Announcements

- **Reading**
  - Chapter 6 (6.1 & 6.2)

- **Project #3**
  - Is on the web

- **Midterm #1**
  - Next Tuesday Oct. 9
  - Cover material through Tuesday’s lecture
Border Gateway Protocol (BGP)

- **Used to route between AS’s**
  - concerned with politics and turf battles
  - supports specific policies
    - don’t send my packets of network X
    - don’t send packets through me

- **Two types of nodes**
  - stub networks (one connection to BGP)
  - multi-connected networks (more than one connection)
    - might also be transit networks (carry traffic for others)

- **Uses Distance Vector**
  - but includes complete path in table and sent to neighbors
  - uses “scoring” function to select among possible routes
Fragmentation

- Sometimes need to split packets into smaller units
  - limits of the hardware being used
  - operating system buffer constraints
  - protocol limits (max permitted packet is x bytes)
  - reduce channel occupancy (head of link blocking)
- Fragmentation
  - where to split it into smaller packets
    - source (requires end-to-end information on max size)
    - when it reaches boundary
  - how to represent split packets
    - need to encode fragment offset
- Reassembly
  - where to re-combine packets
    - destination (may result in poor performance)
    - at the gateway to the subnet that supports the full size
The IP Protocol

- **IP Header**
  - source, destination address, total length
  - version, ihl (header length in 32-bit words), ttl, protocol
  - fragmentation support: identification, df, mf, frag. offset

- **Options**
  - variable length
  - defined options
    - loose source routing
    - timestamp
    - record path

<table>
<thead>
<tr>
<th>Ver</th>
<th>IHL</th>
<th>Service</th>
<th>Total Length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identification</td>
<td>DF</td>
<td>MF</td>
<td>Fragment Offset</td>
</tr>
<tr>
<td>TTL</td>
<td>Protocol</td>
<td>Header Checksum</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Source Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Destination Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0 Or More Options</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

32 bits
Fragmentation in IP

- ID of all fragments is the same
- Fragment offset
  - expressed in fragment units (8 bytes)
  - Supports a maximum of 65536 byte packets
- DF – do not fragment
  - Must remain as a full unit
- MF – more fragment
  - Indicates that there is more data in a fragment after this one
IPv6 Packet Format

<table>
<thead>
<tr>
<th>Ver</th>
<th>Priority</th>
<th>Flow</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Payload Length
- Next Header
- Hop Limit

- Source Address (16 bytes)
- Destination Address (16 bytes)

32 bits
IPv6 Addresses

- Each address is 16 bytes long
  - Divided into several ranges
    - 0000 0000 - Reserved (including IPv4) 1/256
    - 010 - provider based addresses (1/8)
    - 100 - geographic addresses (1/8)
    - 1111 1110 10 link local (1/1024)
    - 1111 1110 11 site local use (1/1024)
  - Notation
    - Hex in groups of 16 bits
      - fec0:0000:0000:0000:0000:0000:0000:0001
    - Can use :: (once) to indicate string of zeros
      - fec0::0001 or fec0::1
Project Architecture

Unix Process

Timer Thread

Routing Thread

Recv Thread

Tracy, 5001

Unix Process

Timer Thread

Routing Thread

Recv Thread

Tracy, 5002

Unix Process

Timer Thread

Routing Thread

Recv Thread

Tracy, 5000
Project Components

- **Shortest path computation**
  - Use Diksta’s algorithm
- **Topology discovery**
  - Send hello packets around
- **Timer thread**
  - Extends project #2
- **Each thread will have an event driven main loop**