IPv6: Internet Protocol v6

- IPv6 - the next generation Internet Protocol
- Much larger address space: 128 bits vs 32 bits
- (Note: not 4x larger, $2^{96}$ times larger!)
- No NAT (restore e2e architectural model)
- Scalable routing (multihoming?)
- Other: ND (better version of ARP), autoconfig
IPv6: Internet Protocol v6

- But primary impetus: is the larger address space
- Impending exhaustion of IPv4 (in progress)
- Adverse consequences of not deploying IPv6
  - IPv4 transfer (black) markets
  - More and more layers of NAT
  - Balkanization; disrupting universal connectivity
IANA Unallocated Address Pool Exhaustion: 01-Feb-2011

Projected RIR Unallocated Address Pool Exhaustion: 29-Jul-2011

ipv4.potaroo.net (Mar 8th 2011)

IPv6 at Penn
IANA Unallocated Address Pool Exhaustion: 01-Feb-2011

Projected RIR Unallocated Address Pool Exhaustion: 18-Apr-2011

Current RIR Address Status

<table>
<thead>
<tr>
<th>RIR</th>
<th>Assigned Addresses (/8s)</th>
<th>Remaining Addresses (/8s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFRINIC</td>
<td>8.2388</td>
<td>4.7573</td>
</tr>
<tr>
<td>APNIC</td>
<td>53.2520</td>
<td>1.7480</td>
</tr>
<tr>
<td>ARIN</td>
<td>77.7171</td>
<td>6.2086</td>
</tr>
<tr>
<td>LACNIC</td>
<td>15.4659</td>
<td>4.5341</td>
</tr>
<tr>
<td>RIPE NCC</td>
<td>44.6711</td>
<td>4.3289</td>
</tr>
</tbody>
</table>

ipv4.potaroo.net (Apr 7th 2011 – one month later only!)

IPv6 at Penn
IPv4 addresses

- 192.168.7.13
- 32 bits
- dotted quad notation (Four 8-bit numbers in range 0..255, separated by dots)
IPv6 addresses

• 128 bits (versus 32-bits for IPv4)
• 8 fields of 16 bits (4 hexadecimal digits) separated by colons (:
• zero compression for more compact format


2001:DB8:3902:7B2::FE04
IPv6 DNS records

Assignments supports the creation of AAAA records and IPv6 PTR (reverse DNS) records.
IPv6 at Penn

IPv4 Header

<table>
<thead>
<tr>
<th>0</th>
<th>4</th>
<th>8</th>
<th>12</th>
<th>16</th>
<th>20</th>
<th>24</th>
<th>28</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>IHL</td>
<td>Type of Service</td>
<td>Total Length</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identification</td>
<td>Flags</td>
<td>Fragment Offset</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time to Live</td>
<td>Protocol</td>
<td>Header Checksum</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Source Address

Destination Address

[followed by options and padding]

IPv6 Header

<table>
<thead>
<tr>
<th>0</th>
<th>4</th>
<th>8</th>
<th>12</th>
<th>16</th>
<th>20</th>
<th>24</th>
<th>28</th>
<th>32</th>
<th>36</th>
<th>40</th>
<th>44</th>
<th>48</th>
<th>52</th>
<th>56</th>
<th>60</th>
<th>63</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>Traffic Class</td>
<td>Flow Label</td>
<td>Payload Length</td>
<td>Next Header</td>
<td>Hop Limit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Source Address

Destination Address

[followed by extensions headers]
Address Assignment

- Servers: use Assignments to statically assign a DNS record and IPv6 address on the subnet the server is on.
- Client computers: will automatically assign themselves an address ("stateless address auto-configuration") - note: no DNS entry typically.
- In the future: we’ll likely deploy DHCPv6 for more active management of client addresses.
IPv6 Subnets

- IPv4 typically has variable sized subnets
  - eg. /24 (256 addr), /23 (512 addr) etc
- Fixed size (usually): 64-bits long
- How many hosts can such a subnet accommodate?
165.123.212.0/24

start: 165.123.212.0
end: 165.123.212.255

2607:f470:ff12:1234::/64

2607:f470:ff12:1234:0000:0000:0000:0000
18,446,744,073,709,551,616

Approximately 18.5 quintillion (US)
Stateless Autoconfig

- Host listens to local Router Advertisements
- Generates host portion of address from MAC
- Router prefix: 2001:468:1802:1::/64
- MAC address: 00:1b:63:94:9d:73
- EUI-64 address: 021b:63ff:fe94:9d73
- This is why you need 64-bit subnets
DHCPv6

