Introduction, History, and Internet Architecture

Introduction

1. Explore major milestones in the history of the Internet and review original design choices and principles of the Internet architecture
2. What would the Internet look like if we redesigned it from scratch?
   - Optimize for network control, management, and accountability, which were not first-class goals when the Internet was first designed
3. Over 22 billion devices connect to the Internet (phones, laptops, fridges)

Why Study Computer Networks?

1. Internet Growth
   - Internet started as a research experiment that escaped from a lab, but eventually evolved into a global communications infrastructure
   - Explosion of applications and technologies (IoT, vehicles, sensors, home devices) cause the number of users on the Internet to be constantly increasing
2. Networks play an instrumental role in our society
   - Internet changed the way we do business
     - E-commerce, advertising, cloud computing applications
   - Internet changed the way we communicate
     - E-mail, instant messaging, social networking, virtual worlds applications
   - Internet changed how we fight
     - Large scale cyber attacks, distribution of fake news, censorship, and nationwide attacks
   - These developments have legal implications both nationally and globally that we haven't had to consider in the past
3. Networking is a playground for interdisciplinary research innovations
   - Internet is an amazing "playground" of ongoing cross-disciplinary innovations from fields such as:
     - Distributed systems
     - Operating systems
     - Computer architecture
     - Software engineering
     - Algorithms and data structures
     - Graph theory
     - Queueing theory
     - Game theory
4. Networking offers multidisciplinary research opportunities with potential for impact
   - Research work spans fields such as Internet security, economics, and social sciences
   - Can design innovative and impactful solutions and immediately put them to use by leveraging existing platforms

A Brief History of the Internet

1. J.C.R Licklider proposed the "Galactic Network" (1962)
   - Licklider envisioned that everyone could quickly access data through a set of interconnected computers (connected computers in California and Massachusetts through low-speed dial-up telephone line)
2. The ARPANET (1969)
   - Results of first experiments showed that time-shared infrastructure was working sufficiently well
   - Researchers indicated the need for packet switching technology
   - ARPANET initially consisted of four computers
3. Network Control Protocol (NCP), an initial ARPANET host-to-host protocol (1970)
   - As more computers were added to ARPANET, research work proceeded to designing protocols
   - Initial ARPANET host-to-host protocol was called Network Control Protocol

- One of the first applications that launched was email in 1972

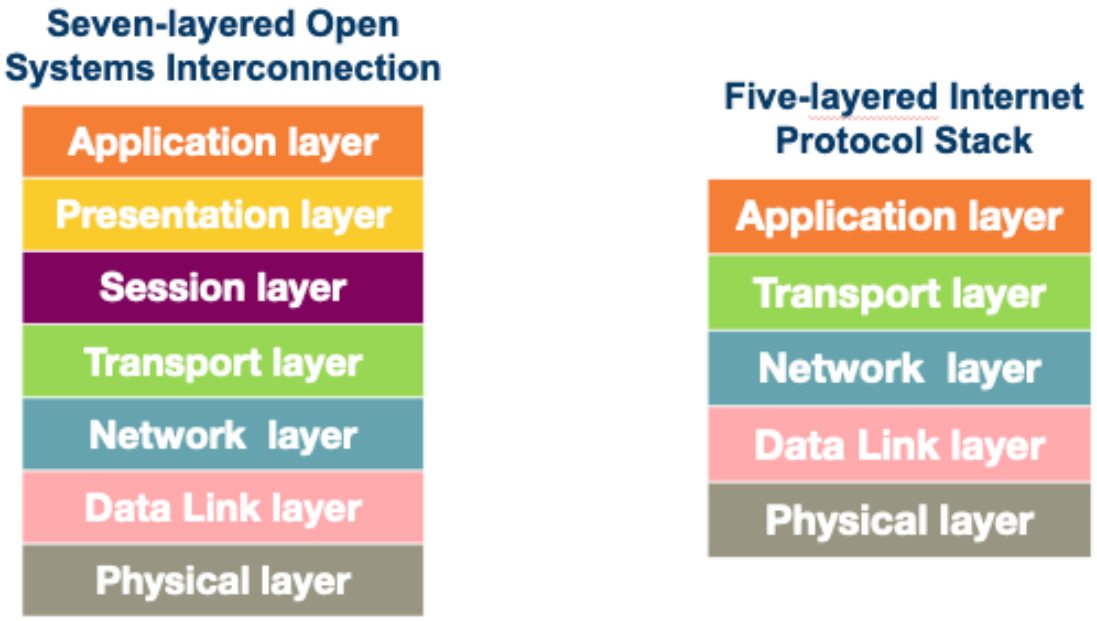
4. Internetworking and TCP/IP (1973)
   - Open-architecture networking enabled individual networks to be independently designed and developed in accordance with the specific environment and user requirements of that network
   - Led researchers to develop a new version of the NCP protocol which would eventually be called the Transmission Control Protocol/Internet Protocol (TCP/IP)
     - TCP was used for service features such as flow control and recovery from lost packets
     - IP was used only for addressing and forwarding individual packets
5. The Domain Name System (DNS) (1983) and World Wide Web (WWW) (1990)
   - As the scale of the Internet grew, it was no longer feasible to have a single table of hosts to store names and addresses
   - Domain Name System (DNS) was designed to translate domain names to IP addresses by a scalable distributed mechanism
   - One of the first and most popular applications was the World Wide Web, introduced by Tim Berners-Lee

