Protocol Stack and Layers Summary

CSC 249
March 29, 2018

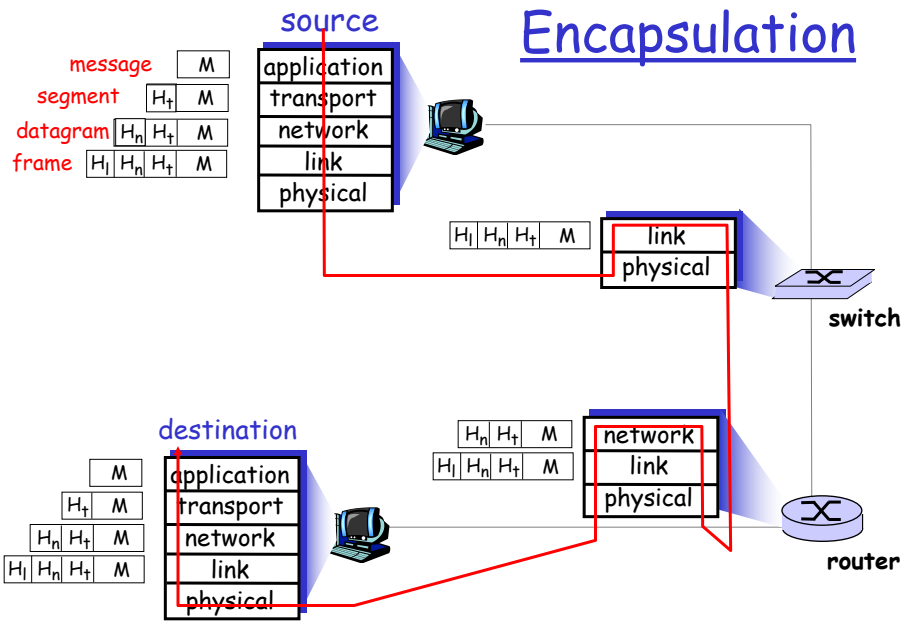
1

Chapter 6: Summary

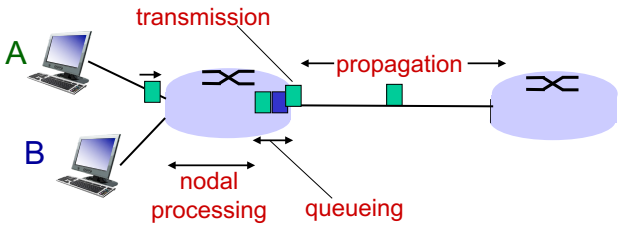
- Data link layer services:
  - error detection and correction
  - sharing a broadcast channel: multiple access
  - link layer addressing
  - Plug-and-play for ARP and switch table learning
- Link layer technologies
  - Ethernet
  - switched LANS (switches v. hubs)

2

Encapsulation



Four sources of packet delay



$$d_{total} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

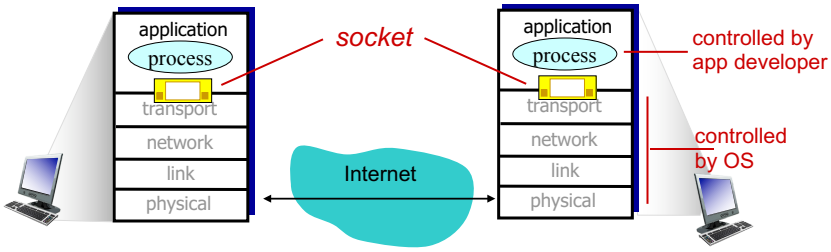
d_{trans}: transmission delay:
- L: packet length (bits)
- R: link *bandwidth (bps)*
- $d_{trans} = L/R$

d_{prop}: propagation delay:
- d: length of physical link
- s: propagation speed in medium (~2×10^8 m/sec)
- $d_{prop} = d/s$

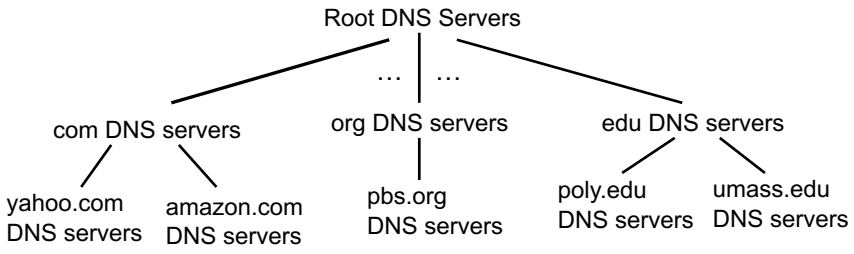
d_{trans} and d_{prop} *very* different

Sockets

- process sends/receives messages to/from its socket
  - sending process sends message through the socket
  - sending process relies on transport infrastructure, UDP or TCP as programmed into the operating system, to deliver the message to the socket at the receiving host & process



DNS: a distributed, hierarchical database



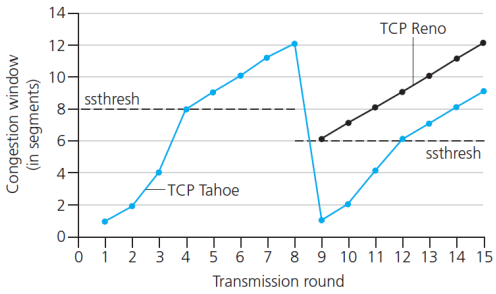
a host, or client, wants the IP address for www.google.com

1) Client (local server) queries root server to find the .com DNS server
2) Client queries .com DNS server to get google.com DNS server
3) Client queries google.com DNS server to get the IP address for www.google.com

