Lecture 12: IPv6 and ICMP

CSE 123: Computer Networks
Aaron Schulman

UCSD CSE
Today’s Lecture

- Finish up describing how addresses are used for **routing**
  - How Longest prefix match example in practice
- Dealing with the space crunch! (**IPv6**)  
- Notifying about errors in the internetwork (**ICMP**)
Route Aggregation

- Combine adjacent networks in forwarding tables
  - Helps keep forwarding table size down

Organization 0
- 200.23.16.0/23

Organization 1
- 200.23.18.0/23

Organization 2
- 200.23.20.0/23

Organization 7
- 200.23.30.0/23

Fly-By-Night-ISP

“Send me anything with addresses beginning 200.23.16.0/20”

ISPs-R-Us

“Send me anything with addresses beginning 199.31.0.0/16”

Internet
Most Specific Route

- But what if address range is not contiguous?

Organization 0
200.23.16.0/23

Organization 2
200.23.20.0/23

Organization 7
200.23.30.0/23

Organization 1
200.23.18.0/23

Fly-By-Night-ISP

"Send me anything with addresses beginning 200.23.16.0/20"

ISP's-R-Us

"Send me anything with addresses 200.23.18.0/23"

Internet (Needs to be do longest prefix match)
Forwarding example

- Packet to 10.1.1.6 is sent to your ISP
- Path is R2 – R1 – H1 – H2
Forwarding example (2)

• Packet to 10.1.1.6
• Matches 10.1.0.0/23

Forwarding table at R2

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.1</td>
<td>loopback</td>
</tr>
<tr>
<td>Default or 0/0</td>
<td>10.1.16.1</td>
</tr>
<tr>
<td>10.1.8.0/24</td>
<td>interface1</td>
</tr>
<tr>
<td>10.1.2.0/23</td>
<td>interface2</td>
</tr>
<tr>
<td><strong>10.1.0.0/23</strong></td>
<td><strong>10.1.2.2</strong></td>
</tr>
<tr>
<td>10.1.16.0/24</td>
<td>interface3</td>
</tr>
</tbody>
</table>
Forwarding example (3)

- Packet to 10.1.1.6
- Matches 10.1.1.4/30
- Longest prefix match

Routing table at R1

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.1</td>
<td>loopback</td>
</tr>
<tr>
<td>Default or 0/0</td>
<td>10.1.2.1</td>
</tr>
<tr>
<td>10.1.0.0/24</td>
<td>interface1</td>
</tr>
<tr>
<td><strong>10.1.1.0/24</strong></td>
<td>interface2</td>
</tr>
<tr>
<td>10.1.2.0/23</td>
<td>interface3</td>
</tr>
<tr>
<td><strong>10.1.1.4/30</strong></td>
<td>10.1.1.101</td>
</tr>
</tbody>
</table>

CSE 123 – Lecture 12: IPv6 and ICMP
Forwarding example (4)

• Packet to 10.1.1.6
• Direct route
  • Longest prefix match

Routing table at R3

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next Hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.1</td>
<td>loopback</td>
</tr>
<tr>
<td>Default or 0/0</td>
<td>10.1.1.1</td>
</tr>
<tr>
<td>10.1.1.0/24</td>
<td>interface1</td>
</tr>
<tr>
<td>10.1.1.4/30</td>
<td>interface2</td>
</tr>
</tbody>
</table>
The space crunch...

- Still running out of IP addresses... what to do?
- Two solutions
  - Network Address Translation – multiple multiple hosts on a single IP address (future class)
  - Get bigger addresses -> IPv6
- IPv6: 128bit addresses... we won’t run out
  - Often 64bit routing prefix, 64bit host id!

An IPv6 address (in hexadecimal)

\[2001:0DB8:AC10:FE01:0000:0000:0000:0000\]

Zeroes can be omitted
Orgs have fun with v6

- What provider is using this address?
  - 2a03:2880:f003:0c07:face:b00c:0000:0002

- Fun opportunities to include vanity HEX in hosts
IPv6 Addresses

- Colon-Hex notation
  - 8 groups of four HEX digits separated by colons, e.g.
    » FEDC:0000:0000:0065:4321:0000:DEAD:BEEF
  - Can drop leading zeros:
  - Can even skip first sequence of all zeros w/ ::
    » FEDC::65:4321:0000:DEAD:BEEF
  - Every IPv4 address is a IPv6 address:
    » E.g., ::222.173.190.239 (prepended w/zeros)

- Network names expressed as prefix/length:
  - Still can do internal subnetting
    » FEDC::65:43/50
Address Types

- Each interface has multiple different addresses
  - Link local, prefixed with FE80::/10 (1111 1110 10)
    » Used only for communication between adjacent IPv6 devices
    » Packets are NOT forwarded by routers
    » Automatically assigned upon boot
  - Unique local, prefixed with FC00::/7 (1111 110 )
    » Used only internal to one network
    » Not routable on the global Internet
  - Global
    » Like an IPv4 address
### IPv6 vs IPv4 header

#### IPv6 header
- **Ver.**
- **Traffic Class**
- **Flow Label**
- **Payload Length**
- **Source Address**
- **Destination Address**

#### IPv4 header
- **Ver.**
- **Hdr Len**
- **Type of Service**
- **Total Length**
- **Identification**
- **Flag**
- **Fragment Offset**
- **Time to Live**
- **Protocol**
- **Header Checksum**
- **Source Address**
- **Destination Address**
- **Options...**

- Gray bits are unique to each header
- **Changes**
  - Eliminate fragmentation-related fields
  - Eliminate header checksum
  - Added flow label
  - Quadruple size of addresses
  - IPv6 header (40 bytes) vs IPv4 (20 bytes)
  - Hop limit == TTL
Extension Headers

- Effectively a linked list of headers
  - The “next header” field is the pointer

- Two different types
  - **Destination**, intended for the IP end point. E.g.,
    - 44: Fragmentation Header (it’s baaack!)
    - 43: Routing header (dictates how to route the packet)

  - **Hop-by-hop**, processed by each node on the path
    - Each node takes a look at it and can modify it
IPv6 Transition is slow

- Need to support both protocols at the same time
  - Complicated… if a destination has both an IPv4 and IPv6 address, which to use?
- Less need in developed world -> slower adoption
- That said
  - All major operating systems now support IPv6
  - All major router vendors
  - US Mobile carriers (e.g., Tmobile, Verizon, etc)
  - Offered as option by many US wired ISPs
- As of this year IPv6 is dominating mobile carrier traffic
Dealing with errors: 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 Error Message 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
    - traceroute traces packet routes through Internet

- **Echo request/reply**
  - Used by the “ping” program
    - ping just tests for host liveness

- **ICMP messages include portion of IP packet that triggered the error (if applicable) in their payload**
ICMP Restrictions

- The generation of error messages is limited to avoid cascades … error causes error that causes error…

- Don’t generate ICMP error in response to:
  - An ICMP error
  - Broadcast/multicast messages (link or IP level)
  - IP header that is corrupt or has bogus source address
  - Fragments, except the first

- ICMP messages are often rate-limited too
  - Don’t waste valuable bandwidth sending tons of ICMP messages
For Next Time

- Homework 2 due Tuesday
- Midterm not next Monday but the following MONDAY
  - Online on GradeScope:
- Read 9.3, 3.3 (DHCP, ARP)