- Stateless DHCPv6
- Stateful DHCPv6
  - Managed address allocation
- Easy to populate DNS & reverse DNS
- Some popular clients don’t (yet) support DHCPv6 (Mac OS X)
Temporary addresses

• RFC 4941: Privacy extensions for Stateless Address Auto-configuration

• “Temporary addresses”

• Change over time; derived from MAC initially (1-way hash algorithm), but not able to be tied back to it

• Duration variable: hours, days

• Cons: complicate network debugging, security/audit implications
IPv6 Address Types

- **Global** Addresses
  - (what we talked about so far, but there are other types)
- **Link Local** Addresses (fe80::/10)
- **Tunneled** Addresses (6to4, Teredo, ISATAP, ...)
- Unique Local Addresses
- Multicast Addresses
Link Local Addresses

- All IPv6 interfaces have a link local address
- A special address for communication on the local subnet only
- Subnet range: FE80::/10
- Autoconfigured
- Could be the same on multiple host interfaces
- fe80::21b:63ff:fe94:9d73%en0
  - EUI-64
  - scope-id
Example IP address config on a Mac

```
$ ifconfig -a

lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
    inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
    inet 127.0.0.1 netmask 0xff000000
    inet6 ::1 prefixlen 128

gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280

stf0: flags=1<UP> mtu 1280
    inet6 2002:805b:8446:1::1 prefixlen 16

en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    inet6 fe80::20d:93ff:fe59:c367%en0 prefixlen 64 scopeid 0x4
    inet 128.91.196.48 netmask 0xffffff80 broadcast 128.91.197.255
    ether 00:0d:93:59:c3:67

fw0: flags=8822<BROADCAST,SMART,SIMPLEX,MULTICAST> mtu 4078
    lladdr 00:0d:93:ff:fe:59:c3:67
    media: autoselect <full-duplex> status: inactive
    supported media: autoselect <full-duplex>

en1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    inet6 fe80::20d:93ff:fe84:f10d%en1 prefixlen 64 scopeid 0x6
    inet 128.91.132.70 netmask 0xffffff80 broadcast 128.91.132.127
    ether 00:0d:93:84:f1:0d
```
Example IP address config on Windows

Windows IP Configuration

[...]

Ethernet adapter Local Area Connection:

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection-specific DNS Suffix</td>
<td></td>
</tr>
<tr>
<td>Temporary IPv6 Address</td>
<td>2607:f470:2:1:806c:86ee:b372:47b2</td>
</tr>
<tr>
<td>Link-local IPv6 Address</td>
<td>fe80::2dde:6914:cafe:15fe%10</td>
</tr>
<tr>
<td>IPv4 Address</td>
<td>128.91.196.91</td>
</tr>
<tr>
<td>Subnet Mask</td>
<td>255.255.254.0</td>
</tr>
<tr>
<td>Default Gateway</td>
<td>fe80::216:9cff:fe6f:5dc0%10</td>
</tr>
<tr>
<td></td>
<td>128.91.196.1</td>
</tr>
</tbody>
</table>
Application migration

- *IPv6 is not backwards compatible with IPv4*
- Applications need to be modified to support IPv6 (many popular ones already do)
- Penn needs to consider locally developed applications too.
IPv6 capable servers

- [Very small sampling of things I know about]
- Webservers: Apache, IIS
- Mailservers: Sendmail, Postfix, UWIMAP, Cyrus, MS Exchange, Exim, Qmail, Dovecot, Courier
- DNS: BIND, NSD, PowerDNS, Microsoft DNS
- LDAP: OpenLDAP, Active Directory
- Kerberos: MIT, Heimdal, Active Directory
IPv6 capable clients

- Web browsers: Firefox, Safari, IE, Opera
- E-mail: Apple Mail, Thunderbird, Outlook
- [many more - needs to be filled out]
IPv6 Deployment to date