Internet Architecture Introduction

1. The Internet architecture is what enables us to connect hosts running the same applications but located in different types of networks
   - How do clients communicate even though they are using very different networks/technologies? (Wifi vs Ethernet)
     - Designers of network protocols provide structure to the network architecture by organizing the protocols into layers
2. Architecture, layers, and functionalities
   - Functionalities in the network architecture are implemented by dividing the architectural model into layers. Each layer offers different services
   - Every layer implements some functionality
   - Every layer works based on the service provided by the layer below it, and also provides some service to the layer that is above
3. Layered architecture advantages: Scalability, modularity, and flexibility
   - Can add or delete components which make for cost-effective implementations

The OSI Model

1. The International Organization for Standardization (ISO) proposed the seven-layered OSI model shown below, which consists of the following layers:
   - Application
   - Presentation
   - Session
   - Transport
   - Network
   - Data link
   - Physical



OSI Model

2. Disadvantages of the layered protocol stack model:
   - Some layers functionality depends on the information from other layers, which can violate the goal of layer separation
   - One layer may duplicate lower layer functionalities. For example, the functionality of error recovery can occur in lower layers, but also on upper layers as well
   - Some additional overhead that is caused by the abstraction between layers
3. The traditional Internet architecture model only has five layers
   - The application, presentation, and session layers are combined into a single layer, and this combined layer is called the application layer
   - Sockets are the interface between the application layer and the transport layer

Application, Presentation, and Session Layers

1. Application Layer
   - Includes multiple protocols
     - HTTP (web)
     - SMTP (e-mail)
     - FTP (transfer files)
     - DNS (translates domain names to IP addresses)
   - Offers multiple services depending on the application that is implemented
   - Same is true for the interface though which it is accessed, and the protocol that is implemented
   - Refer to the packet of information as a message
2. Presentation Layer
   - Formats the information that it receives from the layer below and delivers it to the application layer
     - Formatting a video stream or translating integers from big endian to little endian format
3. Session Layer
   - Responsible for the mechanism that manages different transport streams that belong to the same session between end-user application processes
     - Tie audio and video streams together when teleconferencing

