Networks

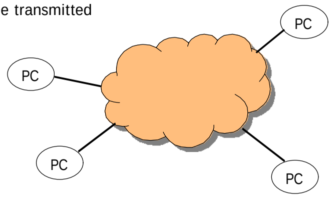
Arvind Krishnamurthy
Spring 2004

Centralized vs. Distributed Systems

- Distributed systems: physically separate computers working together
- Why do we need distributed systems?
  - Independent computers for individual users
  - Cheaper & easier to build lots of simple computers
  - Easier to add power incrementally
- Promise of distributed systems:
  - Higher availability of services
  - Better reliability of data
  - More security
- Reality is different!
- Challenge: coordination is more difficult when performed over a network

Definitions

- Network: physical connection to allow two computers to communicate
- Network interface: computer's interface to the network
- Packet: unit of transfer, sequence of bits carried over network
  - Network carries packet from one CPU to another
  - Destination gets interrupts when packet arrives
- Protocol: agreement between two parties as to how information is to be transmitted



Networking hardware

- Networking history:
  - Point to point links (over phone lines & twisted cables)
    - Departure from "circuit switching"
    - Introduced "packet switching" (Kleinrock, NPL, RAND)
    - Initial ARPAnet consisted of four computers connected to each other (UCSB, SRI, UCLA, Univ. of Utah)
    - Each computer was connected to a smaller computer (Interface Message Processor)
    - Signal then converted to analog and sent over phone lines
  - Broadcast networks
  - Network of broadcast networks
  - Point-to-point networks

Broadcast networks

- Shared communication medium
  - Inspired by Aloha network ('70s): packet radio in Hawaiian islands developed by Norm Abramson
    - Used radio for transmission
    - Communication channel always open, and every computer can receive packets from every other computer
  - Example: wire, all hosts listen to wire
    - Inside a computer: bus
    - Ethernet (10 Mbits/sec)
- Issues:
  - Delivery
  - Arbitration

Delivery

- When you broadcast a packet, how does receiver know who it is for?
- Put header in front of packet: Dest || Packet
- Header would contain unique machine # of target
  - Everyone gets packet, discards if not the target
  - Ethernet: check is done in h/w, no interrupt if not for you
- Huge security problem:
  - Anybody can listen in
  - Break into a machine, become root, reprogram it to pick up every packet, can see passwords go over the network (TCPDUMP program snoops on network packets)

Arbitration

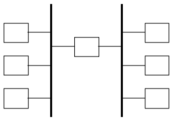
- How do machines arbitrate use of shared medium?
- Arbitration: blind broadcast, with checksum at end of packet; if received ok, send back an acknowledgment
- Need checksum in case packet gets garbled
- Sender waits for a while, if no ack, retransmits
- Problem:
  - If load increases
    - → more collisions
    - → less gets thru'
    - → more re-send
    - → more load

Ethernet

- First practical local area network (Xerox Parc)
  - Initial design by Bob Metcalfe
  - Wire broadcast: everyone taps into single wire
  - New arbitration:
    - carrier sense (don't send unless idle)
    - collision detect (sender checks if packet is trampled)
  - How long before retransmission?
    - If collision, pick bigger and bigger wait times
    - Randomized

Internet

- What happens if you need more b/w than a single ethernet?
- SUN has > 10000 workstations
- Buy two ethernets; how do two machines on different networks talk to each other?



- Put machine that straddles both networks (router, gateway, bridge, repeater)
  - Basically a switch – machine watches packets on both sides
  - If a packet is for machine on other side, then copy
- Internet: generalization of this
  - Challenge: routing packets

Routing

- How do packets get to their destination?
  - Simple if there is a single machine that straddles all networks
  - Does not scale!
- If packet has to go several hops before it gets to destination, and router straddles several networks:
  - How do routers know how to forward packets?
  - Some definitions:
    - Name
    - Address: phone number, postal, IP addresses
    - Route: how do we get there
- Internet solution: routing tables
  - Look at packet header
  - Do table lookup (destination LAN → output link to follow)
  - How do you setup the table?

Routing Table for Small Networks

- Link State Protocol is commonly used
- Every node:
  - Sends information about its links (latency, bandwidth, etc.)
  - To every other node in the system
- Each node can then build a graph that describes the state of the entire system
- It can then calculate shortest paths (or other interesting paths) using local information
- Routes packets along the paths computed by the local algorithm
- Propagate link state updates along "best paths" → cyclic dependency here; compute paths using link state information, propagate link state using paths computed by link state information

Routing table for Internet

- Internet has no centralized state
- No single machine knows entire topology, and topology is constantly changing
- Instead:
  - Routing table has "cost" – number of hops to destination (in practice, consider how heavily used it is, bandwidth)
  - Neighbors periodically exchange routing tables
  - If neighbor has cheaper route, use that one
  - Initially, routers don't know about any destination node (other than neighbors)