IPv6 Address Planning

Philip Smith

APNIC 34

21st – 31st August 2012

Address Planning

- IPv6 address space available to each network operator is very large compared with IPv4
  - Design a scalable plan
  - Be aware of industry current practices
  - Separation of infrastructure and customer addressing
  - Distribution of address space according to function

Why Create an Addressing Plan?

- The options for an IPv4 addressing plan were severely limited:
  - Because of scarcity of addresses
  - Every address block has to be used efficiently
- IPv6 allows for a scalable addressing plan:
  - Security policies are easier to implement
  - Addresses are easier to trace
  - Enables more efficient network management

Nibble Boundaries

- IPv6 offers network operators more flexibility with addressing plans
  - Network addressing can now be done on nibble boundaries
    - For ease of operation
  - Rather than making maximum use of a very scarce resource
    - With the resulting operational complexity
- A nibble boundary means subdividing address space based on the address numbering
  - Each number in IPv6 represents 4 bits
  - Which means that IPv6 addressing can be done on 4-bit boundaries

Nibble Boundaries – example

- Consider the address block 2001:db8:0:10::/61
  - The range of addresses in this block are:

```
2001:0db8:0000:0010:0000:0000:0000:0000
                  to
2001:0db8:0000:0017:ffff:ffff:ffff:ffff
```

  - Note that this subnet only runs from 0010 to 0017.
  - The adjacent block is 2001:db8:0:18::/61

```
2001:0db8:0000:0018:0000:0000:0000:0000
                  to
2001:0db8:0000:001f:ffff:ffff:ffff:ffff
```

  - The address blocks don’t use the entire nibble range

Nibble Boundaries – example

- Now consider the address block 2001:db8:0:10::/60
  - The range of addresses in this block are:

```
2001:0db8:0000:0010:0000:0000:0000:0000
                   to
2001:0db8:0000:001f:ffff:ffff:ffff:ffff
```

  - Note that this subnet uses the entire nibble range, 0 to f
  - Which makes the numbering plan for IPv6 simpler
    - This range can have a particular meaning within the ISP block (for example, infrastructure addressing for a particular PoP)

Addressing Plans – Infrastructure

- Network Operators should procure a /32 from their RIR
- Address block for infrastructure
  - /48 allows 65k subnets in the backbone
- Address block for router loop-back interfaces
  - Number all loopbacks out of one infrastructure /64
  - /128 per loopback
- Point-to-point links
  - /64 reserved for each, address as a /127
- LANs
  - /64 for each LAN

Addressing Plans – Customer

- Customers get one /48
  - Unless they have more than 65k subnets in which case they get a second /48 (and so on)
- However, typical industry customer assignments today:
  - /64 for just one LAN
  - /60 for a small network – 16 subnets
  - /56 for a medium network – 256 subnets
  - /48 for a large network – 65536 subnets
- This is still a very active discussion area

Deployable Address Plan

- Documentation
  - IPv4 addresses are probably short enough to memorise
  - IPv6 addresses are unlikely to be memorable at all
- Document the address plan
  - What is used for infrastructure
  - What goes to customers
  - Flat file, spreadsheet, database, etc
  - But documentation is vital
  - Especially when coming to populating the DNS later on

Addressing Tools

- Examples of IP address planning tools:
  - NetDot netdot.uoregon.edu (recommended!!)
  - HaCi sourceforge.net/projects/haci
  - IPAT nethead.de/index.php/ipat
  - freeipdb home.globalcrossing.net/~freeipdb/
- Examples of IPv6 subnet calculators:
  - ipv6gen code.google.com/p/ipv6gen/
  - sipcalc www.routemeister.net/projects/sipcalc/

Deployable Address Plan

- Pick the first /48 for our infrastructure
  - Reason: keeps the numbers short
  - Short numbers: less chance of transcription errors
  - Compare:

    2001:db8:ef01:d35c::1/128

    with

    2001:db8::1/128

    For Loopback interface addresses
- Out of this /48, pick the first /64 for loopbacks
  - Reason: keeps the numbers short

Deployable Address Plan

- Pick the second /48 for point-to-point links to customers
  - Addresses not a trusted part of Operator’s infrastructure
- Divide the /48 between PoPs
  - e.g. 10 PoPs -> split into /52s -> 4096 links per /52
  - Gives 65536 /64s for 65536 customer links
    - /64 per link, number as /127 as previously
  - Adjust number of /48s to suit PoP size (one /48 per PoP?)
- Alternative is to use unnumbered interfaces

