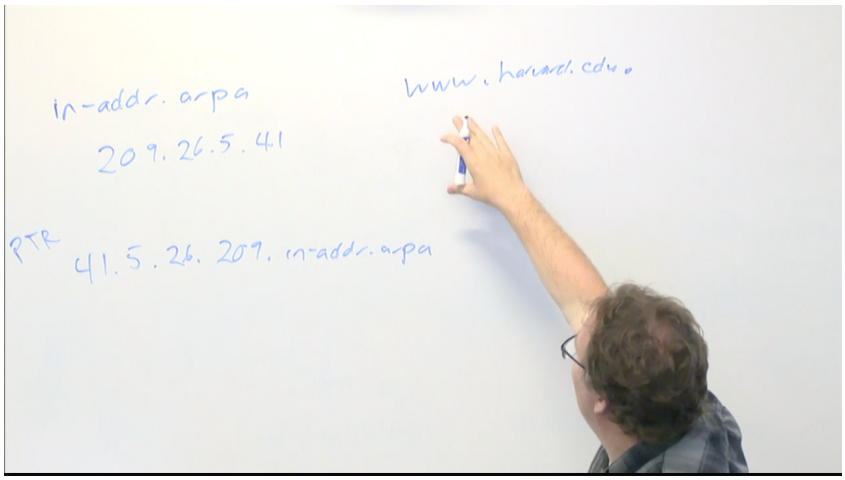
Section 5:

Key Topics:

- DNS

    - Hierarchy
    - Record Types
    - Reverse Lookups
    - TLDs
    - Authoritative Name servers vs. Caching
    - Recursive Lookups
    - Root Servers
- Email Basics
    - SMTP
    - MIME Types
    - Email Flow (through User Agents/MTAs)
- Other Protocols
    - TELNET
    - FTP
    - HTTP
- Security
    - Types of security
    - Firewalls: Different Types/Granularity
    - Symmetric vs. Asymmetric cryptography
    - Hashing
    - Digital signatures
    - Website security
    - VPNs
        - Tunnel & Transport VPNs

- DNS
    - The largest distributed database system ever created (as an aggregate)
    - DNS hierarchy/Tree
    - Root -> TLDs -> Specific Zones -> etc.
        - 

    - One domain name can have many record types:
        - A, AAAA, MX, NS, etc.
    - CNAME is like a symlink
    - Multiple A records can be used for load sharing
    - TLDs:
        - gTLDS & ccTLD
            - .com & .us
            - New TLDs today: .ninja etc.
    - Name servers are queried from root to bottom
        - Caching Name server is queried initially with a recursive DNS query from local machine (this is the NS configured for your local machine to use)
        - Caching NS then makes non-recursive queries to resolve
            - asks root: do you know about www.harvard.edu? root says: "no but I know another NS you can query about .edu domains"
            - asks .edu NS and rinse & repeat
        - Caching name servers know nothing initially but are willing to lookup anything
        - Authoritative Name servers know about their specific friends
        - Reverse DNS:
        - 

- TELNET
    - Older machine administration protocol
    - Connect to a port and give human readable commands to interrogate a machine
- SMTP:
    - Think of it as a suite of mail based protocols
    - MTAs are responsible for sending mail to its intended destination (very possible to send along to another MTA)
    - SMTP port 25 between MTAs
    - Appreciate that the connection to the MTA can vary: HTTPS, SMTP etc.
    - Email is not reliable (could be caught by spam filter, no proof of delivery, improper recipient) even though all of the protocols it uses are (TCP, IP, SMTP, HTTPS)
    - Parts of Email:
        - Envelope: Used by MTAs
        - Headers: Used by User agents (To: FROM: etc.)
        - Body: (Actual message)
    - MIME: Defines a content encoding and file type
        - text/plain, text/html, image/jpeg
- Structured way to think about security

    - Privacy and confidentiality
    - Authentication
    - Authorization
    - Integrity
    - Nonrepudiation
    - Security should happen at every layer
    - Humans are always the weakest link

- Cryptography:

    - Caesar Cyphers (substitution cypher)
    - Symmetric
        - One key can go "back and forth" between encrypted version and original text
    - Asymmetric
        - Need both keys to be able to be able to go "back and forth"
    - Hashing provides integrity
        - One-directional cryptographical method of giving a "fingerprint" of data
    - Appreciate one-time-use session keys, and be able to talk about them
    - A digital signature is hashing plus Asymmetric cryptography

- VPN Tunneling:

    - Tunnel: router to router VPN
    - Transport: Host to router VPN
    - TUNNELING:
        - Olden days we used to lease a physical line between two places, but that became expensive
        - Nowadays we use VPN routers at two locations
        - Encrypted point to point connection
        - Uses IPSEC or something similar to be able to send data over the internet to the next VPN and have it remain encrypted
        - Host machine does not know its on a VPN
    - TRANSPORT:
        - host to routers
        - Host machine uses client software to connect to a VPN server
        - Another NIC will show up in host OS
        - Host knows that its on a VPN

- Website Security:

    - TLS

Lecture 14:

- SIP:

    - Three phases
    - SIP initiation can run over UDP & TCP
    - The capabilities phase can be piggy-backed on the session initiation
    - Really appreciate the SIP trapezoid and how all of the various protocol elements are working together

- SDN:

    - Network function virtualization
        - A router or networking component has been implemented through software on some generic hardware