Transport and Network Layer

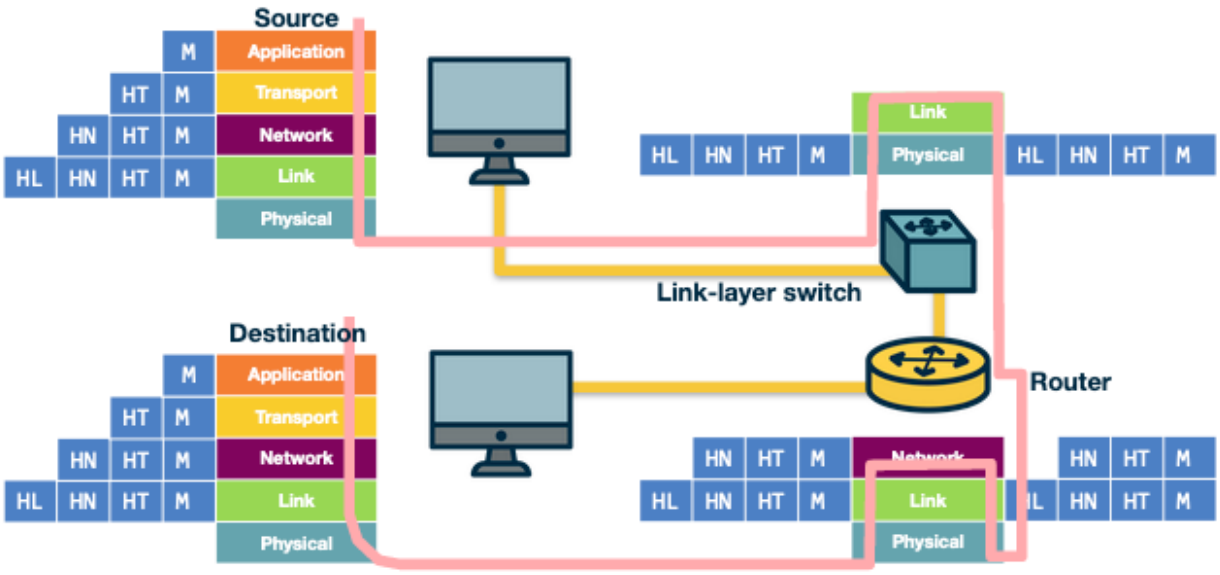
1. Transport Layer
   - Responsible for end-to-end communication between end hosts
   - Two protocols, TCP and UDP
   - TCP offers a connection-oriented service to the applications that are running on the layer above, guaranteed delivery of the application-layer messages, flow control which matches the sender's and receiver's speed, and a congestion-control mechanism so the sender slows its transmission rate when it perceives the network to be congested
   - UDP provides a connectionless best-effort service to the applications that are running in the layer above, without reliability, flow, or congestion control
   - Refer to a packet of information as a segment
2. Network layer
   - Responsible for moving datagrams from one Internet host to another
     - Deliver the datagram to the transport layer in the destination host
   - Protocols
     - IP protocol which defines:
       - The fields in the datagram
       - How the source/destination hosts and intermediate routers use these fields
     - The routing protocols that determine the routes that the datagrams can take between sources and destinations
   - Refer to the packet of information as a datagram

Data Link Layer and Physical Layer

1. Data Link Layer
   - Examples of protocols in this layer include: Ethernet, PPP, and WiFi
   - Responsible for moving frames from one node (host or router) to the next node
     - Assuming we have a sender and receiver host, the network layer will route the datagram through multiple routers across the path between the sender and receiver
   - Offers services that depend on the data link layer protocol that is used over the link, such as reliable delivery (different from TCP reliable delivery)
   - Refer to the packets of information as frames
2. Physical Layer
   - Facilitates the interaction with the actual hardware
   - Responsible for transferring bits within a frame between two nodes that are connected through a physical link
   - One of the main protocols, Ethernet, has different physical layer protocols for twisted-pair copper wire, coaxial cable, and single-mode fiber optics

Layers Encapsulation

1. Encapsulation
   - Process of taking data from one protocol and translating it into data that are used by another protocol, so the data can continue across a network
   - Each layer will add its own header to the message and forward the data to the next layer when going down the stack
   - When going up the stack, each layer strips off its header and forwards the data (called de-encapsulation)



Encapsulation

The End to End Principle

1. End-to-end (e2e) principle is a design choice that characterized and shaped the current architecture of the Internet
   - e2e principle suggests that specific application-level functions usually cannot, and preferably should not, be built into the lower levels of the system at the core of the network
     - The network core should be simple and minimal, while the end systems should carry the intelligence
   - People argue the e2e principle allowed the Internet to grow rapidly because evolving innovation took place at the network edge in the form of numerous application and a plethora of services, rather than in the middle of the network, which could be hard to layer modify
   - Goals: Moving functions and services closer to the application that use them increases the flexibility and the autonomy of the application designer to offer these services to the needs of the specific application
     - Higher-level protocol layers are more specific to an application
     - Lower-level protocol layers are free to organzie the lower-level network resources to achieve application design goals more efficiently and independently of the specific application

