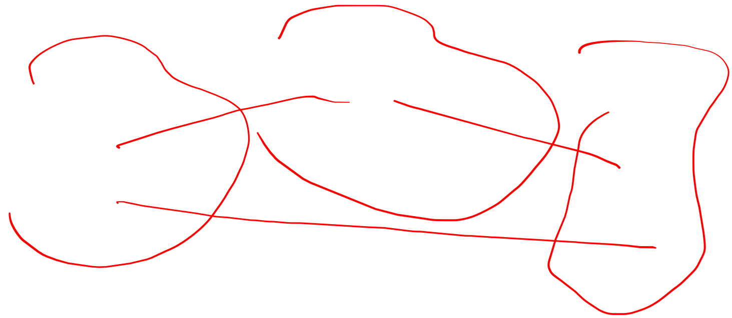
Design and Engineering of Computer Systems

Lecture 21:
Introduction to computer networking

Mythili Vutukuru
IIT Bombay

This week: how the Internet works

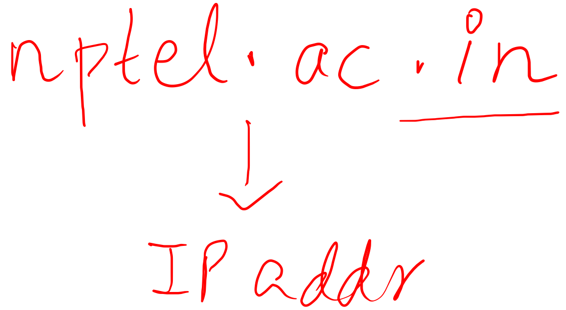


- Real world computer systems have many interconnected components that communicate over a computer network
- Internet: interconnected network of various smaller networks (regional, national, wired, wireless, mobile data networks)
  - Small "private" networks exist too, that are not connected to the Internet
- Clients and servers of a computer system exchange messages over the Internet via packets
  - Multiple components of a system in the cloud also communicate over network
- This week: principles of networking, how the Internet works
  - Not every topic in computer networking will be covered, but enough to help us realize the broad goals of this course

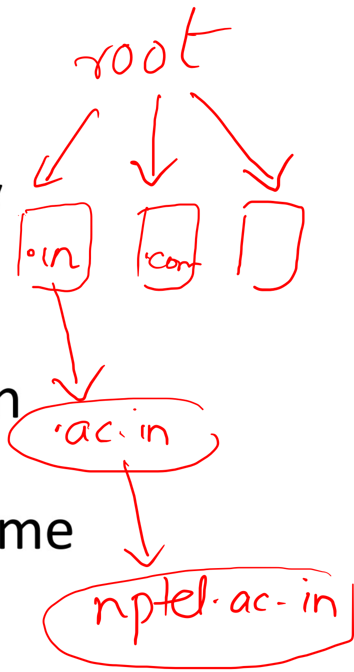
Example: accessing a web site

- What happens when you access a web site?
  - Internet services are provided by application/web servers (i.e., processes on servers running at certain port numbers) connected to the Internet
  - Clients access Internet services via URLs (https://nptel.ac.in/../..)
  - A special service called DNS (Domain Name Service) resolves the domain name of a server (nptel.ac.in) to its IP address (port number is usually well known)
  - Applications running on end hosts (clients and servers) send/receive messages via sockets, OS converts messages into packets and sends them over the network
  - Packets contain actual message being exchanged along with headers (e.g., source/destination IP address/port number)
  - Routers on the Internet route packets from source to destination using IP addresses
  - Transport protocols running on end hosts ensure reliable delivery, congestion control

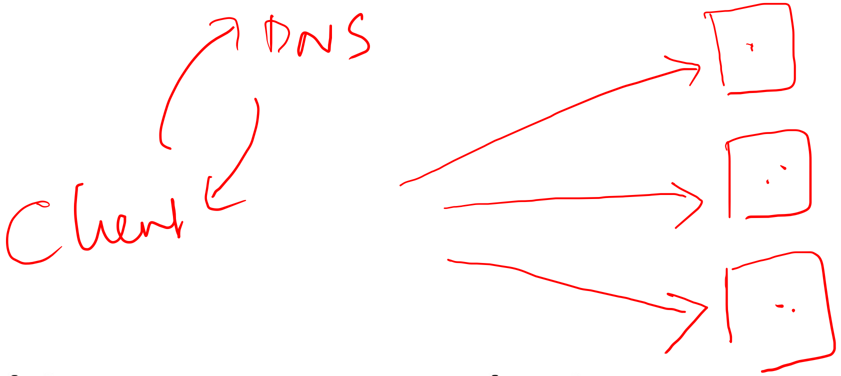
Domain Name System (DNS)



