Recall: IPv4 Addresses

- 32-bit number, must be *globally unique*

- $2^{32} \Rightarrow 4,294,967,296$ possible addresses

- How many do you have?
Address Scarcity

RIPE NCC Begins to Allocate IPv4 Address Space From the Last /8
14 Sep 2012

On Friday 14 September, 2012, the RIPE NCC, the Regional Internet Registry (RIR) for Europe, the Middle East and parts of Central Asia, distributed the last blocks of IPv4 address space from the available pool.

This means that we are now distributing IPv4 address space to Local Internet Registries (LIRs) from the last /8 according to section 5.6 of "IPv4 Address Allocation and Assignment Policies for the RIPE NCC Service Region".

This section states that an LIR may receive one /22 allocation (1,024 IPv4 addresses), even if they can justify a larger allocation. This /22 allocation will only be made to LIRs if they have already received an IPv6 allocation from an upstream LIR or the RIPE NCC. No new IPv4 Provider Independent (PI) space will be assigned.

It is now imperative that all stakeholders deploy IPv6 on their networks to ensure the continuity of their online operations and the future growth of the Internet.

More information on IPv6 and its deployment, advice from experts and where to get training

More information on reaching the last /8
ARIN Finally Runs Out of IPv4 Addresses

IPv4 Address Cupboards are Bare in North America.

It is often said, “the Internet is running out of phone numbers,” as a way to express that the Internet is running out of IPv4 addresses, to those who are unfamiliar with Internet technologies. IPv4 addresses, like phone numbers are assigned hierarchically, and thus, have inherent inefficiency. The world’s Internet population has been growing and the number of Internet-connected devices continues to rise, with no end in sight. In the next week, the American Registry for Internet Numbers (ARIN) will have exhausted their supply of IPv4 addresses. The metaphorical IPv4 cupboards are bare. This long-predicted Internet historical event marks opening a new chapter of the Internet’s evolution. However, it is somehow anti-climactic now that this date has arrived. The Internet will continue to operate, but all organizations must now accelerate their efforts to deploy IPv6.

ARIN IPv4 Address Exhaustion

The Internet Assigned Numbers Authority (IANA) delegates authority for Internet resources to the five RIRs that cover the world. The American Registry for Internet Numbers (ARIN) is the Regional Internet Registry (RIR) for the United States, Canada, the Caribbean, and North Atlantic islands. ARIN has been managing the assignment of IPv4 and IPv6 addresses and Autonomous System (AS) numbers for several decades. Each RIR has been managing their limited IPv4 address stores and going through their various phases of exhaustion policies. ARIN has been in Phase 4 of their IPv4 depletion plan for more than a year now. ARIN will soon announce that they have completely extinguished their supply of IPv4 addresses.
Private Addresses

• Defined in RFC 1918:
  – 10.0.0.0/8 (16,777,216 hosts)
  – 172.16.0.0/12 (1,048,576 hosts)
  – 192.168.0.0/16 (65536 hosts)

• These addresses shouldn’t be routed.
  – Anyone can use them.
  – Often adopted for use with NAT.
NAT: Network Address Translation

rest of Internet | local network (e.g., home network) 10.0.0/24 |

all datagrams leaving local network have same single source NAT IP address: 138.76.29.7, different source port numbers

datagrams with source or destination in this network have 10.0.0/24 address for source, destination (as usual)
Implementing NAT

- **Outgoing datagrams: replace** (source IP address, port #) of every outgoing datagram to (NAT IP address, new port #)
  - remote clients/servers will respond using (NAT IP address, new port #) as destination address

- **Record (in NAT translation table)** every (source IP address, port #) to (NAT IP address, new port #) translation pair

- **Incoming datagrams: replace** (NAT IP address, new port #) in dest fields of every incoming datagram with corresponding (source IP address, port #) stored in NAT table
NAT: network address translation

1: host 10.0.0.1 sends datagram to 128.119.40.186, 80

2: NAT router changes datagram source addr from 10.0.0.1, 3345 to 138.76.29.7, 5001, updates table

3: reply arrives dest. address: 138.76.29.7, 5001

4: NAT router changes datagram dest addr from 138.76.29.7, 5001 to 10.0.0.1, 3345
NAT: network address translation

<table>
<thead>
<tr>
<th>NAT translation table</th>
</tr>
</thead>
<tbody>
<tr>
<td>WAN side addr</td>
</tr>
<tr>
<td>138.76.29.7, 5001</td>
</tr>
</tbody>
</table>

Neither the sender nor receiver need to know that NAT is happening...

