CSEP 561 – Internetworking

David Wetherall
djw@cs.washington.edu
Internetworking

• Focus:
  – Joining multiple, different networks into one larger network

• Forwarding models
• Heterogeneity
• IPv4 and IPv6 formats
• Link layer glue: DHCP, ARP
• ICMP for error reporting
• Fragmentation and Path MTU discovery
Store-and-Forward Packet Switching

Hosts send packets into the network; packets are forwarded by routers
Connectionless Service – Datagrams

Packet is forwarded using destination address inside it
• Different packets may take different paths

![Diagram of network with routers and ISPs]

A's table (initially):

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>B</td>
<td>B</td>
<td>C</td>
<td>C</td>
<td>C</td>
</tr>
</tbody>
</table>

A's table (later):

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>B</td>
<td>B</td>
<td>C</td>
<td>C</td>
<td>C</td>
</tr>
</tbody>
</table>

C's Table:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>B</td>
<td>A</td>
<td>C</td>
<td>C</td>
</tr>
</tbody>
</table>

E's Table:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>B</td>
<td>D</td>
<td>C</td>
<td>C</td>
<td>C</td>
</tr>
</tbody>
</table>

Dest. Line: A's Table (initially)
Connection-Oriented – Virtual Circuits

Packet is forwarded along a virtual circuit using tag inside it
- Virtual circuit (VC) is set up ahead of time
Comparison of Virtual-Circuits & Datagrams

<table>
<thead>
<tr>
<th>Issue</th>
<th>Datagram network</th>
<th>Virtual-circuit network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circuit setup</td>
<td>Not needed</td>
<td>Required</td>
</tr>
<tr>
<td>Addressing</td>
<td>Each packet contains the full source and destination address</td>
<td>Each packet contains a short VC number</td>
</tr>
<tr>
<td>State information</td>
<td>Routers do not hold state information about connections</td>
<td>Each VC requires router table space per connection</td>
</tr>
<tr>
<td>Routing</td>
<td>Each packet is routed independently</td>
<td>Route chosen when VC is set up; all packets follow it</td>
</tr>
<tr>
<td>Effect of router failures</td>
<td>None, except for packets lost during the crash</td>
<td>All VCs that passed through the failed router are terminated</td>
</tr>
<tr>
<td>Quality of service</td>
<td>Difficult</td>
<td>Easy if enough resources can be allocated in advance for each VC</td>
</tr>
<tr>
<td>Congestion control</td>
<td>Difficult</td>
<td>Easy if enough resources can be allocated in advance for each VC</td>
</tr>
</tbody>
</table>
Internetworking

• Issues of heterogeneity and scale
Scale (mostly routing issue)

• Use hierarchy by routing to:
  – IP prefixes = groups of related addresses
  – Autonomous systems = entire (ISP) networks

• We will explore as part of inter-domain routing

• Route scaling methods also affect forwarding:
  – Longest Matching Prefix (LMP) rule
  – Network Address Translation (NAT) boxes
Heterogeneity (mostly forwarding issue)

• 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
Sample issue: differing 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!
Internetworking with IP

• IP enables communication over different networks
  – Defines function of lowest common denominator
  – Includes necessary glue

• Allows innovation of link technologies and apps

Protocols with IP as the “narrow waist”
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
IP Version 6 (1)

Major upgrade in the 1990s due to impending address exhaustion, with various other goals:
- Support billions of hosts
- Reduce routing table size
- Simplify protocol
- Better security
- Attention to type of service
- Aid multicasting
- Roaming host without changing address
- Allow future protocol evolution
- Permit coexistence of old, new protocols, …

Deployment has been slow & painful, but may pick up pace now that addresses are all but exhausted
IPv6 protocol header has much longer addresses (128 vs. 32 bits) and is simpler to process.
IP Version 6 (3)

Additions:
- Longer addresses (128 bits)
- Flow label is added (grouping hint to network)

