Data center networks

- 10's to 100's of thousands of hosts, often closely coupled, in close proximity:
 - e-business (e.g. Amazon)
 - content-servers (e.g., YouTube, Akamai, Apple, Microsoft)
 - search engines, data mining (e.g., Google)
- challenges:
 - multiple applications, each serving massive numbers of clients
 - managing/balancing load, avoiding processing, networking, data bottlenecks



Inside a 40-ft Microsoft container, Chicago data center

Data center networks



Top-Of-Rack (TOR) - Network Connectivity Architecture



Data center networks

- rich interconnection among switches, racks:
 - increased throughput between racks (multiple routing paths possible)
 - increased reliability via redundancy



Synthesis: a day in the life of a web request

- journey down protocol stack complete!
 application, transport, network, link
- putting-it-all-together: synthesis!
 - goal: identify, review, understand protocols (at all layers) involved in seemingly simple scenario: requesting www page
 - scenario: student attaches laptop to campus network, requests/receives www.google.com

When student attaches laptop to campus network, requests/receives www.google.com, how many servers the laptop will send application level requests?

A. 1

B. 2

C. 3

D. 4

E. All above are possible

A day in the life: scenario



When student attaches laptop to campus network, requests/receives www.google.com, how minimum/maximum number of separate protocols the laptop will use?

- A. 1/2
- B. 1/3
- C. 1/4
- D. 2/3
- E. 2/4

A day in the life... connecting to the Internet



- connecting laptop needs to get its own IP address, addr of first-hop router, addr of DNS server: use DHCP
- DHCP request encapsulated in UDP, encapsulated in IP, encapsulated in 802.3 Ethernet
- Ethernet frame broadcast (dest: FFFFFFFFFFF) on LAN, received at router running DHCP server
- Ethernet demuxed to IP demuxed, UDP demuxed to DHCP

A day in the life... connecting to the Internet



- DHCP server formulates
 DHCP ACK containing
 client's IP address, IP
 address of first-hop router
 for client, name & IP
 address of DNS server
- encapsulation at DHCP server, frame forwarded (switch learning) through LAN, demultiplexing at client
- DHCP client receives DHCP ACK reply

Client now has IP address, knows name & addr of DNS server, IP address of its first-hop router

A day in the life... ARP (before DNS, before HTTP)



- before sending HTTP request, need IP address of www.google.com: DNS
- DNS query created, encapsulated in UDP, encapsulated in IP, encapsulated in Eth. To send frame to router, need MAC address of router interface: ARP
- ARP query broadcast, received by router, which replies with ARP reply giving MAC address of router interface
- client now knows MAC address of first hop router, so can now send frame containing DNS query



- IP datagram containing DNS query forwarded via LAN switch from client to 1st hop router
- IP datagram forwarded from campus network into comcast network, routed (tables created by RIP, OSPF, IS-IS and/or BGP routing protocols) to DNS server
- demux' ed to DNS server
- DNS server replies to client with IP address of www.google.com

A day in the life...TCP connection carrying HTTP



A day in the life... HTTP request/reply



When student attaches laptop to campus network, requests/receives www.google.com, what separate protocols the laptop might use?

- A. DNS, HTTP
- B. UDP, DNS, TCP, HTTP
- C. DHCP, ARP, DNS, HTTP
- D. DHCP, ARP, DNS, TCP, HTTP
- E. UDP, DHCP, ARP, DNS, TCP, HTTP

Next lecture

• Wireless and Mobile Networks Readings Chapter 6