4: NAT router changes datagram dest addr from 138.76.29.7, 5001 to 10.0.0.1, 3345
NAT Advantages

• Organizations need fewer IP addresses from their ISP.
  – With a 16-bit port field, we can put 65535 connections behind one external IP address!

• Organizations can change internal network IPs without having to change outside world IPs.
What about the following statement?

Devices inside the local network are not explicitly addressable or visible by outside world.

A. This is an advantage.

B. This is a disadvantage.
Port Forwarding

• What if we wanted to run a web server on both these hosts?

<table>
<thead>
<tr>
<th>In port</th>
<th>Goes to</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
How do we feel about NAT?

A. NAT is great! It conserves IP addresses and makes it harder to reach non-public machines.

B. NAT is mostly good, but has a few negative features. No big deal.

C. NAT is mostly bad, but in some cases, it’s a necessary evil.

D. NAT is an abomination that violates the end to end principle, and we should not use it!
IPv6

• *Initial motivation:* 32-bit address space soon to be completely allocated, any day now™.

• Additional motivation:
  – header format helps speed processing/forwarding
  – header changes to facilitate QoS

*IPv6 datagram format:*
  – fixed-length 40 byte header
  – no fragmentation allowed
IPv6 datagram format

**class:** identify class/priority of packet

**flow label:** identify datagrams in same “flow.”
(purpose of “flow” not well defined).

**next header:** identify upper layer protocol for data

<table>
<thead>
<tr>
<th>class</th>
<th>flow label</th>
<th>payload len</th>
<th>next hdr</th>
<th>hop limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>ver</td>
<td></td>
<td>source address</td>
<td>next hdr</td>
<td>hop limit</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(128 bits)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>destination address</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(128 bits)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>data</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

32 bits
Other changes from IPv4

• *checksum*: removed entirely to reduce processing time at each hop
• *options*: allowed, but outside of header, indicated by “Next Header” field
• *ICMPv6*: new version of ICMP
  • additional message types, e.g. “Packet Too Big”
  • multicast group management functions
IPv6 (vs. IPv4)

• Simpler, faster, better

• How much traffic on the Internet is IPv6?

• Why!? 
Transitioning to IPv6

• Option 1: “Flag day”
  – How do we get everyone on the Internet to agree?
  – Can you imagine how much would break?

• Option 2: Slow transition
  – Some hosts/routers speak both versions
  – Must have some way to deal with those who don’t
  – Lack of incentive to switch
Dual Stack

IPv6
IPv6
IPv4
IPv4
IPv6
IPv6

flow: X
src: A
dest: F
data

flow: ??
src: A
dest: F
data

A-to-B: IPv6
B-to-C: IPv4
C-to-D: IPv4
D-to-E: IPv4
E-to-F: IPv6
Tunneling

- IPv6 datagram carried as *payload* in IPv4 datagram among IPv4 routers
Tunneling

logical view:

physical view:
ICMP: Internet Control Message Protocol

• Used to communicate network information
  – “Control messages”, i.e., not data themselves
  – Error reporting
    • Unreachable host
    • Unreachable network
    • Unreachable port
    • TTL expired
  – Test connectivity
    • Echo request/response (ping)
ICMP: Internet Control Message Protocol

• Header:
  – 1-byte type
  – 1-byte code
  – 2-byte checksum
  – 4 bytes vary by type

• Sits above IP
  – Type 1 in IP header
  – Usually considered part of IP

<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>
Ping Demo
Traceroute and ICMP

• Source sends sets of UDP segments (usually 3) to dest
  ▪ first set has TTL = 1
  ▪ second set has TTL = 2, etc.
  ▪ unlikely port number

• When $n$th set of datagrams arrives to $n$th router:
  ▪ router discards datagrams
  ▪ and sends source ICMP messages (type 11, code 0)
  ▪ ICMP messages includes name of router & IP address

• When ICMP messages arrives, source records RTTs
Traceroute and ICMP

**stopping criteria:**

- UDP segment eventually arrives at destination host
- destination returns ICMP “port unreachable” message (type 3, code 3)
- source stops
Traceroute Demo
IPsec – Security at the Network Layer

- Encryption

- Data Authentication
  - Verify the data was not modified in transit

- Host Authentication
  - Verify the data comes from who it says it comes from
Two Modes

• Works entirely on endhosts

• Transport
  – Payload is encrypted, IP header is normal

• Tunneling
  – Entire packet is encrypted, stuck inside a different IP packet
  – Can be used to create Virtual Private Networks
Reading

• Routing
  – Section 4.5