- Mapping between domain names of servers to IP addresses are stored in DNS records
- How are DNS records accessed?
  - DNS records are stored at authoritative domain name servers hierarchically
  - A few root servers know information about the top-level domain servers, which each handle one top-level domain (“.in”, “.com”, “.edu”)
  - Domain name servers at each level know about the domain name servers in the next level (“.in” server knows IP address of server handling “.ac.in”)
  - Local DNS resolver (one per group of hosts) recursively contacts domain name servers down the chain to obtain final IP address corresponding to domain name
  - DNS records are cached for small periods at local DNS resolver

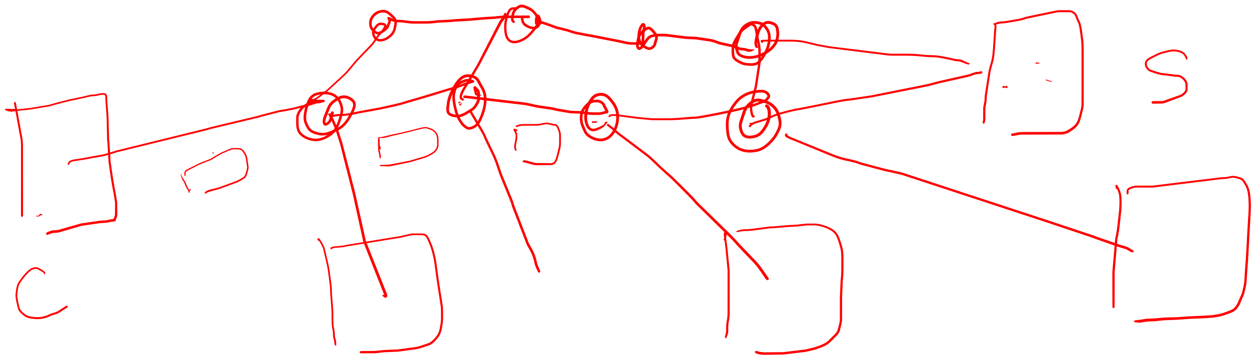


DNS and CDNs

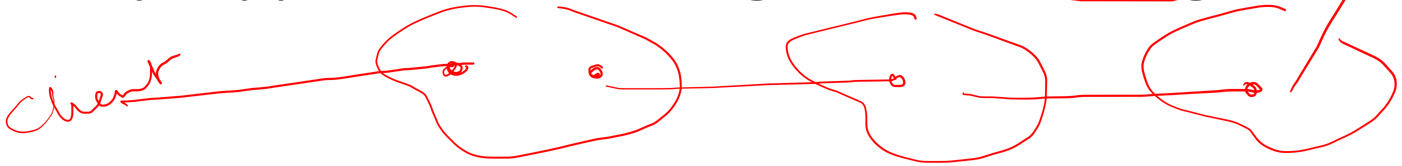


- DNS is useful for purposes beyond just name resolution
- Load balancing: DNS can return multiple IP addresses of a server, client can contact any one, to distribute load across servers
- Content Distribution Networks (CDNs): organizations that take static content from web sites and replicate it geographically across multiple servers on the Internet, for faster access
  - Dynamic content (e.g., user-specific content) still obtained from main website
- DNS query for a domain name redirects to the geographically closest CDN server replica with the content, for quicker access