- MAGPI (to Internet2; Penn, Princeton, NJEdge, Rutgers)
- Entire core network & external connection
- Central server subnets
- Schools: SEAS (Engineering & Applied Science), Law School
- Annenberg school: 1 subnet for video-conferencing with an IPv6 service in China
- Departments: ISC N&T, TSS, other scattered subnets
Penn IPv6 address space

- Older block being phased out:
  - 2001:468:1802::/48

- New portable address space:
  - 2607:f470::/32

- Can deploy 4 billion 64-bit subnets!!
Application deployment

- DNS (authoritative)
- NTP
- SSH (internal server administration)
- Jabber
- Assignments (creation of IPv6 DNS records)
Problem Applications

- Penn Web Server (Akamai hasn’t deployed IPv6, but has plans in the works)
  - Note: www.magpi.net is done though
- Penn Mail Service (Message Labs hasn’t deployed IPv6, and has announced no plans)
Applications planned..

- Kerberos, WebLogin, Shibboleth, LDAP, RADIUS, VoIP, DHCP,

- No timeline yet, but we’ll be working on these in the near future
World IPv6 Day

• http://isoc.org/wp/worldipv6day/

On 8 June, 2011, Google, Facebook, Yahoo!, Akamai and Limelight Networks will be amongst some of the major organisations that will offer their content over IPv6 for a 24-hour "test flight". The goal of the Test Flight Day is to motivate organizations across the industry – Internet service providers, hardware makers, operating system vendors and web companies – to prepare their services for IPv6 to ensure a successful transition as IPv4 addresses run out.

Please join us for this test drive and help accelerate the momentum of IPv6 deployment.
Google over IPv6

Access Google services over IPv6

At Google, we believe that IPv6 is essential to the continued health and openness of the Internet – and that by allowing all devices on a network to talk to each other directly, IPv6 will enable innovation and allow the Internet's continued growth. Typical Google users do not need to do anything to prepare for IPv6, but we are working with network operators to support the transition.

In March 2008, we began offering Google search over IPv6 on IPv6-only websites like ipv6.google.com (IPv6 connection required), but other Google products were not generally available over IPv6.

That's why we created Google over IPv6. If you operate a network that supports IPv6, we may be able to enable Google over IPv6, letting you give users seamless access to most Google services over IPv6 simply by going to the same websites they usually use, such as www.google.com.

How it works

Google over IPv6 uses the IPv4 address of your DNS resolver to determine whether a network is IPv6-capable. If you enable Google over IPv6 for your resolver, IPv6 users of that resolver will receive AAAA records for IPv6-enabled Google services.

Normally, if a DNS resolver requests an IPv6 address for a Google web site, it will not receive one…

www.google.com AAAA?
IPv6 Trials Have Started

This site is intended to provide the latest information about Comcast’s IPv6-related work. We are conducting several IPv6 technical trials in our production network, with customers, in order to prepare for the IPv6 transition. This site will be updated as new information about these trials comes out, and as other IPv6-related work occurs.

IPv6 Trial News and Information:

Comcast to Participate in World IPv6 Day
Tuesday, February 15, 2011

Comcast and the Internet Society today announced that Comcast will participate in World IPv6 Day on June 8, 2011. We anticipate having our IPv6 trial users participate in this event, which will give them the opportunity to access many more sites natively over IPv6. In addition, we plan to have more of our websites available over IPv6 (the current list of IPv6-ready sites is here).
Other developments

- OMB mandate: All major US government agencies were required to demonstrate their adoption of IPv6 recently
- OECD declaration of IPv6 support
- And generally, accelerated deployment of IPv6 in AsiaPac, Europe, and elsewhere
Upcoming Steps for Penn

- Campus rollout plan (ie. rest of campus)
- More application services:
  - Penn website, Email, AuthN/Z, Directory, DHCPv6
IPv6 for the rest of campus

- ISC plans to enable IPv6 on all PennNet wired subnets throughout campus, starting end of semester and proceeding through the early part of the summer.

- We think this is quite safe, based on many years of experience with it in other areas of campus: SEAS, ISC (and more recent deployments elsewhere).

- Also based on the experience of several peer universities, who have done this for many years.