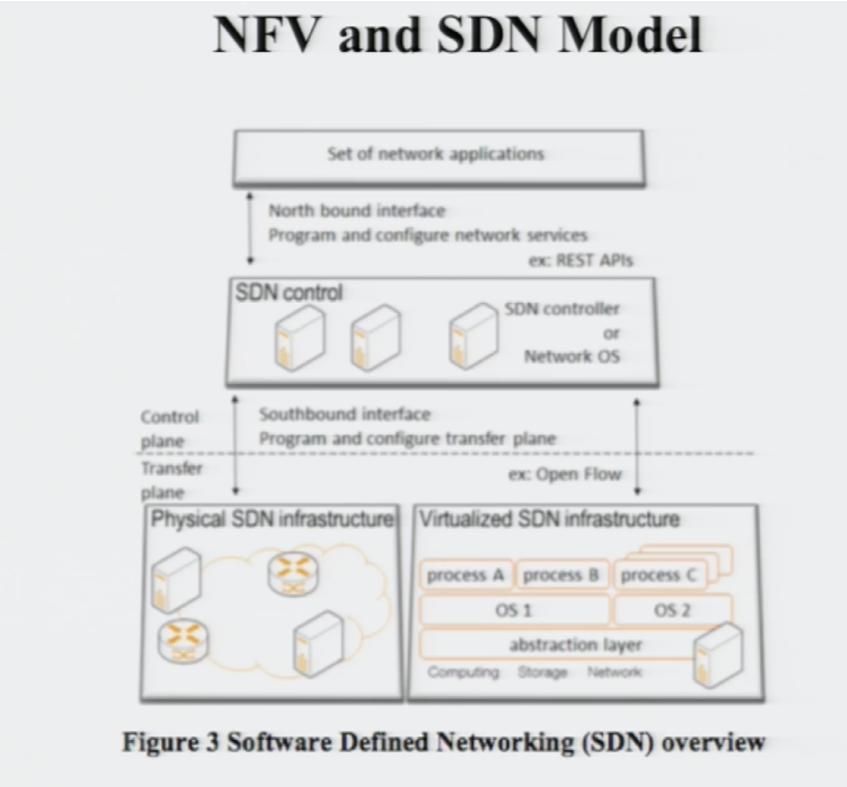
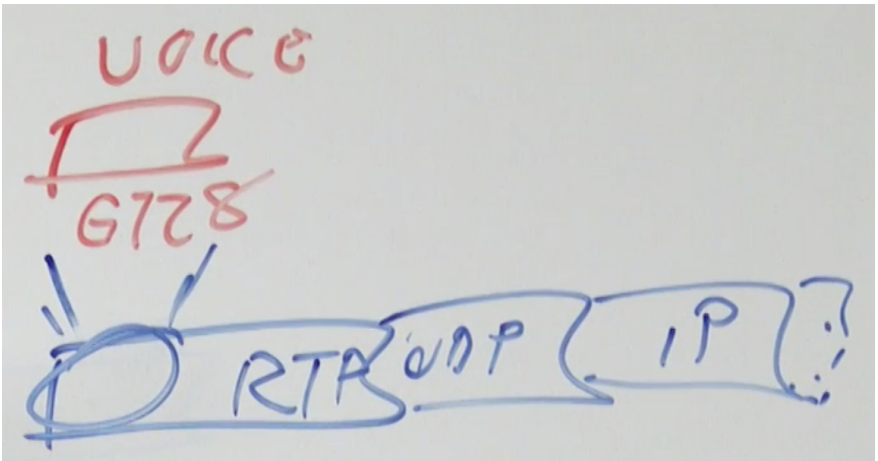


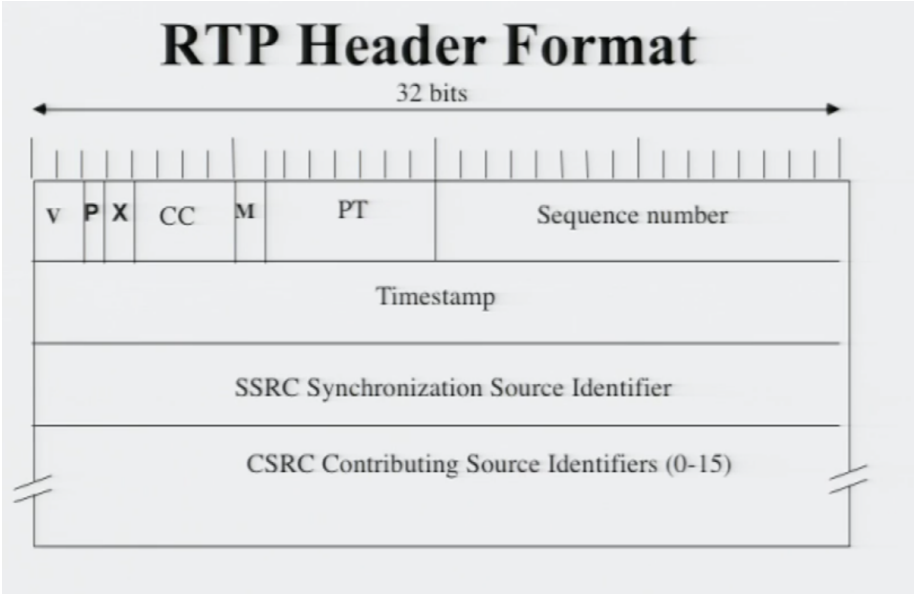
Figure 3 Software Defined Networking (SDN) overview

