CSE 461: Computer networks

Spring 2021

Ratul Mahajan
Network Layer (IP)
Recall the protocol stack

<table>
<thead>
<tr>
<th>Layer</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>Programs that use network service</td>
</tr>
<tr>
<td>Transport</td>
<td>Provides end-to-end data delivery</td>
</tr>
<tr>
<td>Network</td>
<td>Send packets over multiple networks</td>
</tr>
<tr>
<td>Link</td>
<td>Send frames over one or more links</td>
</tr>
<tr>
<td>Physical</td>
<td>Send bits using signals</td>
</tr>
</tbody>
</table>
Network Layer

- How to get packets from source to destination

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td></td>
</tr>
<tr>
<td>Transport</td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td></td>
</tr>
<tr>
<td>Link</td>
<td></td>
</tr>
<tr>
<td>Physical</td>
<td></td>
</tr>
</tbody>
</table>
Why do we need a Network layer?

• Cannot afford to directly connect everyone
  • Cost and link layer diversity
Why do we need a Network layer? (2)

• Cannot broadcast all packets at global scale
Why do we need a Network layer? (3)

• Internetworking
  • Need to connect different link layer networks
• Addressing
  • Need a globally unique way to “address” hosts
• Routing and forwarding
  • Need to find and traverse paths between hosts
Routing versus Forwarding

• **Routing:** deciding the direction to send traffic

• **Forwarding:** sending a packet on its way
Network Service Models
Network service models

• What kind of service does the Network layer provide to the Transport layer?
  • How is it implemented at routers?
Two Network Service Models

• Datagrams, or connectionless service
  • Like postal letters
  • (IP as an example)

• Virtual circuits, or connection-oriented service
  • Like a telephone call
Datagram Model

• Packets contain a destination address; each router uses it to forward packets, maybe on different paths
Datagram Model (2)

- Each router has a forwarding table keyed by address
  - Gives next hop for each destination address; may change

```
A's table (initially)     A's table (later)    C's Table          E's Table
A | B | B
B | C | C
C | D | B
D | E | B
E | F | C
F |  |  
```

```
A | A | A
B | B | E
C | C | D
D | D | E
E | F | F
F |  |  
```
IP (Internet Protocol)

- Network layer of the Internet, uses datagrams (next)
- IPv4 carries 32 bit addresses on each packet
Virtual Circuit Model

• Three phases:
  1. Connection establishment, circuit is set up
     • Path is chosen, circuit information stored in routers
  2. Data transfer, circuit is used
     • Packets are forwarded along the path
  3. Connection teardown, circuit is deleted
     • Circuit information is removed from routers

• Just like a telephone circuit, but virtual in that no bandwidth need be reserved; statistical sharing of links
Virtual Circuits

- Packets contain a short label to identify the circuit
- Labels don’t have global meaning, only unique for a link
Virtual Circuits (2)

• Each router has a forwarding table keyed by circuit
  • Gives output line and next label to place on packet
MPLS (Multi-Protocol Label Switching, §5.6.5)

- A virtual-circuit like technology widely used by ISPs
  - ISP sets up circuits inside their backbone ahead of time
  - ISP adds MPLS label to IP packet at ingress, undo at egress
Datagrams vs Virtual Circuits

- Complementary strengths

<table>
<thead>
<tr>
<th>Issue</th>
<th>Datagrams</th>
<th>Virtual Circuits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup phase</td>
<td>Not needed</td>
<td>Required</td>
</tr>
<tr>
<td>Router state</td>
<td>Per destination</td>
<td>Per connection</td>
</tr>
<tr>
<td>Addresses</td>
<td>Packet carries full address</td>
<td>Packet carries short label</td>
</tr>
<tr>
<td>Forwarding</td>
<td>Per packet</td>
<td>Per circuit</td>
</tr>
<tr>
<td>Failures</td>
<td>Easier to mask</td>
<td>Difficult to mask</td>
</tr>
<tr>
<td>Quality of service</td>
<td>Difficult to add</td>
<td>Easier to add</td>
</tr>
</tbody>
</table>
Internetworking (IP)
Topic

• How do we connect different networks together?
  • This is called internetworking
  • We’ll look at how IP does it
How Networks May Differ

• Lot of ways:
  • Service model (datagrams, VCs)
  • Addressing (what kind)
  • QOS (priorities, no priorities)
  • Packet sizes
  • Security (whether encrypted)

• Internetworking hides the differences with a common protocol. (Uh oh.)
Connecting Datagram and VC networks

• An example to show that it’s not so easy
  • Need to map destination address to a VC and vice-versa
  • A bit of a “road bump”, e.g., might have to set up a VC
Internetworking – Cerf and Kahn

• Pioneers: Cerf and Kahn
  • “Fathers of the Internet”
  • In 1974, later led to TCP/IP

• Tackled the problems of interconnecting networks
  • Instead of mandating a single technology
Internet Reference Model

• Internet Protocol (IP) is the “narrow waist”
  • Supports many different links below and apps above

1. Link
   - Ethernet
   - Cable
   - DSL
   - 802.11

2. Internet
   - IP

3. Transport
   - TCP
   - UDP

4. Application
   - SMTP
   - HTTP
   - RTP
   - DNS
IP as a Lowest Common Denominator

• Suppose only some networks support QOS or security etc.
  • Difficult for internetwork to support
• Pushes IP to be a “lowest common denominator”
  • Asks little of lower-layer networks
  • Gives little as a higher layer service
IPv4 (Internet Protocol)

• Various fields to meet straightforward needs
  • Version, Header (IHL), Total length, Protocol, and Header Checksum
IPv4 (2)

- Some fields to handle packet size differences (later)
  - Identification, Fragment offset, Fragment control bits

![IPv4 format diagram](image-url)
IPv4 (3)

- Other fields to meet other needs (later, later)
  - Differentiated Services, Time to live (TTL)

<table>
<thead>
<tr>
<th>Version</th>
<th>IHL</th>
<th>Differentiated Services</th>
<th>Total length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>D M F</td>
<td>Fragment offset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Time to live</td>
<td>Protocol</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Source address</td>
<td>Destination address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Options (0 or more words)</td>
<td></td>
</tr>
</tbody>
</table>

Payload (e.g., TCP segment)
IPv4 (4)

- Network layer of the Internet, uses datagrams
  - Provides a layer of addressing above link addresses (next)
IP Addresses

• IPv4 uses 32-bit addresses
  • Later we’ll see IPv6, which uses 128-bit addresses

• Written in “dotted quad” notation
  • Four 8-bit numbers separated by dots
    
    8 bits  8 bits  8 bits  8 bits
    aaaaaaaaaabbbbbbbbcddddddddd  ↔  A.B.C.D
    00010010001111100000000000000001  ↔  ??
IP Prefixes

- Addresses are allocated in blocks called **prefixes**
- Addresses in an L-bit prefix have the same top L bits
- There are $2^{32-L}$ addresses aligned on $2^{32-L}$ boundary
IP Prefixes (2)

• Written in “IP address/length” notation
  • Address is lowest address in the prefix, length is prefix bits
  • E.g., 128.13.0.0/16 is 128.13.0.0 to 128.13.255.255
  • So a /24 ("slash 24") is 256 addresses and /32 is 1 address