IPv6 - Benefits and Deployment Issues

Dr. Chris Edwards
Lancaster University, UK
ce@comp.lancs.ac.uk
Agenda for Today

- Introduction to IPv6
  - Benefits of IPv6

- Deployment Issues
  - Transitioning
  - A full service?
Networking Today

- Severe shortage of IP addresses
  - Limits growth for existing users
  - Hinders use for new users
- Routing table explosion
- Management nightmare
- No support for new applications
  - Mobility, QoS, etc...
- Drive for commercialisation killed network exp.
What is IPv6

- IETF standard for the next generation IP
  - AKA IPng

- Design goals
  - Address the failings of IPv4
  - Namely:
    - Scalability
    - Efficiency
    - Extensibility
IPv4 lacks scalability due to addressing
  - 32 bits address space (4.4 Bn addresses)
  - Most addresses allocated to US

More addresses, please!
  - Individually address all mobile handsets
  - Growth of “always on”, globally addressable devices
  - Peer-to-peer computing, e.g. ICQ, video/VoIP
  - Home networking appliances, pervasive computing devices
Users on the Internet – September 2002

- CAN/US 182.67M
- Europe 190.92M
- Asia/Pac 187.24M
- Latin Am 33.35M
- Africa 6.31M
- Mid-east 5.12M

Total 605.6 M

Thanks to Vint Cerf, WorldCom, and www.nua.com
Internet User Trends

Source: Nua Internet Surveys + Vint Cerf predictions
More Predictions…

605 Million users

Source: Cerf, based on www.nw.com, Jun 2000 + Ericsson
IPv6 – Size Matters...

- Extended address space
  - 128 bits long
  - Unicast, Multicast or Anycast formats
  - Written in hex notation as 16-bit integers
    - E.g. 2001:630:80:0:0:0:0:1

- $3.4 \times 10^{38}$ Addresses
- $6.7 \times 10^{23}$ Addresses / m$^2$ on the earth
IPv6 Addressing Model

- Addresses are assigned to interfaces
- Interfaces can have multiple addresses
- Addresses have scope: *link local, site local, global*

- Addresses are formed through the combination of:
  - Routing Prefix – *where you are connected to*
  - Interface ID – *who you are*
**Aggregatable Addresses**

- **Format prefix**: 3 bits (001)
- **Reserved**: 8 bits
- **Prefix length**: 13, 24, 16, 64

**TLA** (Top Level Aggregation identifier)
**NLA** (Next Level Aggregation identifier)
**SLA** (Site Level Aggregation identifier)

IPv6 terminology can drop a single string of all 0s…
- 2001:630:0080:7030::1/64
- 2001:630:0080:7030::/64

**loopback**: ::1  **unspecified**: ::0  **IPv4 Compatible**: ::148.88.8.6
IPv6 General Concepts

- Improved routing techniques
  - Aggregated routing entries designed to reduce routing table sizes
- Multicast supported as native communication mode
- Authentication and privacy capabilities
  - Authentication header
  - Transport + Tunnel Mode
Efficient Header Construction

- IPv4 contains many redundant features...
  - Variable length IP header options
  - IP header checksum
- ...some inefficient ones...
  - Packet fragmentation
- ... and some omitted
  - Packet classification
- All of which impact network performance
### IP: Head to Head

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| **Version** | **IHL** | **Type of Service** | **Total Length** |
| **Identification** | **Flags** | **Fragment Offset** |
| **Time to Live** | **Protocol** | **Header Checksum** |
| **Source Address** |
| **Destination Address** |
| **Options** | **Padding** |
IP: Head to Head

+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Version | Traffic Class |           Flow Label                  |
+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |  Next Header  |   Hop Limit   |
+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                         Source Address                        |
|                                                               |
+                                                               +
|                                                               |
|                                                               |
+--+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                      Destination Address                      |
|                                                               |
+                                                               +
|                                                               |
|                                                               |
+--+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
What’s missing?

- The IPv6 protocol header is streamlined for the common-case...
  - Fixed format header (no options)
  - No checksum - left to transport and data link layers, no need to check/recalculate each hop
  - No fragmentation (except at source)
    - Agree path **MTU** at the source using **Path MTU discovery**
What’s new

- Revised fields
  - Payload length vs. Total length
  - Next Header vs. Protocol type
  - Hop Limit vs. TTL

- New fields
  - Traffic Class:
    - To support differentiated services (e.g. prioritised best effort queuing)
  - Flow Label:
    - Along with source address, allows identification of packets which are part of a ‘flow’
Extensible headers

- Custom headers for specialist functionality...
  - Fragmentation Headers
  - Routing Headers
  - Destination Options
  - Hop by Hop Headers
  - Authentication and ESP
IPv6 Extension Headers

- In IPv6, Options are daisy-chained in extension headers...

<table>
<thead>
<tr>
<th>IPv6 Header</th>
<th>Routing Hdr</th>
<th>Fragment Hdr</th>
<th>TCP Header + Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next header = TCP</td>
<td>Next header = Routing</td>
<td>Next header = Fragment</td>
<td>TCP Header + Data</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IPv6 Header</th>
<th>Routing Hdr</th>
<th>TCP Header + Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next header = TCP</td>
<td>Next header = TCP</td>
<td>TCP Header + Data</td>
</tr>
</tbody>
</table>
Autoconfiguration

- Plug ‘n’ Play Networking...
  - IPv6 host requires three pieces of info
    - IPv6 Address
    - IPv6 Network
    - IPv6 Gateway
  - Router Solicitation and Advertisement...
Autoconfiguration

- Plug ‘n’ Play Networking...
  - IPv6 host requires three pieces of info
    - IPv6 Address
    - IPv6 Network
    - IPv6 Gateway
  - Router Solicitation and Advertisement...