- Separation of control and data planes
  - Has been done for a number of years in proprietary ways by different vendors
  - Could do some packet processing on hardware specifically meant to do so
  - This is now done in a more general way through the use of software
- Network Virtualization and NFV are not the same!!!
  - Think VLANs & VPNs
- OpenFlow

- Voice and Video over IP:
  - Taking analog voice or video and packetizing it
  - Delay should be 100ms or less
  - RTP (Real Time Transport Protocol):
    - Uses UDP
    - Example of Voice Data with G728 Codec encapsulated within a UDP packet



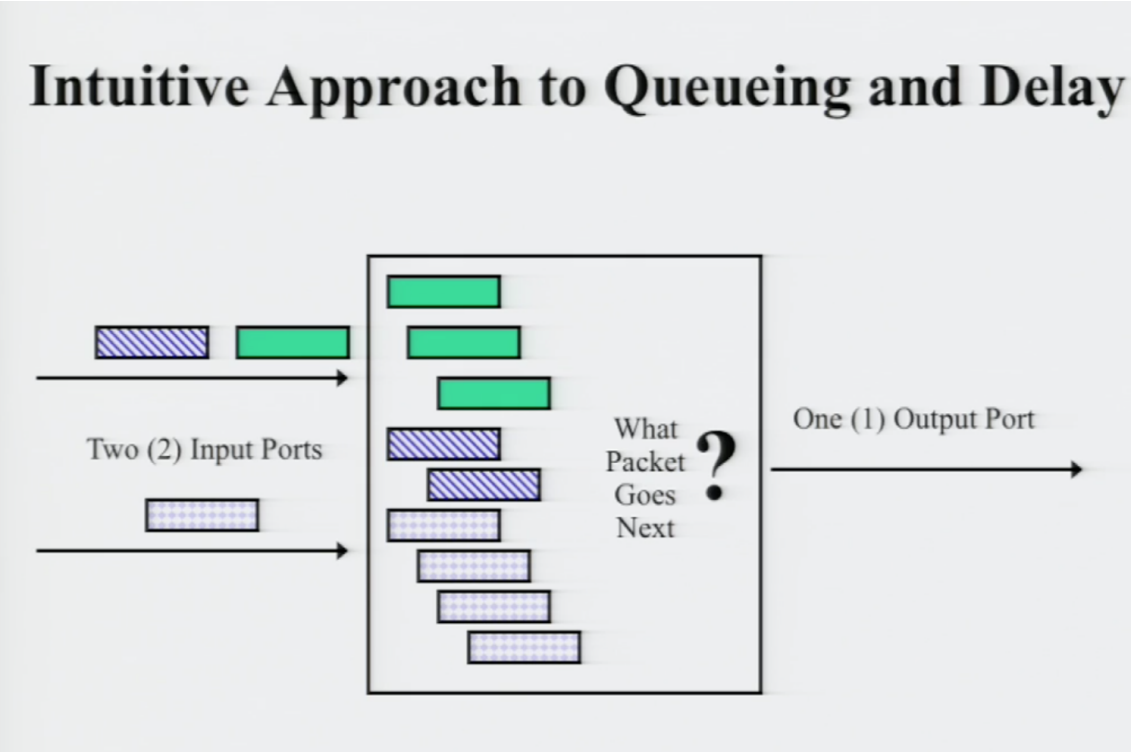
    - Because we don't want the delays that come with a reliable protocol, we use UDP.
    - This is troublesome because UDP doesn't provide sequence # for ordered delivery

- RTP's functionality (SP3) provides sequence #'s so that the receiving application can reassemble that data in the proper order
- RTP Header Format:
  - 