Deployable Address Plan

- For the infrastructure /48:
  - First /64 for loopbacks
  - Maybe reserve the final /60 for the NOC
    - Gives 16 possible subnets for the Network Operations Centre (part of the Infrastructure)
  - Remaining 65519 /64s used for internal point-to-point links
    - More than any network needs today

Example: Loopback addresses

- 2001:db8:0::/48 is used for infrastructure
- Out of this, 2001:db8:0:0::/64 is used for loopbacks
- Network Operator has 20 PoPs
  - Scheme adopted is 2001:db8::XXYY/128
  - Where X is the PoP number (1 through FF)
  - Where Y is the router number (1 through FF)
  - Scheme is good for 255 PoPs with 255 routers per PoP, and keeps addresses small/short

Example: Loopback addresses

- Loopbacks in PoP 1:

CR1	2001:db8::101/128
CR2	2001:db8::102/128
BR1	2001:db8::103/128
BR2	2001:db8::104/128
AR1	2001:db8::110/128
AR2	2001:db8::111/128
AR3	2001:db8::112/128
AR4	2001:db8::113/128

…etc…

Loopbacks in PoP 10:

CR1	2001:db8::a01/128
CR2	2001:db8::a02/128
BR1	2001:db8::a03/128
BR2	2001:db8::a04/128
AR1	2001:db8::a10/128
AR2	2001:db8::a11/128
AR3	2001:db8::a12/128
AR4	2001:db8::a13/128

…etc…

Example: Backbone Point-to-Point links

- ISP has 20 PoPs
  - Scheme adopted is 2001:db8:0:XXYY::Z/64
  - Where:
    - XX is the PoP number (01 through FF)
    - YY is the LAN number (when YY is 00 through 0F)
    - YY is the P2P link number (when YY is 10 through FF)
    - Z is the interface address (2 or 3)
    - /64 is reserved, but the link is numbered as a /127
  - Scheme is good for 16 LANs and 240 backbone PtP links per PoP, and for 255 PoPs

Example: Backbone Point-to-Point links

- PtP & LANs in PoP 1:

LAN1	2001:db8:0:100::/64
LAN2	2001:db8:0:101::/64
LAN3	2001:db8:0:102::/64
PtP1	2001:db8:0:110::/64
PtP2	2001:db8:0:111::/64
PtP3	2001:db8:0:112::/64
PtP4	2001:db8:0:113::/64
PtP5	2001:db8:0:114::/64
…etc…	

- PtP & LANs in PoP 14:

LAN1	2001:db8:0:e00::/64
LAN2	2001:db8:0:e01::/64
LAN3	2001:db8:0:e02::/64
LAN4	2001:db8:0:e03::/64
LAN5	2001:db8:0:e04::/64
PtP1	2001:db8:0:e10::/64
PtP2	2001:db8:0:e11::/64
PtP3	2001:db8:0:e12::/64
…etc…	

Links to Customers (1)

- Some Network Operators use unnumbered IPv4 interface links
  - So replicate this in IPv6 by using unnumbered IPv6 interface links
  - This will not require one /48 to be taken from the ISP's /32 allocation

Links to Customers (2)

- Other Network Operators use global unicast addresses
  - So set aside the second /48 for this purpose
    - And divide the /48 amongst the PoPs
  - Or set aside a single/48 per PoP (depending on network size)
  - Each /48 gives 65536 possible customer links, assuming a /64 for each link
- Scheme used:
  - 2001:db8:00XX::/48 where XX is the PoP number
  - Good for 255 PoPs with 65536 point-to-point links each

Example

- Customer PtP links
  - PoP1

Reserved	2001:db8:1:0::/64
Customer1	2001:db8:1:1::/64
Customer2	2001:db8:1:2::/64
Customer3	2001:db8:1:3::/64
Customer4	2001:db8:1:4::/64

  - PoP12

Reserved	2001:db8:c:0::/64
Customer1	2001:db8:c:1::/64
Customer2	2001:db8:c:2::/64
Customer3	2001:db8:c:3::/64

  - …etc…

Example: Customer Allocations

- Master allocation documentation would look like this:

2001:db8:0::/48	Infrastructure
2001:db8:1::/48	PtP links to customers (PoP1)
2001:db8:2::/48	PtP links to customers (PoP2)
2001:db8:3::/48	PtP links to customers (PoP3)
…	
2001:db8:100::/48	Customer 1 assignment
…	
2001:db8:ffff::/48	Customer 65280 assignment

- Infrastructure and Customer PtP links would be documented separately as earlier

Summary

- First /48 for infrastructure
  - Out of that, first /64 for Loopbacks
- Defined structure within IPv6 addressing is recommended
  - Greater flexibility than with IPv4
  - Possible to come up with a simple memorable scheme
- Documentation vitally important!