Simplifications:
- Header checksum is gone
- Weird stuff moved to optional extensions (e.g., fragments and identification)
- (Upper) Protocol combined with Next Header
- Header length is now fixed
- TTL renamed “Hop Limit”
IP Version 6 (3)

IPv6 extension headers handles other functionality

<table>
<thead>
<tr>
<th>Extension header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hop-by-hop options</td>
<td>Miscellaneous information for routers</td>
</tr>
<tr>
<td>Destination options</td>
<td>Additional information for the destination</td>
</tr>
<tr>
<td>Routing</td>
<td>Loose list of routers to visit</td>
</tr>
<tr>
<td>Fragmentation</td>
<td>Management of datagram fragments</td>
</tr>
<tr>
<td>Authentication</td>
<td>Verification of the sender’s identity</td>
</tr>
<tr>
<td>Encrypted security payload</td>
<td>Information about the encrypted contents</td>
</tr>
</tbody>
</table>
IPv4 Exhaustion

- IANA allocated last /8s to RIRs in 2/2011
- APNIC ran out in 4/2011, other RIRs following soon

Uh oh, no more IPv4 addresses!
IPv6 Deployment

Time for some growth!
IPv6 Deployment Strategies

Dual stacks
- Host reachable via IPv4 and IPv6 connectivity
- DNS can return both IPv6 and IPv4 addresses

Tunneling, e.g., join IPv6 islands over IPv4 core
- Want to configure tunnels automatically
- Many schemes, e.g., “6to4”, “6rd”, …
Link layer glue

• Binds network (IP) and link (Ethernet) layers together

• In the Internet:
  – Host getting an IP address: DHCP
  – Host getting IP address of the default gateway: DHCP
  – Host getting local Ethernet address given an IP host: ARP
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
  – Example of a local discovery protocol

• Extensions:
  – Supports temporary allocation (“leases”) of IP addresses
  – DHCP client can acquire all IP configuration parameters
DHCP protocol stack

- Runs on top of UDP/IP
  - Port 67 and 68

- Ethernet / IP destination of broadcast (ff:ff:ff:ff:ff:ff and 255.255.255.255) are used for discovery / coordination
  - Everyone on LAN sees them

- Host (client) uses IP address of 0.0.0.0 meaning “this host on this network” until DHCP server gives it another IP
DHCP operation (DORA)

- Discovers finds DHCP server, requests address and more
- Offer advertises lease to client
- Request asks for a given lease
- Ack grants it and more info

- Client then switches to new IP
- Client can send a release message to relinquish IP
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
IP/Ethernet Addressing Example

ARP (Address Resolution Protocol) lets nodes find target Ethernet addresses [pink] from their IP addresses.
ARP (Address Resolution Protocol) lets nodes find target Ethernet addresses [pink] from their IP addresses

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Host 1 to 2, on CS net</td>
<td>IP1</td>
<td>E1</td>
<td>IP2</td>
<td>E2</td>
</tr>
<tr>
<td>Host 1 to 4, on CS net</td>
<td>IP1</td>
<td>E1</td>
<td>IP4</td>
<td>E3</td>
</tr>
<tr>
<td>Host 1 to 4, on EE net</td>
<td>IP1</td>
<td>E4</td>
<td>IP4</td>
<td>E6</td>
</tr>
</tbody>
</table>
IP works with the help of several control protocols:

- **ICMP** is a companion to IP that returns error info
  - Required, and used in many ways, e.g., for traceroute
- **ARP** finds Ethernet address of a local IP address
  - Glue that is needed to send any IP packets
  - Host queries an address and the owner replies
- **DHCP** assigns a local IP address to a host
  - Gets host started by automatically configuring it
  - Host sends request to server, which grants a lease
ICMP Generation

Error during forwarding!
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
Fragmentation Issue

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

• One solution: network fragmentation (IPv4 only)
  – 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 with ICMP

- 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