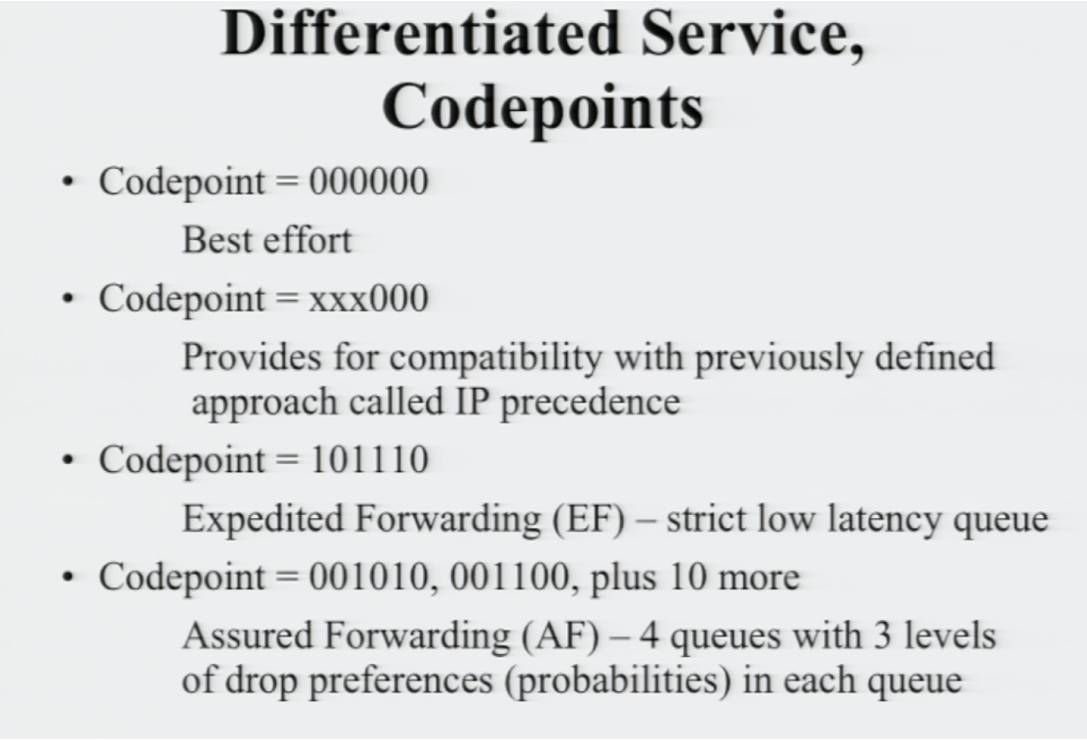
- Voice and video packets will have separate sequence numbers
- Port number tells the receiver whether its voice or video and port number is set during the capabilities phase
- Another issue arises here (think the "DUBBING"/"LipSync" problem)
- RTP Header has a notion of a timestamp in the header for both audio and video packets
- The receiver can then say that: "this audio and video go together, I'll play them at the same time"
- Codecs could change during the duration of a stream, but it would take a very sophisticated system to do this
- In the case of more complex applications (VR, multi-channel audio) each media stream will have its own port number and RTP sequence numbers
- Jitter: Variability in delay
  - There should be less than 40ms of Jitter for good quality streams
- There should also be less than 0.1% Error Rate (meaning that packets just aren't showing up)
- The aforementioned metrics are QoS (Quality of Service) parameters

◦ QoS (Quality of Service):

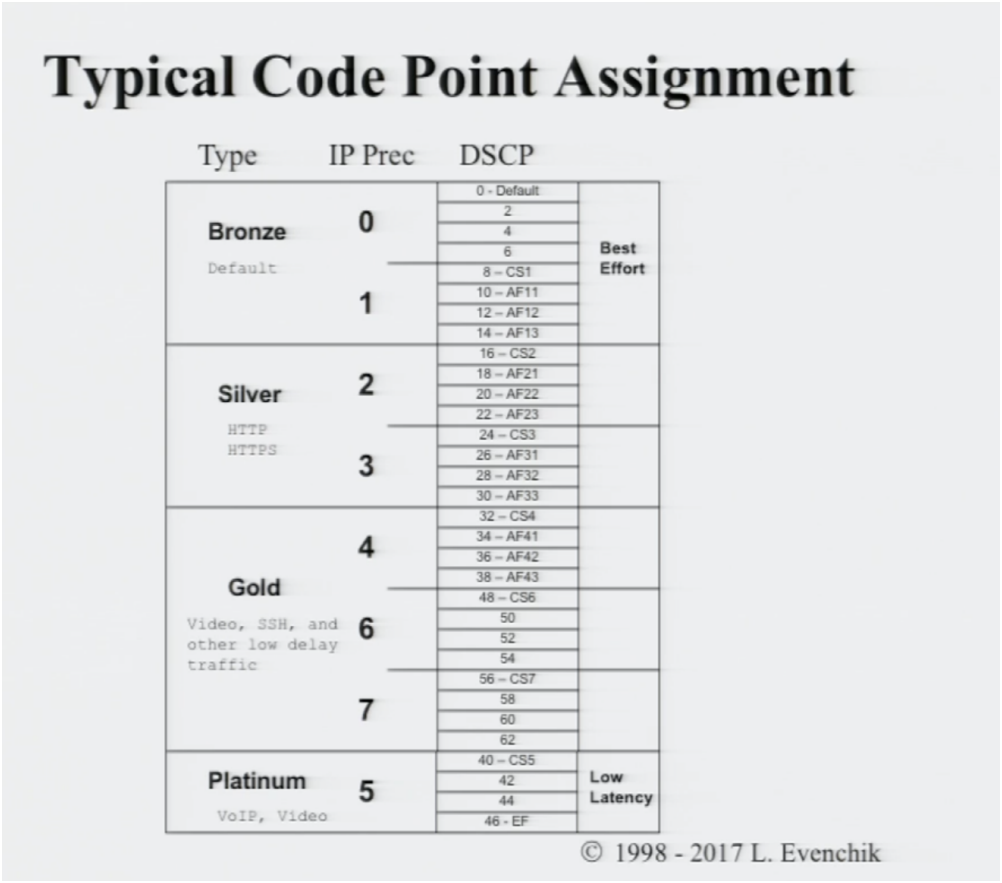
- The requirements focus on:
  - Bandwidth
  - Error Rate
  - Jitter
  - Delay
- QoS is not a new issue
- If you have a lot of bandwidth available you don't have to worry about QoS (as much)
- Thinking back to IP which included QoS as a part of its header format, we can imagine specific types of traffic being given a higher priority than others