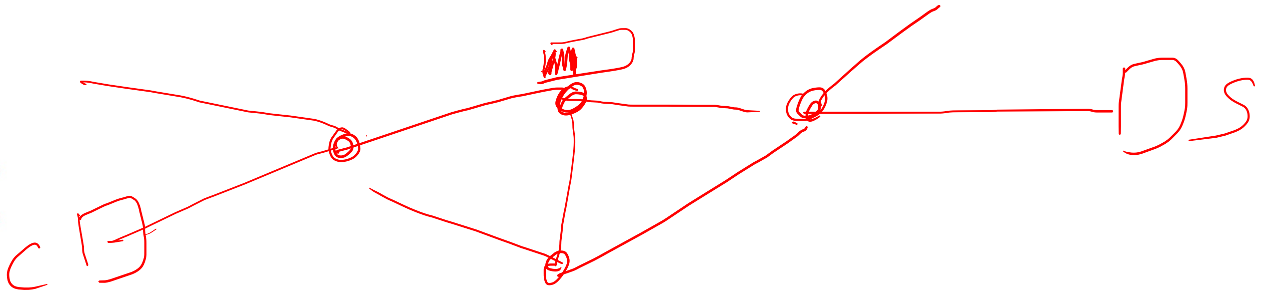
IP routing



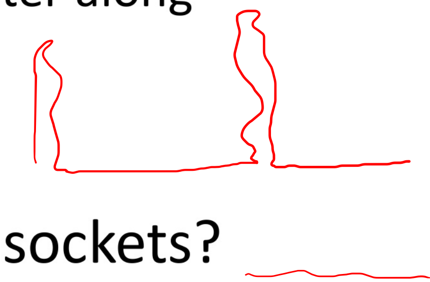
- Every end host on internet has a unique IP (Internet Protocol) address
- IPv4 addresses (widely used) are 32 bits (e.g., 192.168.10.1)
  - Can only support 2^32 unique end hosts (what if there are more?)
  - Upgraded IPv6 addresses are 128 bits in size
- Destination-based forwarding: IP routers look at destination IP address and route a packet on its (approx.) shortest path to its destination
- Routing protocols running on routers compute approx. shortest paths between different end hosts, and use this information for forwarding
- Many different types of networks, different applications and services can all connect to each other if they support IP addressing and forwarding



End-to-end argument



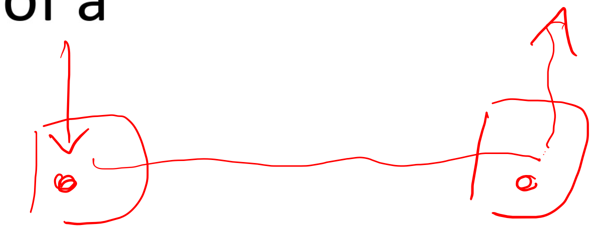
- Internet follows packet switching architecture: inspect each packet header and forward it as best as possible, no notion of end to end connection
  - If link is busy, incoming packet is stored at router, forwarded later (store-and-forward)
  - Best effort forwarding, packets may be dropped (storage at router overflows) or delayed
  - Packet headers add overhead, especially for very small packet sizes
- Earlier telephone networks followed concept of circuit switching
  - Setup an end to end path for a particular connection, reserve resources at each router along the path for duration of connection
- Packet switching works better for bursty Internet traffic due to statistical multiplexing (multiple connections share same link on demand)



- What about end-to-end connections established using connection-based sockets?
  - Protocols running at end host take care of reliability, IP routers not aware of connection
- This is called the end-to-end argument in Internet design: keep all complexity like reliability at end hosts, switches in the middle of network are simple

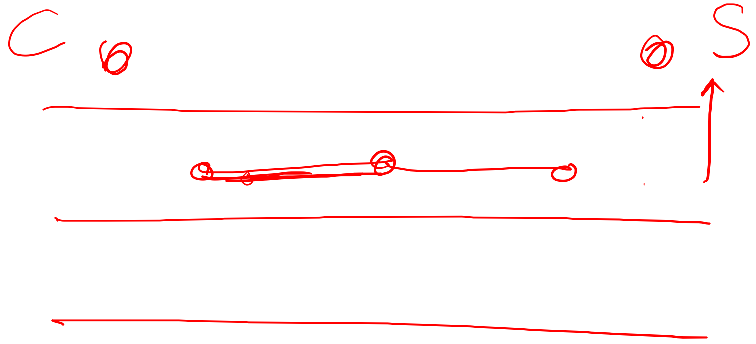
Transport protocols

- Transport protocols like TCP (transmission control protocol) running at the end hosts guarantee in-order reliable delivery of byte stream of a connection
  - Slow down sender if network is congested (congestion control)
  - Slow down sender if receiver is overwhelmed (flow control)
  - Retransmit packets that have been lost
  - Reorder received packets and deliver in order to application



- Several transport protocols available today (TCP, UDP, SCTP)
- Different applications can reuse common transport protocol logic by using suitable sockets, no need to reinvent mechanisms
- Protocol defines set of messages exchanged, packet format required to achieve a specific purpose, and so on

