Network Layer: Overview

- Network layer functions
- IP
- Routing and forwarding

Network layer functions

- Transport packet from sending to receiving hosts
- Network layer protocols in every host, router

Three important functions:

- **Path determination**: route taken by packets from source to dest. *Routing algorithms*
- **Switching**: move packets from router's input to appropriate router output
IP Addressing

- **IP address**: 32-bit identifier for host, router *interface*
- **Interface**: connection between host, router and physical link
  - Routers typically have multiple interfaces
  - Host may have multiple interfaces
  - IP addresses associated with interface, not host, router

223.1.1.1 = 11011111 00000001 00000001 00000001

223.1.1.1 = 11011111 00000000 00000001 00000001

IP Addressing (2)

- **IP address**:
  - Network part (high order bits)
  - Host part (low order bits)
- **What’s a network?**
  (from IP address perspective)
  - Device interfaces with same network part of IP address
  - Can physically reach each other without intervening router

Network consisting of 3 IP networks
(for IP addresses starting with 223, first 24 bits are network address)
**IP Addressing (3)**

How to find the networks?
- Detach each interface from router, host
- Create “islands” of isolated networks

**Interconnected system consisting of six networks**

**IP Networks: Subnets**

- Sub divide address space
  - Network part
  - Host address
- Address format: `a.b.c.d/x`, where `x` is # bits in subnet portion of address

```
11001000  00010111 00010000 00000000
```

`200.23.16.0/24`
**Address Management**

- **Problem:** we are running out of networks
- **Solution (a):**
  - **subnetting:** e.g., Class B Host field (16 bits) is subdivided into <subnet;host> fields
- **Solution (b):**
  - **CIDR** (Classless Inter Domain Routing)
CIDR

- Classless InterDomain Routing
- Class A is too large, Class C is too small
- Everyone has a Class B address!!!

- Solution: sites are given contiguous blocks of class-C addresses (256 addresses each) and a mask or parts of former class A/B networks.

CIDR: Classless InterDomain Routing

- Subnet portion of address of arbitrary length
- Address format: a.b.c.d/x, where x is # bits in subnet portion of address

```
11001000 00010111 00010000 00000000
subnet part
11001000 00010111 00010000 00000000
host part
200.23.16.0/23
```
IP addresses: how to get one?

Q: How does host get IP address?
- Hard-coded by system admin in a file
  - Windows: control-panel → network → configuration → tcp/ip → properties
  - UNIX: /etc/rc.config
- DHCP: Dynamic Host Configuration Protocol:
  - dynamically get address from as server
  - “plug-and-play”
- IP / Subnets allocated by provider (RIPE/ARIN/…)

Hierarchical address structure

- Recall: CIDR
  - 128.119.48.12/18 = 10000000 01110111 00110000 00001100
  - High order bits form the prefix
  - Once inside the network, can subnet: divide remaining bits
  - Subnet example:
    - 129.128.0.0/10
    - 129.160.0.0/12
    - 129.176.0.0/14
    - 129.188.0.0/14

- Forwarding decision: longest prefix match

Note: Picture shows prefix masks, not interface addrs!
**Forwarding vs. Routing**

- **Forwarding**: the process of moving packets from input to output
  - The forwarding table
  - Information in the packet

- **Routing**: process by which the forwarding table is built and maintained
  - One or more routing protocols
  - Procedures (algorithms) to convert routing info to forwarding table.
  
  (Much more later …)

**Forwarding with CIDR**

- Packet should be sent towards the interface with the **longest matching prefix**

```
  1000 1101 0110
  1000 1110
  1000 1100 1101
  1000 1101 0011
```

Advertised addresses
**Lookup (longest prefix match):**

- Forwarding table:
  \(<\text{Network}/\text{mask}> <\text{next-hop}>\)

- IP Packets: destination IP address
  - Find next-hop

- Example:

<table>
<thead>
<tr>
<th>Forwarding table</th>
<th>Packets</th>
</tr>
</thead>
<tbody>
<tr>
<td>134.96.252.0/24</td>
<td>A 134.96.252.200</td>
</tr>
<tr>
<td>134.96.0.0/16</td>
<td>C 134.96.254.2</td>
</tr>
<tr>
<td>134.96.240.0/20</td>
<td>B 134.96.239.200</td>
</tr>
<tr>
<td>134.96.252.192/28</td>
<td>B 134.97.239.200</td>
</tr>
<tr>
<td>134.96.252.128/28</td>
<td>A 134.96.252.191</td>
</tr>
</tbody>
</table>

---

**IPv4 datagram format**

- IP protocol version number
- Header length (bytes)
- “Type” of data
- Max number remaining hops (decremented at each router)
- Upper layer protocol to deliver payload to

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protocol</td>
<td>8-bit number representing the IP protocol</td>
</tr>
<tr>
<td>Source address</td>
<td>32-bit IP address of the source node</td>
</tr>
<tr>
<td>Destination</td>
<td>32-bit IP address of the destination node</td>
</tr>
<tr>
<td>Time to live</td>
<td>16-bit field indicating the time remaining before the datagram expires</td>
</tr>
<tr>
<td>Identification</td>
<td>16-bit identifier used for fragmentation</td>
</tr>
<tr>
<td>Flag</td>
<td>3-bit field indicating fragmentation-related options</td>
</tr>
<tr>
<td>Fragment offset</td>
<td>13-bit field indicating the offset of the fragment within the datagram</td>
</tr>
<tr>
<td>Fragment length</td>
<td>8-bit field indicating the length of the fragment</td>
</tr>
<tr>
<td>Data (variable)</td>
<td>Variable length field representing the payload data</td>
</tr>
<tr>
<td>Options (if any)</td>
<td>Additional fields for protocol-specific information</td>
</tr>
</tbody>
</table>

- Total datagram length (bytes)
- For fragmentation/reassembly
- E.g. timestamp, record route taken, specify list of routers to visit.

---
ICMP: Internet Control Message Protocol

- Used by hosts, routers, gateways to communication network-level information
  - Error reporting: unreachable host, network, port, protocol
  - Echo request/reply (used by ping)
- Network-layer “above” IP:
  - ICMP msgs carried in IP datagrams
- ICMP message: type, code plus first 8 bytes of IP datagram causing error

<table>
<thead>
<tr>
<th>Type</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>echo reply (ping)</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>dest network unreachable</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>dest host unreachable</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>dest protocol unreachable</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>dest port unreachable</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>dest network unknown</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>dest host unknown</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>source quench (congestion control – not used)</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>echo request (ping)</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>route advertisement</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>router discovery</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>TTL expired</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>bad IP header</td>
</tr>
</tbody>
</table>

Network Layer: Summary

- Network layer functions
- IP
- Routing and forwarding