Quiz 1

1. Some data link layer protocols, such 802.11 (WiFi), implement some basic error correction as the physical medium used is easily prone to interference and noise (such as a nearby running microwave). Is this a violation of the end-to-end principle?
   - No, because violations of the e2e principle typically refer to scenarios where it is not possible to implement a functionality entirely at the end hosts, such as NAT and firewalls.
   - In this question, we have a lower level protocol implementing error checking

Violations of the End-to-End Principle and NAT Boxes

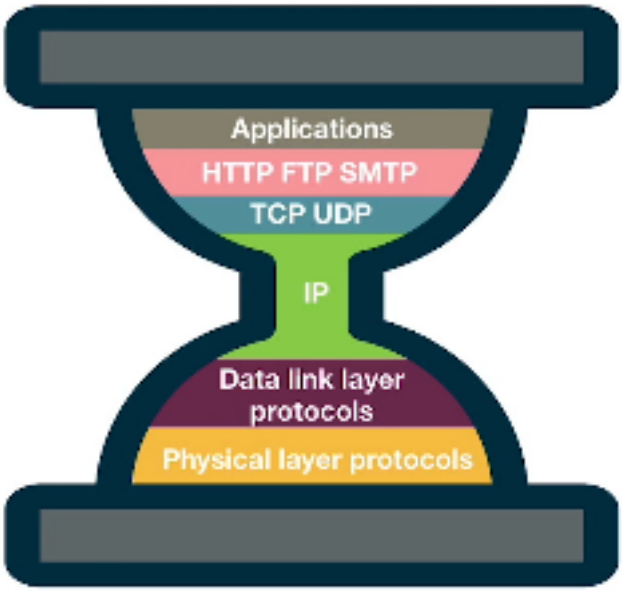
1. Firewalls and traffic filters
   - Firewalls operate at the periphery of a network and monitor the network traffic going through

- They violate the e2e principle since they are intermediate devices operated between two and hosts and can drop the end hosts' communication

2. NAT boxes
   - Hosts behind NAT boxes are not globally addressable or routable
   - Therefore, it is not possible for other hosts on the public Internet to initiate connections to these devices
   - A host behind a NAT and a host on the public Internet cannot communicate by default without the intervention of a NAT box

The Hourglass Shape of Internet Architecture

1. Many different protocols exist at the outer layers of the OSI model, while few exist at the transport layer
   - Why have there been more frequent innovations at other layers of the protocol hourglass?
   - Why have protocols at the waist of the hourglass (IPv4, TCP, and UDP) been difficult to replace, and have they outcompeted any protocols that offer the same or similar functionalities?



Hourglass

Evolutionary Architecture Model

1. EvoArch is a model for understanding why some protocols persist and others are replaced often
   - Model the connections between layers as a directed acylic graph
   - Associate probabilities for new protocols to be added and removed
   - Discrete-time model that is executed over rounds
2. Implications for the Internet Architecture
   - TCP/IP was initially not trying to compete with telephone network services, so it was able to grow without competing or being threatened by the telephone network
     - FTP, e-mail, Telnet
   - As it grew, numerous powerful applications relied on it
   - Because TCP/IP, UDP, and IPv4 have so many things depending on them, they are difficult to replace
   - EvoArch suggests that even if new architectures do not have the shape of an hourglass initially, they probably will as they evolve, which will lead to new ossified protocols

Quiz 2

1. Which of the following are ramifications of the "hourglass shape of the Internet"?
    - Many technologies that were not originally designed for the internet have been modified so that they have versions that can communicate over the Internet (such as Radio over IP). (true)
    - It has been a difficult and slow process to transition to IPv6, despite the shortage of public IPv4 addresses. (true)
    - Applications like BitTorrent leverage peer-to-peer networking instead of a more traditional client-server model for better performance. (false)