- In some cases it will solve problems caused by the use of tunneling protocols (6to4, Teredo).
IPv6 for the rest of campus

• Note: we are only enabling it on the network

• We aren’t asking people to deploy IPv6 on their servers or applications, but recommend that you plan to do so at your own pace

• IPv6 capable clients will configure a working IPv6 address, and may try to use IPv6 with the few (but growing) IPv6 capable services out there

• This is generally transparent. Our experience is that most users don’t notice that IPv6 is even on
Campus Rollout Schedule

• All wired subnets: start mid-May and end by June 30th

• A building by building schedule has been published by our Network Operations group

• http://www.upenn.edu/computing/pennnet/maintschedule.html
Automatic Tunneling

- Even without IPv6 deployed in your network, computers may be using IPv6

- Via automatic tunneling mechanisms: 2 popular ones are 6to4 and Teredo

- These work by encapsulating IPv6 packets inside IPv4 packets and sending them to a relay router that is connected to the IPv6 Internet

- Tunnels sometimes cause connectivity problems. Native IPv6 deployment usually fixes all of the problems.
IPv6 at Penn

- IPv4 Internet
  - 6to4 Relay (inbound)
  - IPv6 Service
- IPv6 Internet
  - 6to4 Relay (outbound)
  - PennNet
  - 6to4 tunnelled IPv6 client
  - Native IPv6 client

Poorer, unpredictable performance, via more circuitous path, through relay routers not operated by us, which may not always be up, and with asymmetrical path in outbound & inbound direction.

Better, more predictable performance, more direct path; easier to troubleshoot.
Identifying Tunneled IPv6

- 6to4 uses the well known prefix 2002::/16
- Teredo uses the well known prefix 2001::/32 (ie. 2001:0000::/32)
- Both use the value 41 (IPv6 encapsulation) in the IP protocol field
- 6to4 IPv6 packets are encapsulated directly in IPv4
- Teredo is encapsulated in UDP inside IPv4
- 6to4 often uses well-known anycast relay routers (192.88.99.1)
- There are also public Teredo servers and relays
Future Deployment

- We won’t enable IPv6 on any network behind an officially recognized firewall, IDS or security device.
- In these cases, we’ll ask the operator to examine their security device configuration to assess the impact of IPv6 traffic passing through it first. And will work on a mutually convenient schedule to turn on IPv6 sometime in the future (FY’12).
- IPv6 on the Penn wireless networks will come later in the year (summer?)
Migrating Applications

- See if your application service, and supported client software, already supports IPv6
- If so, test it with clients, and plan to turn it on
- May need to upgrade the app, or port it to IPv6 if not
- A lot of popular software supports IPv6 already
- Take into account IPv6 addresses in logging, auditing functions, etc
Preparation time

- Starting early is better
- Develop plan
- Training your staff and users
- Ordering/updating hardware & software
- Installing/testing/debugging h/w and s/w
- On both server and client side
Transition vs Co-existence

- IPv4 isn’t going away anytime soon, possibly not for decades
- So we are **not transitioning** to IPv6 (yet)
- We are deploying IPv6 to **co-exist** with IPv4 for the foreseeable future
- To allow us to communicate with both the IPv4 and IPv6 Internet
References

• http://www.upenn.edu/computing/ipv6/
• http://www.upenn.edu/computing/ipv6/strategy.html
• http://www.getipv6.info/index.php/Main_Page
• http://ipv6.com/
References

• RFC 2460: Internet Protocol Version 6 Specification
• RFC 4291: IPv6 Addressing Architecture
• RFC 4861: Neighbor Discovery for IPv6
• RFC 4862: IPv6 Stateless Address Autoconfiguration
• RFC 4941: Privacy Extensions for Stateless Autoconfig
• RFC 3315: DHCP v6
• RFC 3736: Stateless DHCPv6
• RFC 3056: 6to4: Connection of IPv6 domains via IPv4 Clouds
• RFC 4380: Teredo: tunneling IPv6 over UDP through NATS
Questions?

Shumon Huque
shuque -@- upenn.edu
To add to talk

- Dual stack address selection algorithm
- Rogue router advertisements
- Private addresses: ULA
- Subnet addressing scheme for servers
- Impact on IP based billing models at Penn
- ICMP essential (ND, Router disc, Multicast, etc)
- AF Translators: NAT64/DNS64, CGN, DSLite