- When a router is receiving packets and implementing a smarter approach to sending them out the other side:
  - The Sender identifies the type of packet (what type of service?)
    - Think about sending USPS mail (Priority vs. regular)
  - All devices in the network treat high QoS packets first
- Network delay in the real world means that the QoS metric is extremely important
  - The most significant delay component nowadays is the queueing of a packet in a given networking device while it is waiting to reach the front of a queue
  - Only one packet can be sent at a time and all other packets must wait for their time to be sent
  - Delay becomes cumulative over the course of hops taken throughout a network
- Policing of QoS traffic can be done with a Firewall or something similar so that a bad actor doesn't exploit the QoS functionally to their advantage
  - QoS bits in IP packets could be set to 0 upon crossing the firewall boundary
- If any devices in the network don't implement QoS, then they will probably use a general means of choosing packets which come into the queue and QoS benefits are lost at those hops
- Approaches to managing QoS:
  - Have a ton of available bandwidth
  - Allocate bandwidth to particular users or conversations and manage that bandwidth allocation as they change (this is a very outdated and unused approach today)
  - DiffServ (Differentiated Services) Marking of packets in a specific manner so they get treated differently over a network
- DiffServ "packet marking" ended up being implemented within some bits that were available in the IP header
  - Specifically, in the TOS (Type of Service) portion of the IP Header (6 bits)



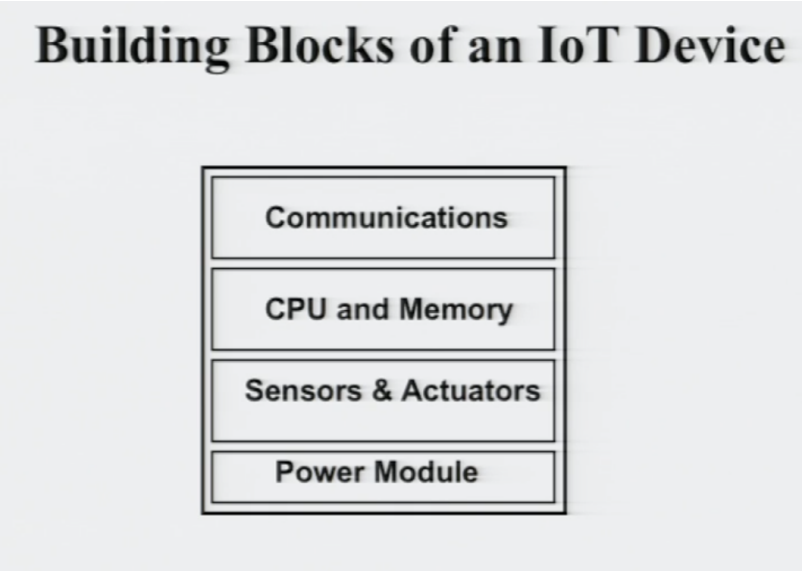
- Typical assignment for DiffServ:

