ECE 448/528
Application Software Design

Lecture 4. TCP/IP Networking
Spring 2025

Won-Jae Yi, Ph.D.

Department of Electrical and Computer Engineering
Illinois Institute of Technology

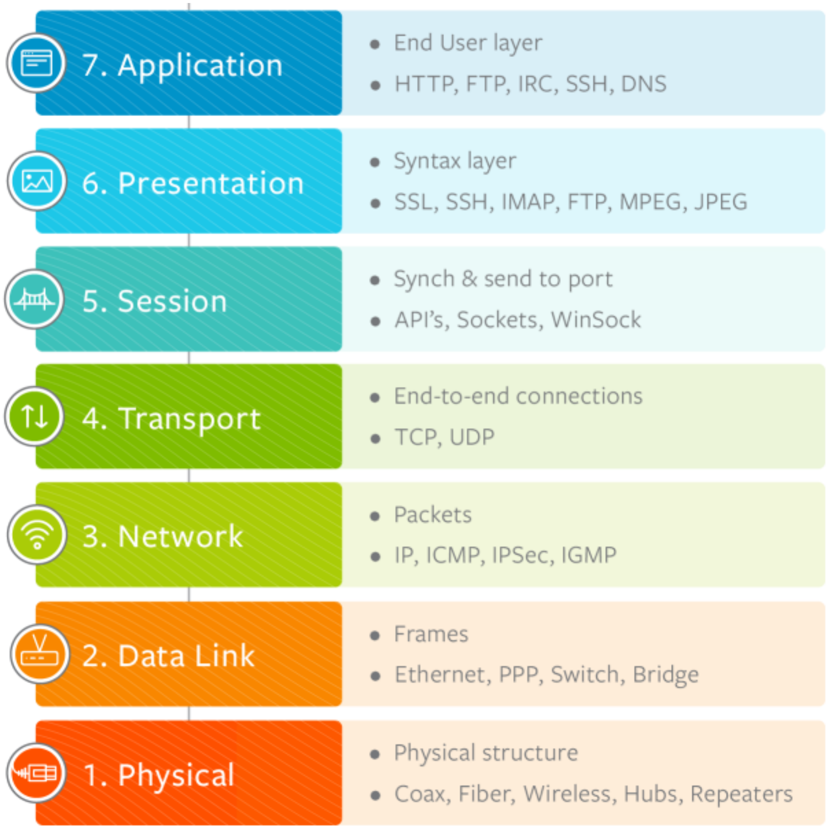
Computer Networking

Computer Networking as Layered Services

- Computer networking involves a lot of physical devices and communication protocols.
  - Different physical media: wired (copper, fiber), wireless, etc.
  - Different requirements: latency, throughput, reliability, etc.
  - Different vendors in different sectors.
- Use a layered approach to ensure everything works together
  - Layers are stacked on top of each other.
  - Each layer provides services to layers above by using services of the layer below.

ISO/OSI 7-Layer Model

- A well-established model of networking.
  - Consists of 7 layers.
  - Cover areas from application to physical media.
- While practical network services do not follow this theoretical model exactly, it will help us understand many protocols' purposes.
  - As "the big picture" to facilitate our learning process.



The Upper 5 ISO/OSI Layers

- **Application**: how applications make use of the network.
  - Via sending and receiving data.
  - e.g. RESTful, HTTP, FTP.
- **Presentation**: represent data and data structures as bytes.
  - Bytes sent and received over end-to-end channels.
  - e.g. JSON, HTML, XML.
- **Session**: better utilization of end-to-end channels.
  - Authentication, encryption, compression, multiplexing, etc.
  - e.g. RPC, PPTP.
- **Transport**: form end-to-end communication channels.
  - On top of packets moving between nodes (hosts).
  - e.g. TCP, UDP.
- **Network**: node addressing and packet routing.
  - On top of communication between directly connected nodes.
  - Mostly IP nowadays – store and forward IP packets.

The Lower 2 ISO/OSI Layers

- **(Data) Link**: communication between directly connected node
  - Consists of a few sub-layers to support the complex hardware/software communication interface.
  - Master/slave synchronization
  - Addressing and multiplexing.
  - Media access control (MAC).
  - e.g. part of Ethernet and Wi-Fi protocols.
- **Physical**: how to represent bits as physical signals.
  - We'll focus on the upper 5 layers, introduce link layer topics as needed, and leave physical layers to ECE courses on communication.