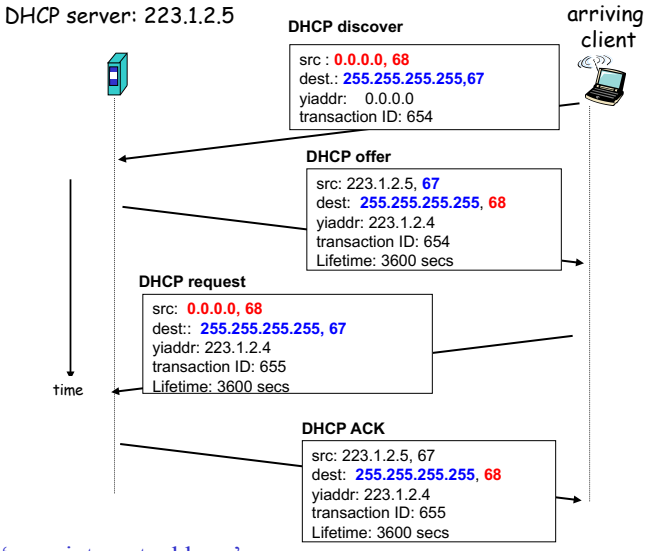
TCP: slow start & congestion avoidance

Implementation:

- variable CWND
- variable **`ssthresh`**
- Loss event
  - Triple duplicate ACK
  - timeout



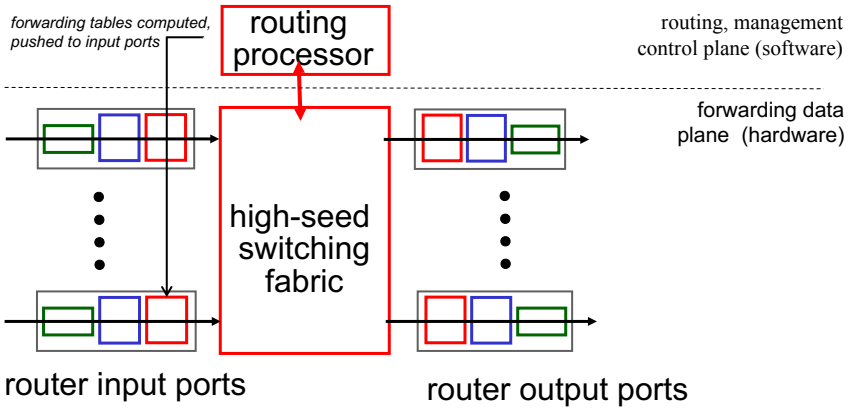
DHCP client-server scenario



yiaddr = 'your internet address'

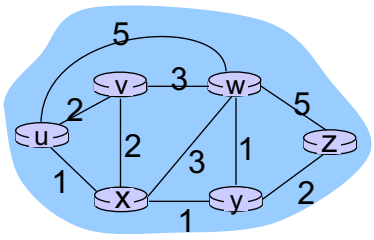
broadcast address, 255.255.255.255 → sent to every host *in the subnet*

Router architecture overview

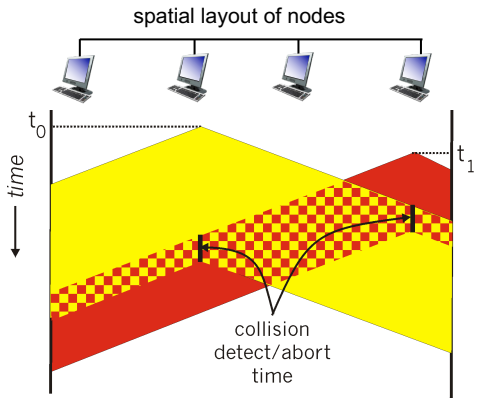


Routing: Dijkstra's algorithm

Step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



CSMA/CD (collision detection)



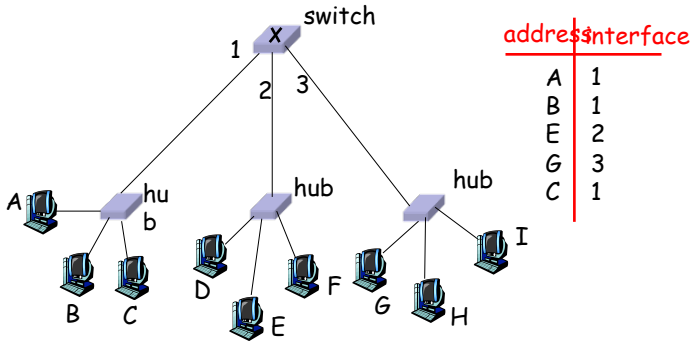
Slide Example: Creating an ARP Table

For the same LAN segment:

- 'A' wants to send datagram to 'B,' and B's MAC address not in A's ARP table.
- 'A' broadcasts ARP query packet, containing B's IP address
  - Dest MAC address = FF-FF-FF-FF-FF-FF
  - All machines on LAN receive ARP query
  - ARP Packets contain IP & MAC address for source and destination
  - A caches (saves) IP-to-MAC address pair in its ARP table
- B receives ARP packet, responds to A with its (B's) MAC address
  - **Why does only 'B' respond?**
  - frame sent directly to A's MAC address (not broadcast)
- ARP is "plug-and-play":
  - nodes create their ARP tables *without intervention from net administrator*

Switch table example

Suppose C sends frame to B



- Switch receives frame from from C
  - Notes that B and C are in same segment
  - Switch does nothing

13

Layer	Protocols, and their main features	Happens TO Packet	Action caused BY packet	Action/Event happens on own
Application				
Transport				
Network				
Link				