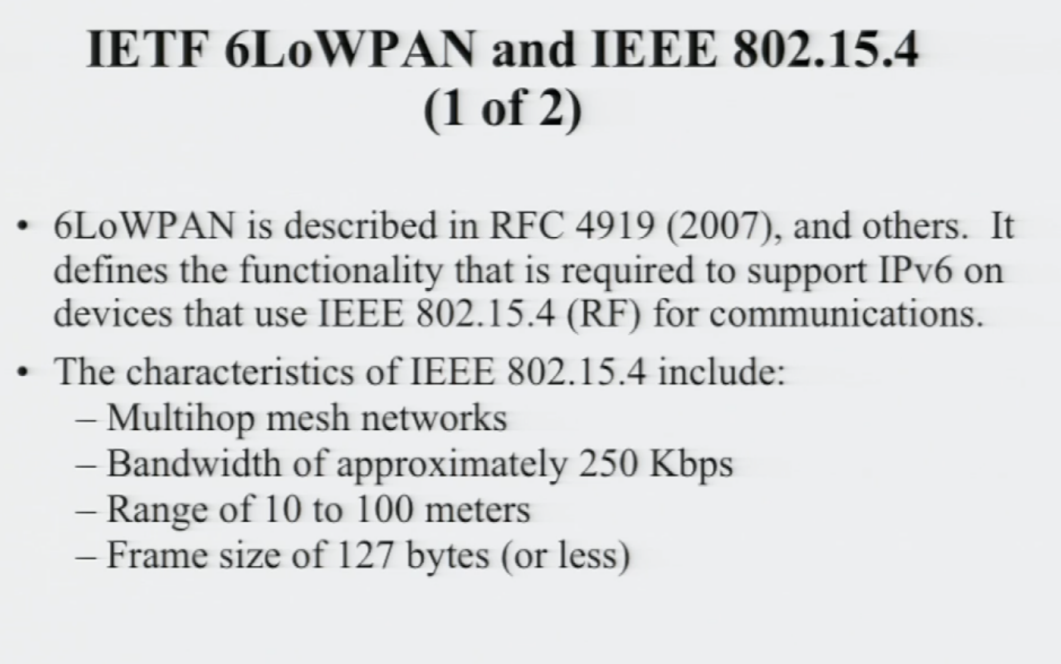


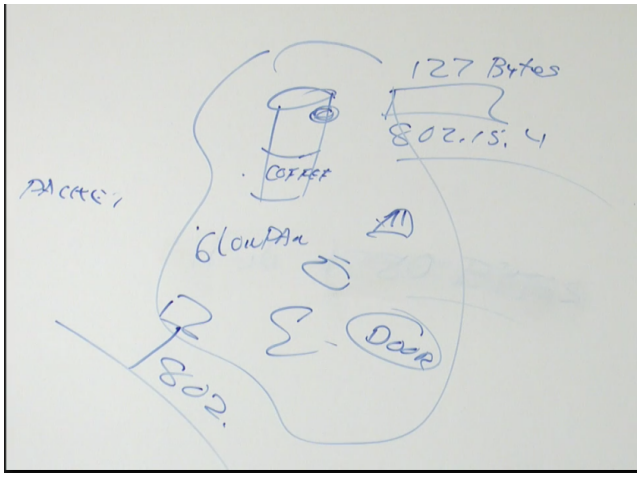
- All of the network service carriers have different price points for traffic that should be treated better than other traffic
- The general internet does not implement QoS
- DSCP to UP mapping example for 802.11 (recent example 2017)
- QoS really works well when implemented properly

◦ IoT:

- "If you can come up with a device that shouldn't be connected to the internet you haven’t been creative enough"

- Approx. 20 billion IoT devices by 2020
- If 20 billion, then its really time to use IPv6
- 

- IETF 6LoWPAN and IEEE 802.15.4
- 

- Important to appreciate the adaption layer provided to these networks to allow for use of IPv6:
  - Small packet size of 127 bytes didn't meet the need of the original IPv6 spec (1280-byte minimum)
  - 

- Concern around IoT devices not being able to be updated with security fixes

        - DDoS Botnets against DNS servers

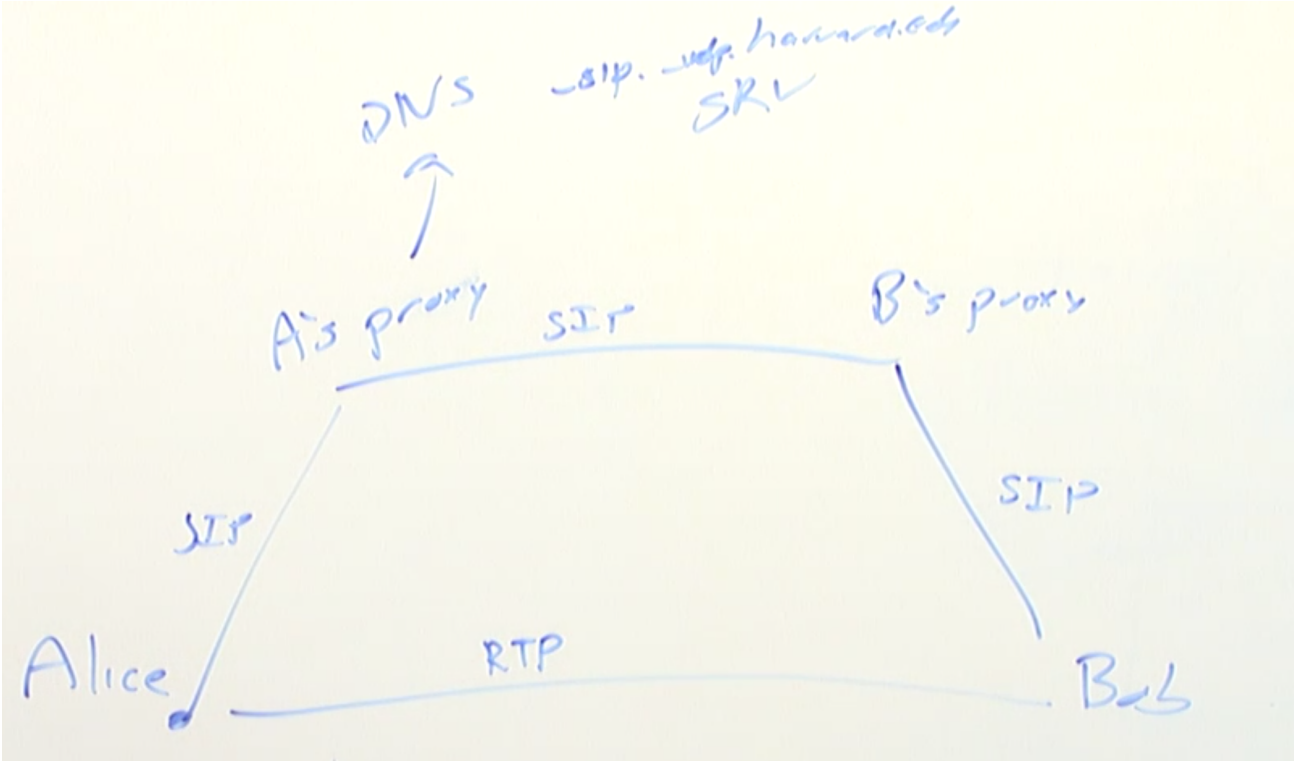
- Important things from this lecture:
    - 6LoWPAN and the adaptations around it to be able to work with IPv6
        - Low power, low cost devices with small MTU networks
    - QoS and the configurable
        - what parameters need to be managed with QoS

- Course Summary:
    - Go back to midterm review
    - IP and Ethernet Headers
    - Important info in TCP/UDP headers
    - What are some of the great ideas in networking?
        - Network and protocol layering/encapsulation
        - Multiplexing
            - Physical vs. Logical multiplexing
        - Addressing:
            - Local significance vs. global significance
        - Circuit vs. packet switching
            - Smart vs. dumb networks
        - Understanding the end-to-end approach vs. "middle boxes"
        - Applying the SP3 framework to learn and understand protocols
        - 5-tuple, 7-tuple, N-tuple, time sequence diagrams, and other simple tools
        - Data plane vs the control plane
            - signaling vs data transport