IP (Internet Protocol)

Ethernet

- A family of protocols to support TCP/IP networking covering data links and physical layers.
  - Commercially available and standardized in the early ’80s.
- Shared medium, nodes may enter and leave freely.
  - Originally coaxial cable, nowadays twisted pair (Cat 5e, Cat 6, etc.) and fiber optic.
  - Speed: 100Mb/s, 1Gb/s, 10Gb/s, 40Gb/s, 100Gb/s, etc.

Ethernet Data Link

- Usually known as Layer 2 or L2.
- Network interface: a software entity to access the network.
  - Managed with the command `ifconfig` or `ipconfig`.
  - It is common for computers nowadays to have multiple network interfaces, some associating with actual hardware and some not.
- MAC address: one per network interface
  - 48-bit, globally unique.
  - OUI (Organizationally Unique Identifier), the first 24 bits of MAC, is assigned by IEEE to uniquely identifies a vendor or manufacturer.
- Ethernet frame: unit of data link layer data.
  - Addresses of source and destination, payload, checksum, etc.
  - Minimum size: 64 bytes.
  - Plus 20 bytes of physical layer overhead. You can’t send more than 1500 bytes (MTU) Ethernet frames over 1Gb/s Ethernet.

Ethernet Hub and Switch

- How to extend Ethernet networks by interconnecting cables?
  - As the same physical medium.
  - Beyond cable lengths allowed by Ethernet specifications.
  - Allow cables to branch.
- Use a device that have multiple ports, where each port can connect to a cable.
- (Ethernet) Hub: simply repeat frames on all ports.
  - Simple but dumb. Not suitable for faster network. Obsolete.
- (Network) Switch: repeat frames when necessary.
  - Smart: memorize which MAC addresses are from which ports.
  - Need additional processing power than hub. Usually based on ASIC chips but could be done via CPUs.

IP Networking

- Internet Protocol (IP)
  - A network layer protocol, usually on top of Ethernet.
  - Two popular versions: IPv4, IPv6.
  - Let's focus on IPv4.
- IP packets
  - As payload of Ethernet frame.
  - Size-limited to facilitate store-and-forward communication.
  - Between nodes that are not necessarily directly connected.
- Node needs to be configured to join an IP network.
  - Manually or automatically via DHCP (Dynamic Host Controller Protocol).
  - The configuration reveals a lot of how IP networking works.
  - Minimum: IP addresses and subnets.
  - Optional: gateway addresses and routing table.

IPv4 Addresses and Subnets

- IPv4 address: dotted quad format, e.g. 192.168.1.100
  - 32 bits – we only have about 4.3 billion addresses.
- IPv4 subnet: group of IP addresses with the same prefix.
  - Prefix is either as a mask in dotted quad format, e.g. 255.255.255.0, or as the length of the prefix, e.g. /24
  - Overall, as the IP address plus the prefix, e.g. 192.168.1.100/255.255.255.0, or 192.168.1.100/24
  - Lowest address, e.g. 192.168.1.0, is for the network itself.
  - Highest address, e.g. 192.168.1.255, is the broadcast address.
- Typically, one address and subnet prefix per network interface.
  - A node may connect to multiple subnets via multiple interfaces.
- Interfaces on different nodes with IP addresses belonging to the same subnet are assumed to be connected to the same physical medium.

IPv4 Routing: Same Subnet

- How to send an IP packet to an IP address belonging to a subnet this node connects to?

1. Locate the source interface with the same subnet.
2. Discover the destination MAC address associated with the destination IP address.
    - Use the ARP (Address Resolution Protocol).
    - Broadcast over the physical medium or use cached information.
3. Create and send the Ethernet frame.
    - The IP packet as the payload.
    - From the source interface
    - (Directly) To the destination MAC.

IPv4 Routing: Different Subnet

- How to send an IP packet to an IP address not belonging to any subnet this node connects to?
- Default gateway address
  - The IP address of a node that is able to relay the packets.
  - Must belong to a subnet this node connects to.
- Gateway operation
  - The source node creates and sends an Ethernet frame with the IP packet as the payload (actual destination IP address) and the gateway MAC address as the destination MAC.
  - The gateway node receives such an Ethernet frame and forwards the IP packet payload to the destination IP address directly or via additional gateways.

IPv4 Routing: Summary

- Overall, all routing information on a node is stored in a routing table.
  - Managed via the command `route`.
- Typically, the routing table contains
  - Routing rules for connected subnets to respective interfaces.
  - Routing rules for other subnets to (default) gateways.
- Gateway nodes utilize routing protocols to update their routing table and use the routing table to decide where to forward IP packets.
- Router: dedicated gateway node
  - Use ASIC instead of software for better performance.
  - May contain a firewall to filter packets and control routing.
  - Advanced Ethernet switches may also perform routing – we call them L3 switches although they are routers.
- Use the command `ping` to check if a node can be reached.