The concept of layering

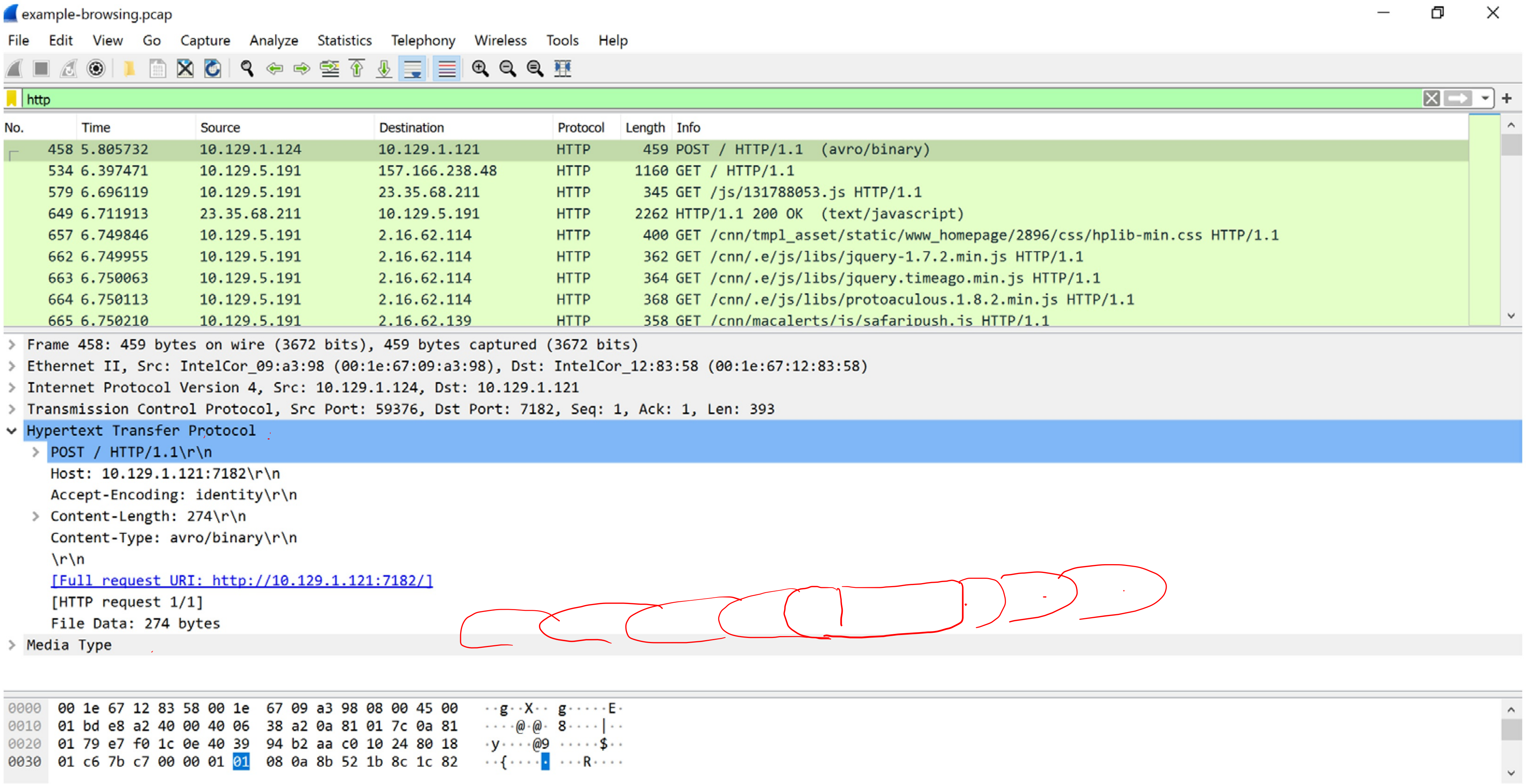


- Layering: a way of imposing modularity in computer systems
  - Peers in a layer only communicate with each other, not with entities in other layers
  - Lower layers provide a service/abstraction to higher layers
- Internet software is designed according to principle of layering
  - Clients and servers only exchange messages with each other, not concerned with how message reaches other side (application layer)
  - Transport protocol running at end hosts takes care of reliability and congestion/flow control, irrespective of application (transport layer/L4)
  - IP routers forward packets from one IP address to another, without worrying about the content of packets (IP/network layer/L3)
  - Link layer/L2 forwards traffic from one IP router to another, across many different types of links (wireless/wired) via link layer switches
  - Physical layer/L1 takes care of how bits are sent/received over wire or over air
- Layering simplifies the design of computer network software and hardware

Where are the layers?

- Application layer is user software that sends/receives messages using sockets or other such networking APIs
- Transport layer present in OS exchanges transport layer segments between end hosts with suitable reliability and congestion control
  - Segment = App layer message + transport layer headers added by OS
  - Modern OS support multiple transport layer protocols (e.g., TCP, UDP, SCTP)
- Network layer at end hosts and intermediate IP routers forwards IP datagrams based on destination IP address
- Link layer present in OS device drivers, network interface cards (NICs), switches forwards link layer frames from one IP router to another
- Physical layer implemented fully in NIC hardware, forwards digital bits via modulation/demodulation into analog signals
- Packet = generic term for unit of communication, has specific name at each layer

Packet capture tools: Wireshark



Summary

- In this lecture:
  - Introduction to the Internet
  - Design principles of layering, packet switching, end-to-end argument
- Use packet capture tools such as Wireshark to capture the packets going in and out of your computer. Look at the packets to see the various protocols and headers, across different layers.