Section 6:

- Key Topics:
    - RTP/RTCP
    - SIP/SIP Trapezoid
    - VoIP
    - QoS
    - SDN
    - IoT
    - SDN
    - Email and Spam

- VoIP/SIP
    - Is a big suite/collection of protocols
    - Companies aren't implementing SIP as we see in class, but are using smaller pieces of it
        - Facetime can't talk to hangouts
    - SDP is important to know
    - SIP uses RTP underneath the hood
        - RTP is an unreliable, sequenced protocol
            - Use of codecs and content encoding for specific types of media transfers
            - G series is audio, H series is video
        - RTCP is a status protocol. "How your transmission is doing", Jitter metrics
        - SIP Proxies use DNS lookups to find the SRV record

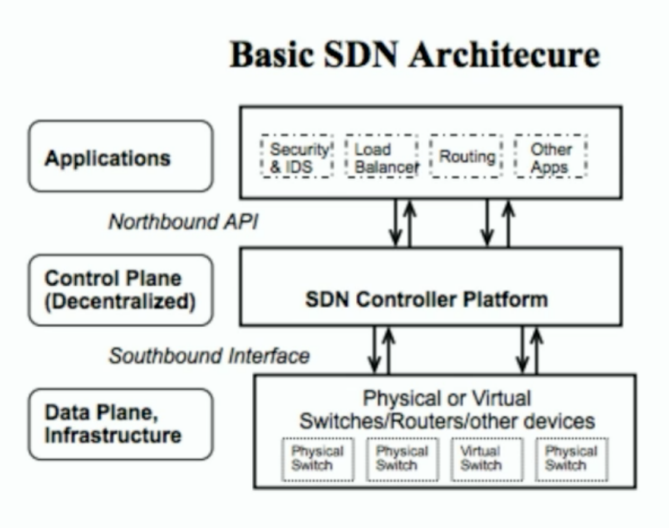
        - A given proxy needs to know where a registered user is at a given time
            - Many devices at many different locations
        - Trapezoid!
            - Top 3 sides of the trapezoid are SIP while the bottom (media exchange) is done with RTP
        - 

        - Uses SRV DNS records (_sip._udp.*) to determine where a connection should be established __
        - Uses the SDP protocol
        - Why is TCP a poor choice for audio and video real time transmission?
        - RTCP: Tells you how your RTP is doing (report on jitter and delay and packet loss)
        - RTP Header:
            - Sequence number is important
            - Voice or video data over UDP
            - Timestamp for sync across streams
        - NAT traversal makes SIP tough

- Email and Spam:
    - DNSBLs
    - Greylisting
    - Rules-based systems
    - CAN-SPAM and opt outs
    - Spam is unsolicited and bulk
    - Anti spam techniques:
        - ACLs
        - DNSBLs
        - Greylisting ("Spammers almost never retry")
        - Rules based filters

- QoS:
    - IP type of service field in header inspected and prioritizes traffic based on that
    - Generally only useful within the scope of a specifi domain or organization

- SDNs (Software Defined Networks):

    - "Virtualization for networks"
    - Network Virtualization
        - Separates the logical from the physical (think VLANs)
    - Network Function Virtualization
        - Networking function in software on generic hardware and not an a typical network appliance
    - SDN Planes:
        - Application Plane
        - Control Plane
        - Data Plane
        - 

- IOT
    - "Everything that could be conected, should be connected"
    - 6LoWPAN:
        - IPv6 implementation for IoT devices
        - Low bandwith low power
        - Mesh networking

Final Exam Review:

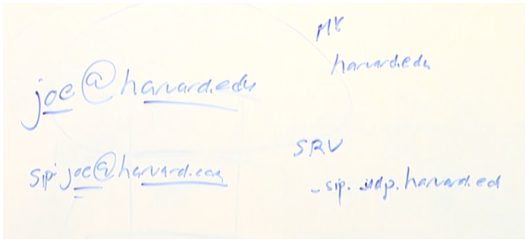
- Harvard Hall Room 102 6:30PM Aug 7th
- Review the midterm review!
- Drawing of:
    - Network packets
    - Time sequence diagrams
    - Switch configuration
- DNS:
    - Understand the hierarchy:
    - Different kinds of records and what they do (A, AAAA, CNAME, MX, SRV, PTR)
    - Reverse DNS
    - TLDs
    - Authoritive vs caching
    - Recursive lookups:
        - This is done to your local name server and the NS makes non-recursive lookups down the hierarchy

