CSEP561 – Internetworking

David Wetherall
djw@cs.washington.edu
Internetworking

- Focus:
  - Joining multiple, different networks into one larger network

- Heterogeneity
- IPv4 and IPv6 formats
- Path MTU discovery
- Error reporting with ICMP
- Other glue: DHCP, ARP
Forwarding Packets in a Network

- We can do it:
  - Routers compute routes (DV or LS)
  - Each router builds its forwarding table
  - Packets carry addresses; routers look them up
Internetworking

• Issues of heterogeneity and scale (→ routing problem)
Heterogeneity

• How might networks differ?
  – Service model (datagrams vs. connections)
  – Quality of service / priorities
  – Security
  – Maximum packet length

• How can we deal with these differences?
  – Service model: not easily
  – QOS: we’re screwed, or we overprovision
  – Security: add what we can end-to-end
  – Packet lengths: path MTU discovery
Service Models

- Datagram (connectionless, best-effort) delivery: postal service
  - Network can’t guarantee delivery of the packet
  - Each packet from a host is routed independently
  - Example: IP, switched Ethernet

- Virtual circuit (connection-oriented) delivery: telephone
  - Signaling: connection establishment, data transfer, teardown
  - All packets from a host are routed the same way (router state)
  - Example: MPLS, ATM, Frame Relay, X.25

- Q: How do we combine them? A: Not easily!
IPv4 Packet Format

- Version is 4; addresses are 32 bit addresses
- Header length in 32 bit words, limits size of options
- DiffServ field used to be TOS
IPv6 packet format

- Version is 6
- 128 bit addresses
- Fields renamed & streamlined
- FlowLabel added
- Checksum gone
Fragmentation

• Sending small packets is wasteful, but don’t know a priori how large a packet will fit through the network

• One solution: network fragmentation
  – Network breaks large packets that are too large
  – Reassemble at destination (Why?)
  – Turns out to be bad (Why?)

• Better solution: discover largest packet for each a path (the “path MTU”) and tell the sender. (Downsides?)
Path MTU Discovery

- Path MTU is the smallest MTU along path
  - Packets less than this size don’t get fragmented
Path MTU Discovery

- Hosts send packets, routers return error to host if packet too large
  - Use DF (Don’t Fragment) header flag
  - Hosts discover limits, can fragment at source
  - Reassembly at destination as before

- Even better:
  - Host IP tells higher layer the right MTU to use; no fragmentation
  - At the cost of a layering violation
ICMP

- What happens when things go wrong?
  - Need a way to test/debug a large, widely distributed system

- ICMP = Internet Control Message Protocol (RFC792)
  - Companion to IP – required functionality

- Used for error and information reporting:
  - Errors that occur during IP forwarding
  - Queries about the status of the network
ICMP Generation

Error during forwarding!

source

IP packet

dest

ICMP IP packet
Common ICMP Messages

- **Destination unreachable**
  - “Destination” can be host, network, port or protocol
- **Redirect**
  - To shortcut circuitous routing
- **TTL Expired**
  - Used by the “traceroute” program
- **Echo request/reply**
  - Used by the “ping” program

- ICMP messages include portion of IP packet that triggered the error (if applicable) in their payload
Glue: Dynamic Host Configuration Protocol (DHCP)

• Q: How does a host get an IP address?
• A: DHCP, designed in 1993

• DHCP is widespread for the dynamic assignment of IP addresses, e.g., CSE, your cable company, …
• Host broadcasts request; DHCP server responds with IP
• Extensions:
  – Supports temporary allocation (“leases”) of IP addresses
  – DHCP client can acquire all IP configuration parameters
DHCP Interaction (simplified)

 DHCP Request:  
00:a0:24:71:e4:44  
Sent to 255.255.255.255

 DHCP Response:  
IP address: 128.143.137.144  
Default gateway: 128.143.137.1  
Netmask: 255.255.0.0
Address Resolution Protocol (ARP)

- Problem: We want to send to an IP address, but how do we find the right link layer address to put in the frame?
- Solution: ARP maps next IP to local Ethernet address

** ARP Request:**
Who has 192.168.1.1?

** ARP Reply:**
Ethernet address:
00:FE:2B:54:39:A1