Architecture Redesign

1. Why a clean-slate design approach?
    - Major design principles of current Internet architecture
        - Layering
        - Packet switching
        - Network of collaborating networks
        - Intelligent end-systems
        - End-to-end argument
    - Challenges with current Internet architecture
        - Security
        - Resilience and availability
        - Scalability and management
        - Quality of service
        - User experience
        - Economics
    - Clean-slate approach means using out of the box thinking to redesign the Internet based on evolving needs and what we've learned
2. Clean-slate design as a process
    - Clean-slate design means creating an ecosystem where different ideas can be compared and evaluated
    - Allows us to determine what the blueprint for the future Internet might look like
3. Redesigning the Internet architecture to optimize for Control and Management
    - Research group 4D investigates an extreme design point where the decision logic is completely separated from distributed protocols
    - Trends toward more powerful, reliable, and inexpensive computing platforms make their design point attractive
4. Redesigning the Internet Architecture to offer better accountability
    - IP network layer provides little to no protection against misconfiguration or malicious actions which occur frequently
    - Source accountability is the ability to trace actions to a particular end host and stop that host from misbehaving
    - Control-plane accountability is the ability to pinpoint and prevent attacks on routing

Interconnecting Hosts and Networks

1. Repeaters and hubs
    - Operate in the physical layer (L1)
    - Provide connectivity between hosts that are directly connected in the same network
2. Bridges and Layer 2 Switches
    - Enable communication between hosts that are not directly connected
    - Operate in the data link layer (L2) based on MAC addresses
    - Receive packets and forward them to reach the appropriate destination
3. Routers and Layer 3 Switches

Learning Bridges

1. A bridge is a device with multiple inputs/outputs
   - Transfers frames from an input to one (or multiple) outputs
   - Doesn't need to forward all the frames in receives
   - A learning bridge learns, populates, and maintains forwarding table
     - Consults its forwarding table so that it only forwards frames on specific ports, rather than all ports
2. How does a bridge learn?
   - When the bridge receives any frame, it is a "learning opportunity" to know which hosts are reachable through which ports
   - The bridge can view the port over which a frame arrives and the source host

Looping Problem in Bridges and the Spanning Tree Algorithm

1. Using bridges to connect LANs fails if the network topology results in loops
   - Solution: Exclude links that lead to loops by running the spanning tree algorithm
   - Bridges are represented as nodes and links between bridges are represented as edges
2. Spanning Tree Algorithm
   - Every node (bridge) in the graph has an ID. The bridges eventually select one bridge as the root of the topology
   - Algorithm runs in rounds. Each round, each nodes to each neighbor a message with three fields
     - Sending node's ID
     - ID of the root as perceived by the sending node
     - The number of hops between that (perceived) root and the sending node
   - At every round, each node keeps track of the best configuration message that it has received so far and compares that against the configuration messages it receives from the neighboring nodes at that round
   - The first round of the algorithm, every node thinks that it is the root
   - How does a node compare two configuration messages? Between two configurations, a node selects one configuration as better if:
     - The root of the configuration has a smaller ID
     - The roots have equal IDs, but one configuration indicates a smaller distance from the root
     - Both root IDs are the same and the distances are the same, then the node breaks the tie by selecting the configuration of the sending node that has the smallest ID
   - A node stops sending configuration messages when the node receives a configuration message that indicates that it is not the root. This happens when it receives a message that is either:
     - Closer to the root
     - Has the same distance from the root, but a smaller ID

Quiz 3

1. Which of the following statements are correct?
   - The Spanning Tree Algorithm helps to prevent broadcast storms (true)
   - The Spanning Tree Algorithm presented in this lecture always results in a spanning tree that places the root in a topologically central location, so that all the nodes are as "close" as possible to the root. (false)
   - Network traffic cannot traverse an inactive link. (false)

Layer	Function	Example
Application (7)	Services that are used with end user applications	SMTP,
Presentation (6)	Formats the data so that it can be viewed by the user Encrypt and decrypt	JPG, GIF, HTTPS, SSL, TLS
Session (5)	Establishes/ends connections between two hosts	NetBIOS, PPTP
Transport (4)	Responsible for the transport protocol and error handling	TCP, UDP
Network (3)	Reads the IP address form the packet.	Routers, Layer 3 Switches
Data Link (2)	Reads the MAC address from the data packet	Switches
Physical (1)	Send data on to the physical wire.	Hubs, NICS, Cable

OSI with Protocol Examples