Router Advertisement
2001:630:80:7000::/64
Autoconfiguration

- Host builds IPv6 address from prefix
  - Using EUI-64 identifier of interface
  - Or padded MAC address...
  - In two frame message exchange

Router Advertisement
2001:630:80:7000::/64
Deployment Issues: Transitioning to IPv6...

Contrary to popular belief, IPv6 is not backward compatible...
Compatibility Issues

- Introduce IPv6 connectivity into the IPv4 world
- IPv6 hosts must be able to communicate with each other across IPv4 nets
- As native IPv6 networks become commonplace
  - IPv4 hosts will need to communicate with each other across IPv6 networks
  - IPv6 hosts will need to communicate with IPv4 hosts
- What about dual stacks?
Three broad categories

- **Tunnelling** (6to4, 6over4, Tunnel brokers, Teredo, ISATAP)
  - Allowing hosts that support IPv6 to talk IPv6 to other IPv6 hosts
- **Translation** (SIIT, NAT-PT, SOCKS, BIS)
  - Translation between the two
- **Dual Stack** (DSTM)
Dual IP Stacks

- Simplest method: Both stacks in parallel in hosts and routers
- Upgrade routers, and host OS
  - Host upgrade can be gradual
- Application support:
  - Existing applications continue to run
  - IPv6 applications (experimental or not...) can be introduced
  - Interoperation of v4 and v6 is another issue
    - Applications to be modified to handle both?
Dual IP Stacks (2)

- **Issues**
  - Solution does not scale:
    - New IPv6 hosts that need IPv4 compatibility will quickly eat up IPv4 address space
    - Two IP routing tables will place a burden on routers
  - Entire path dual stack?
  - No real stimulus for moving to IPv6
Dual Stack Transition Mechanism (DSTM)

- Addresses problem of new dual stack hosts exhausting sparse IPv4 address space
- Allows IPv6 hosts to temporarily acquire an IPv4 global address
  - Uses a DHCPv6 server within each domain
  - Assigns IPv4 address on temporary basis
- In instances where IPv6 hosts remain online, temporary assignment becomes permanent... i.e. does not eradicate the problem altogether
Tunnelling

- Common mechanism, where one protocol is encapsulated in another
- IPv6 over IPv4 tunnelling
  - Used to transport IPv6 packets over networks that can only understand IPv4
  - Normally the most common transition mechanism adopted in early stages
  - 6BONE is an example of a virtual overlay network of interconnected IPv6 over IPv4 tunnels
  - Can work in a variety of ways: host to router, router to router, router to host, host to host
IPv6 over IPv4 tunnels are classified as either *configured* or *automatic*, depending on the way the IPv4 address of the endpoint is determined.

**Approaches**
- **6to4** – popular, automatic, router to router
- **6over4** – single site, relies on IPv4 multicast
- **Teredo** – connectivity to v4 hosts behind NAT
- **ISATAP** – site based, where v6 host and gateway is separated
- **Tunnel brokers** – (web-based) mechanism for obtaining a tunnel
Translation Tools

- Translation necessary for IPv6-only and IPv4-only hosts to communicate, should be done near network edge
- Translates packets from one protocol to another, taking form of header processing
- Can take place at a number of layers
  - IP layer
  - Transport layer
  - Application layer
Translation Tools (2)

Series of tools available

- **SIIT** – translates between IPv4 and IPv6 headers using a translation algorithm located in the network
- **NAT-PT**– maintains a pool of unique v4 add. that it dynamically allocates to v6 nodes
- **BIS** – takes NAT-PT with SIIT functionality and moves it to the OS protocol stack within each host
- **SOCKS** – application layer IPv6/IPv4 gateway based on SOCKS, translating between two terminated v4 and v6 connections
The IPv6 Operations
IETF Working Group (v6ops)

- Ngtrans group closed, replaced by v6ops
  - More “operationally oriented” wg
  - Operating the “combined net” + avoiding a division
  - Develop guidelines:
    - Operation of shared v4/v6 Internet
    - How to deploy v6 into both v4-only and new installations

- Why?
  - V6 is deploying today
  - V6 has been hiding in a corner of the IETF

Deployment Issues:

A full service?
The bigger questions

- How long will deployment take?
  - 2 addressing modes co-existing
  - 5/10/15 years?

- What support do we have now?
  - Hosts
  - Routers
  - Applications
  - ISPs
  - (see Tim and Duncan presentation later)
A Full IPv6 Service?

- Production service needs commercial code
  - Stability, reliability issues
  - On backbone
    - Possible now
    - Hardware-enabled?
  - Applications
    - Perhaps some way off yet
A Full IPv6 Service? (2)

- Do we have a significant driver?
  - Less demand where IPv4 address space seen as being sufficient
  - Users will not demand "IPv6" but demand IPv6-based applications
  - IPv4 address exhaustion?
  - Major network infrastructure deployment (e.g., 3G)

- Significant research activities in UK and Europe
  - Universities, NRENs, Consortiums

- UK IPv6 Task Force
Thank you...

Questions?