Network Layer
Topics

• Network service models
  • Datagrams (packets), virtual circuits

• IP (Internet Protocol)
  • Internetworking
  • Forwarding (Longest Matching Prefix)
  • Helpers: ARP and DHCP
  • Fragmentation and MTU discovery
  • Errors: ICMP (traceroute!)
    • IPv6, scaling IP to the world
    • NAT, and “middleboxes”

• Routing Algorithms
Network Address Translation (NAT)
Problem: Internet Growth

• Many billions of hosts

Source: Internet Systems Consortium (www.isc.org)
The End of New IPv4 Addresses

• Now running on leftover blocks held by the regional registries; much tighter allocation policies

Exhausted on 2/11!
Exhausted on 4/11 and 9/12!

IANA (All IPs)

ARIN (US, Canada)
APNIC (Asia Pacific)
RIPE (Europe)
LACNIC (Latin Amer.)
AfriNIC (Africa)

ISPs
Companies
Solution 1: Network Address Translation (NAT)

- Basic idea: Map many “Private” IP addresses to one “Public” IP.
- Allocate IPs for private use (192.168.x, 10.x)

I’m a NAT box too!

Internet
Layering Review

- Remember how layering is meant to work?
  - “Routers don’t look beyond the IP header.” Well ...
Middleboxes

- Sit “inside the network” but perform “more than IP” processing on packets to add new functionality
  - NAT box, Firewall / Intrusion Detection System
Middleboxes (2)

• Advantages
  • A possible rapid deployment path when no other option
  • Control over many hosts (IT)

• Disadvantages
  • Breaking layering interferes with connectivity
    • strange side effects
  • Poor vantage point for many tasks
NAT (Network Address Translation) Box

• NAT box maps an internal IP to an external IP
  • Many internal hosts connected using few external addresses
  • Middlebox that “translates addresses”

• Motivated by IP address scarcity
  • Controversial at first, now accepted
NAT (2)

• Common scenario:
  • Home computers use “private” IP addresses
  • NAT (in AP/firewall) connects home to ISP using a single external IP address
How NAT Works

• Keeps an internal/external translation table
  • Typically uses IP address + TCP port
  • This is address and port translation

<table>
<thead>
<tr>
<th>Internal IP:port</th>
<th>External IP : port</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.1.12 : 5523</td>
<td>44.25.80.3 : 1500</td>
</tr>
<tr>
<td>192.168.1.13 : 1234</td>
<td>44.25.80.3 : 1501</td>
</tr>
<tr>
<td>192.168.2.20 : 1234</td>
<td>44.25.80.3 : 1502</td>
</tr>
</tbody>
</table>

• Need ports to make mapping 1-1 since there are fewer external IPs
How NAT Works (2)

- **Internal → External:**
  - Look up and rewrite Source IP/port

<table>
<thead>
<tr>
<th>Internal source</th>
<th>Internal IP:port</th>
<th>External IP : port</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.1.12 : 5523</td>
<td>44.25.80.3 : 1500</td>
<td></td>
</tr>
</tbody>
</table>

**External destination**
IP=X, port=Y

- Source: 192.168.1.12, Port: 5523
  - Source IP: 192.168.1.12
  - Source Port: 5523
- Destination IP: 44.25.80.3
  - Port: 1500
How NAT Works (3)

- External → Internal
  - Look up and rewrite Destination IP/port

<table>
<thead>
<tr>
<th>Internal IP:port</th>
<th>External IP : port</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.1.12 : 5523</td>
<td>44.25.80.3 : 1500</td>
</tr>
</tbody>
</table>

External source
IP=X, port=Y

Src =
Dst =

Src =
Dst =
How NAT Works (4)

• Need to enter translations in the table for it to work
  • Create external name when host makes a TCP connection

<table>
<thead>
<tr>
<th>Internal source</th>
<th>Internal IP:port</th>
<th>External IP : port</th>
</tr>
</thead>
<tbody>
<tr>
<td>Src = 192.168.1.12</td>
<td>5523</td>
<td></td>
</tr>
</tbody>
</table>

External destination
IP=X, port=Y

NAT box
NAT Downsides

• Connectivity has been broken!
  • Can only send incoming packets after an outgoing connection is set up
  • Difficult to run servers or peer-to-peer apps (Skype)
• May not work when there are no connections (UDP)
• Breaks (naively designed) apps that expose their IP addresses as data (e.g., FTP)
NAT Upsides

• Relieves a lot of IP address pressure
  • Many home hosts behind NATs

• Easy to deploy
  • Just plug in the device...

• Useful functionality
  • Primitive firewall, helps with privacy

• NAT traversal approaches exist...
  • E.g., TURN, STUN, ICE, UPnP IGDP
IPv6
Not enough IPv4 Addresses? IP Version 6 to the Rescue

• Effort started by the IETF in 1994
  • Much larger addresses (128 bits)
  • Many sundry improvements

• Became an IETF standard in 1998
  • Nothing much happened for a decade
  • Hampered by deployment issues, and a lack of adoption incentives
  • Big push ~2011 as exhaustion loomed
IPv6 Traffic

IPv6 Adoption

We are continuously measuring the availability of IPv6 connectivity among Google users. The graph shows the percentage of users that access Google over IPv6.

Native: 19.48% 6to4/Teredo: 0.00% Total IPv6: 19.48% | May 17, 2018
IPv6

• Features large addresses
  • 128 bits, most of header

• New notation
  • 8 groups of 4 hex digits (16 bits)
  • Omit leading zeros, groups of zeros

Ex: 2001:0db8:0000:0000:0000:ff00:0042:8329
→ 2001:db8::ff00:42:8329
IPv6 (2)

• Lots of other changes
  • Only public addresses
    • No more NAT!
  • Streamlined header processing
    • No checksum (why’s that faster?)
  • Flow label to group of packets
  • Better fit with “advanced” features (mobility, multicasting, security)
IPv6 Stateless Autoconfiguration (SLAAC)

- Replaces DHCP (sorta...)
- Uses ICMPv6
- Process:
  - Send broadcast message
  - Get prefix from router
  - Attach MAC to router Prefix
IPv6 Transition

• The Big Problem:
  • How to deploy IPv6?
  • Fundamentally incompatible with IPv4

• Dozens of approaches proposed
  • Dual stack (speak IPv4 and IPv6)
  • Translators (convert packets)
  • Tunnels (carry IPv6 over IPv4)
Tunneling

- Native IPv6 islands connected via IPv4
- Tunnel carries IPv6 packets across IPv4 network
Tunneling (2)

• Tunnel acts as a single link across IPv4 network
Tunneling (3)

- Tunnel acts as a single link across IPv4 network
- Difficulty is to set up tunnel endpoints and routing