Special IPv4 Addresses

- **Loopback/localhost/`lo`**
  - Provide network access to the node itself, w/o the need to have actual networking hardware.
  - The subnet 127.0.0.0/8, though usually as 127.0.0.1
  - Widely used for development and testing.
  - Improve security for production by limiting access of network services to applications running on the same node.
- **Public IP addresses**
  - Addresses that can be reached over the Internet.
  - Scarce resources considering how many devices we have today.
  - Allocated hierarchically from Internet Assigned Numbers Authority (IANA).
- **Private networks**
  - Allow organizations and families to manage IP addresses for their own devices without the need to contact any authority.
  - Three subnets: 10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16
  - But how could we access the Internet with such addresses?

UDP and TCP

Transport Layer Protocols

- IP networking enables nodes to communicate with each other
  - Not convenient for applications if they need to communicate over the same pair of nodes.
- The size of IP packets is limited.
  - In theory 65,535 bytes, but practically much smaller for efficiency.
  - Applications prefer to work with arbitrary message sizes.
- IP networking performs best-effort delivery.
  - Packets may be dropped if a router is dead or busy.
  - Packets may arrive out-of-order as routing configurations may change dynamically.
- Transport layer services
  - Multiplexing: build end-to-end channels for applications on top of node-to-node packet communication.
  - Support arbitrary message sizes.
  - Guaranteed and in-order delivery (and error reporting).

UDP (User Datagram Protocol)

- A simple transport layer protocol supporting <u>byte messages</u>.
  - A “thin” layer on top of IP – message delivery is not guaranteed, and messages may arrive out-of-order.
  - Good for latency-sensitive applications that don’t or don’t need to care about lost messages or out-of-order arrivals.
- Use a port number to distinguish different channels.
  - 16 bits, 0–65,535
- Together with the IP address, a UDP address is usually specified as `ip:port`.
- Connectionless: once an application opens a UDP port, it can send/receive UDP messages to/from any other UDP address.
- Support long messages by breaking them into multiple IP packets.
  - Receiver must wait for all IP packets to arrive.
  - Not an ideal use case for UDP.

TCP (Transmission Control Protocol)

- Guaranteed and in-order delivery of a stream of bytes.
  - NOT messages as assumed for IP and UDP protocols – a presentation layer protocol is always needed to extract messages from the byte stream.
  - Use several timers to report communication errors.
- Similar to UDP, use a port number 0–65,535 to support channels, and the TCP address is written as `ip:port` as well.
- Connection-oriented: server and client
  - Server: open a TCP port and wait for clients to connect.
  - Client: open a TCP port and connect to a single server.
- While the above would be sufficient for us to write simple server/client applications using TCP, you are recommended to take a course on networking to learn more about it, especially for performance tuning.

Network Address Translation (NAT)

- A set of mechanisms that modify IP packets to remap IP addresses and/or transport layer ports.
- IP masquerading: a NAT mechanism enabling UDP and TCP communications between private networks and the Internet.
  - Need one public IP address for a private network.
  - Help to save IPv4 addresses.
- A TCP example
  - Client at 192.168.1.100:5678 and server at 172.217.9.36:80
  - Gateway: private side 192.168.1.1, public side 104.194.116.100

NAT Example: Client Sending

1. First TCP packet: from the client to the server.
   - 192.168.1.100:5678 → 172.217.9.36:80
2. Packet reaches the gateway. The gateway allocates and memorizes an unused TCP port from itself, say 12345.
3. The gateway modifies the source IP address and the source TCP port of the packet.
   - 104.194.116.100:12345 → 172.217.9.36:80
   - Otherwise, when the server replies with a packet, no gateway knows who 192.168.1.100 is. Of course, many gateways have their own 192.168.1.100, so who is who?

- The gateway repeats Step 3. for any additional packets from 192.168.1.100:5678 to 172.217.9.36:80
  - Until disconnected.

NAT Example: Server Replying

1. The server replies with a packet.
   - 172.217.9.36:80 → 104.194.116.100:12345
2. Packet reaches the gateway. The gateway recalls that it is for 192.168.1.100:5678.
3. The gateway modifies the destination IP address and the destination TCP port of the packet.
   - 172.217.9.36:80 → 192.168.1.100:5678
   - The client believes that it is talking to the server directly.

Summary

- Computer networking utilizes layers to facilitate reasoning and implementation.
- Use commands like `ifconfig`, `route`, and `ping` to learn more about IP networking.
- UDP and TCP build on top of IP and provide quite different services.