Outline

- The IP protocol
  - IPv4
  - IPv6
- IP in practice
  - Network address translation
  - Address resolution protocol
  - Tunnels

Altering the Addressing Model

- Original IP Model: Every host has unique IP address
- This has very attractive properties ...
  - Any host can communicate with any other host
  - Any host can act as a server
    - Just need to know host ID and port number
  - … but the system is open – complicates security
  - Any host can attack any other host
  - It is easy to forge packets
    - Use invalid source address
  - … and it places pressure on the address space
  - Every host requires "public" IP address
Challenges When Connecting to Public Internet

- Not enough IP addresses for every host in organization
  - Increasingly hard to get large address blocks
- Security
  - Don’t want every machine in organization known to outside world
  - Want to control or monitor traffic in / out of organization

But not All Hosts are Equal!

- Most machines within organization are used by individuals
  - For most applications, they act as clients
- Only a small number of machines act as servers for the entire organization
  - E.g., mail server, web, ...
  - All traffic to outside passes through firewall

(Most) machines within organization do not need public IP addresses!

Reducing Address Use: Network Address Translation

- Within organization:
  assign each host a private IP address
  - IP addresses blocks 10/8 & 192.168/16 are set aside for this
  - Route within organization by IP protocol
  - Can do subnetting, ...
- The NAT translates between public and private IP addresses as packets travel to/from the public Internet
  - It does not let any packets from internal nodes “escape”
  - Outside world does not need to know about internal addresses

NAT: Opening Client Connection

- Client 10.2.2.2 wants to connect to server 198.2.4.5:80
  - OS assigns ephemeral port (1000)
  - Connection request intercepted by firewall
  - Maps client to port of firewall (5000)
  - Creates NAT table entry
NAT: Client Request

- Firewall acts as proxy for client
  - Intercepts message from client and marks itself as sender

<table>
<thead>
<tr>
<th>Int Addr</th>
<th>Int Port</th>
<th>NAT Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.2.2.2</td>
<td>1000</td>
<td>5000</td>
</tr>
</tbody>
</table>

Client Request Mapping

- NAT manages mapping between two four-tuples
- Mapping must be unique: one to one
- Must respect practical constraints
  - Cannot modify server IP address or port number
  - Client has limited number of IP addresses, often 1
  - Mapping client port numbers is important!
- Mapping must be consistent
  - The same for all packets in a communication session

NAT: Server Response

- Firewall acts as proxy for client
  - Acts as destination for server messages
  - Relabels destination to local addresses

<table>
<thead>
<tr>
<th>Int Addr</th>
<th>Int Port</th>
<th>NAT Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.2.2.2</td>
<td>1000</td>
<td>5000</td>
</tr>
</tbody>
</table>

NAT: Enabling Servers

- Use port mapping to make servers available

<table>
<thead>
<tr>
<th>Int Addr</th>
<th>Int Port</th>
<th>NAT Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3.3.3</td>
<td>80</td>
<td>80</td>
</tr>
</tbody>
</table>
Additional NAT Benefits

- They significantly reduce the need for public IP addresses.
- NATs directly help with security:
  - Hides IP addresses used in internal network:
    - Easy to change ISP: only NAT box needs to have IP address
  - Fewer registered IP addresses required
  - Basic protection against remote attack:
    - Does not expose internal structure to outside world
    - Can control what packets come in and out of system
    - Can reliably determine whether packet from inside or outside
- And NATs have many additional benefits:
  - NAT boxes make home networking simple
  - Can be used to map between addresses from different address families, e.g., IPv4 and IPv6

NAT Challenges

- NAT has to be consistent during a session.
  - Mapping (hard state) must be maintained during the session
    - Recall Goal 1 of Internet: Continue despite loss of networks or gateways
    - Recycle the mapping after the end of the session
      - May be hard to detect
  - NAT only works for certain applications:
    - Some applications (e.g., ftp) pass IP information in payload - oops
    - Need application level gateways to do a matching translation
  - NATs are a problem for peer-peer applications:
    - File sharing, multi-player games, ...
    - Who is server?
    - Need to “punch” hole through NAT

Principle: Fate Sharing

- “You can lose state information relevant to an entity’s connections if and only if the entity itself is lost”:
  - Example: OK to lose TCP state if either endpoint crashes
  - The TCP connection is no longer useful anyway!
- It is NOT okay to lose it if an unrelated entity goes down:
  - Example: if an intermediate router reboots
  - NATs violate this principle: if a NAT goes down, all communication session it supports are lost!
    - Unless you add redundancy and put state in persistent storage
  - Bad news: many stateful “middleboxes” violate this rule:
    - Firewalls, mobility services, … - more on this later
  - Good news: today’s hardware is very reliable

Many Options Exist for Peer-Peer

- NAT recognizes certain protocols and behaves as an application gateway:
  - Used for standard protocols such as ftp
- Applications negotiate directly with NAT or firewall – need to be authorized:
  - Multiple protocols dealing with different scenarios
- Punching holes in NAT: peers contact each other simultaneously using a known public (IP, port), e.g. used with rendezvous service
  - Use publicly accessible rendezvous service to exchange accessibility information
  - Assumes NATs do end-point independent mapping
- But remains painful!