    - Root Servers
- SMTP:
    - Think of it as a suite of protocols
        - POP, IMAP, HTTP, HTTPS
    - Envelope vs. Mail Headers vs. Body
    - MIME:
        - Taking the concept of old-school text only mail and extending it to work with specific types of media
    - Telnet:
        - A text-based administration protocol
        - Fundemental basis for these other text mode protocols used today:
            - Can send mostly human readable text based commands
            - SMTP
            - FTP
            - HTTP
            - SIP
        - UDP/TCP/IP do not fall into this category (Humans can't really type out the 32 bit packet information by hand)
- Types of Security:
    - Privacy & confidentiality
    - Authentication
    - Authorization
    - Integrity
    - Nonrepudiation
- Symmetric vs Asymmetric Cryptography:
    - Symmetric cryptography is a simple single shared key (fast encryption of data)
    - Asymmetric cryptography is a shared public/private key pair system
        - Keys work in tandem
        - One can't derive one key from the other
- Hashing:
    - One way mathematical equation
    - Yields a fingerprint
    - A tiny change in message yeilds more than 50% change in fingerprint (message digest)
- Digital Signatures:
    - Combination of hashing and symmetric cryptography
    - Usually utilizes a "Chain of trust" (Think trusted CAs)
- VPNs:
    - Tunnel vs. Transport
    - Tunnel mode is essentially router to router and end users don't know its happening
    - Transport mode is a client that is sitting on your laptop and the connection is initiated to the VPN Router
- VoIP:
- RTP the sequenced, unreliable protocol used to send the audio and video data
- RTCP reports on performance metrics of an RTP media transfer

- SIP:
  - SIP Trapezoid:
    - Connection establishment system that is used for SIP
  - Be able to speak about the underlying protocols
    - Audio 7 Video codecs
    - SDP
    - DNS
  - Similarities with SMTP
- QoS:
  - Jitter
  - Bandwidth
    - Is the easiest to deal with (just buy better/more resources)
    - Can just throw money at it
  - Error Rate
    - Wall between you and wireless router yields a higher error rate
    - Not a big deal nowadays
  - Delay
    - limited by the speed of light and the # of routers between you and your destination
- IoT:
  - 6LoWPAN
- SDN:
  - Application Plane
  - Control Plane
  - Data Planes
  - What do these planes do?
  - How do they work in a system such as OpenFlow?
- Spam:
  - DNSBLs
  - Greylisting
  - Rules-based systems
  - CAN-SPAM
- Back to Basics:
  - Bandwidth
    - Touched upon as a QoS metric
  - Multiplexing
    - RTP allows you to multiplex audio and video streams together at layer 5 in order to share the common SIP infrastructure
  - Switching
    - Think of DNS as switching done at layer 5
  - 5-Layer Model
    - Roughly defines everything we do, but its good to understand that there are a ton of gaps in this model
  - Encapsulation
    - MIME allows for encapsualtion of data at layer 5
  - Packets:
    - Are not what they used to be

        - We initially learned about packets as 32 or 16 bit wide things
        - We now know that packets are more "Telnet-able" (able to be constructed by a human on the commandline)
    - Time Sequence Diagrams:
        - SIP Trapezoid!!!!!
        - TCP handshake & retransmission
    - Flow Control
    - Error Control
        - RTP does error detction but not error correction
    - State transition Diagram for TCP **Lookup in reading!**
    - Core Protocols:
        - Ethernet
        - IPv4
        - TCP/UDP/RTP
        - DNS
        - Email & VoIP
        - What layers are each of these protocols?

- Example Questions:
    - Describe how DNS usesrecursive and non-recursive lookups to efficiently serve Internet Domain Names
    - HTTP over TLS is sometimes referred to as Layer-6, why is this naming convention accurate or inaccurate?
        - In the layering model we have used in this class, there is no layer 6 and since HTTP and TLS are realy working in unison they both belong at layer-5.
        - The application layer uses TLS for secure connection establishment, and sends an HTTP payload over said connection
    - Math problems:
        - Bandwidth calculation from HW1
        - IP Subnetting and binary arithmetic
            - How many hosts in a /24, /23, /25?
            - Network Mask
            - Brodcast addresses
    - Memorization of what the headers do in:
        - Ethernet
        - IPv4
        - TCP
        - UDP
    - Lots of tables:
        - Switch Table
        - ARP Table
        - Routing Forwarding Table
        - Connection Table
        - NAT Table
        - How is each table populated?
        - What protocols are used by each?
    - Application "Tables":
        - DNS records
        - Email Addresses
        - SIP addresses
    - Both Email and SIP addresses look and work similarly. Describe how they are formatted and how the application knows how to communicate with the appropriate host.
        - Both utilize DNS to query for their respective record types MX & SRV. MTA will do lookup for email and SIP proxy will for SIP



- Compare and contrast the FCS used in ethernet, the header checksum used in IPv4, and hash functions
  - The FCS in Ethernet is a simple sequence included at the end of an ethernet frame that does a complete checksum against the entire ethernet frame to be able to tell if any bits were modified on the wire
  - The header checksum in IPv4 is similar, though it only does a checksum for the IP header not the entire IP Datagram contents
  - Hash functions are similar, but use much more sophisticated and cryptograichally secure math (no twos compliment)
- How might IPv6 change spam prevention?
  - It becomes a lot harder due to the huge address space. Its very easy for spammers to change IP addresses. The DNSBLs are going to have to use full subnets in IPv6, but that comes with many downsides
- Internet Protocols are secure by default. Is this statement accurate or innaccurate?
  - Innaccurate! New protocols are required by the IETF to have a security component, but the protocols of old are definitely not secure by default. Security wasn;t a huge concern back in the day.
- True or False? Jitter is an important aspect of QoS when viewing webpages. Why or Why Not?
  - False! Due to the makeup of a webpage and how it is loaded, it is not affected by the Jitter metric. Jitter is used to measure the "Lip Sync" problem in real time voice & video streaming applications.