IPv6 Deployment

Contact: training@apnic.net

APNIC

TIP602_v1.0

Overview

- Introduction to IPv6
- IPv6 Protocol Architecture
- Mobile IPv6 Operation
- IPv6 Security Features
- IPv6 Addressing and Subnetting
- IPv4 to IPv6 Transition Technologies
- IPv6 Services

Overview

- **Introduction to IPv6**
- IPv6 Protocol Architecture
- Mobile IPv6 Operation
- IPv6 Security Features
- IPv6 Addressing and Subnetting
- IPv4 to IPv6 Transition Technologies
- IPv6 Services

Intro to IPv6

What is IPv6?

- **IP** stands for Internet Protocol which is one of the main pillars that supports the Internet today
- Current version of IP protocol is IPv4
- The new version of IP protocol is IPv6
- There is a version of IPv5 but it was assigned for experimental use [RFC1190]
- IPv6 was also called IPng in the early days of IPv6 protocol development stage

Background of IPv6 Protocol

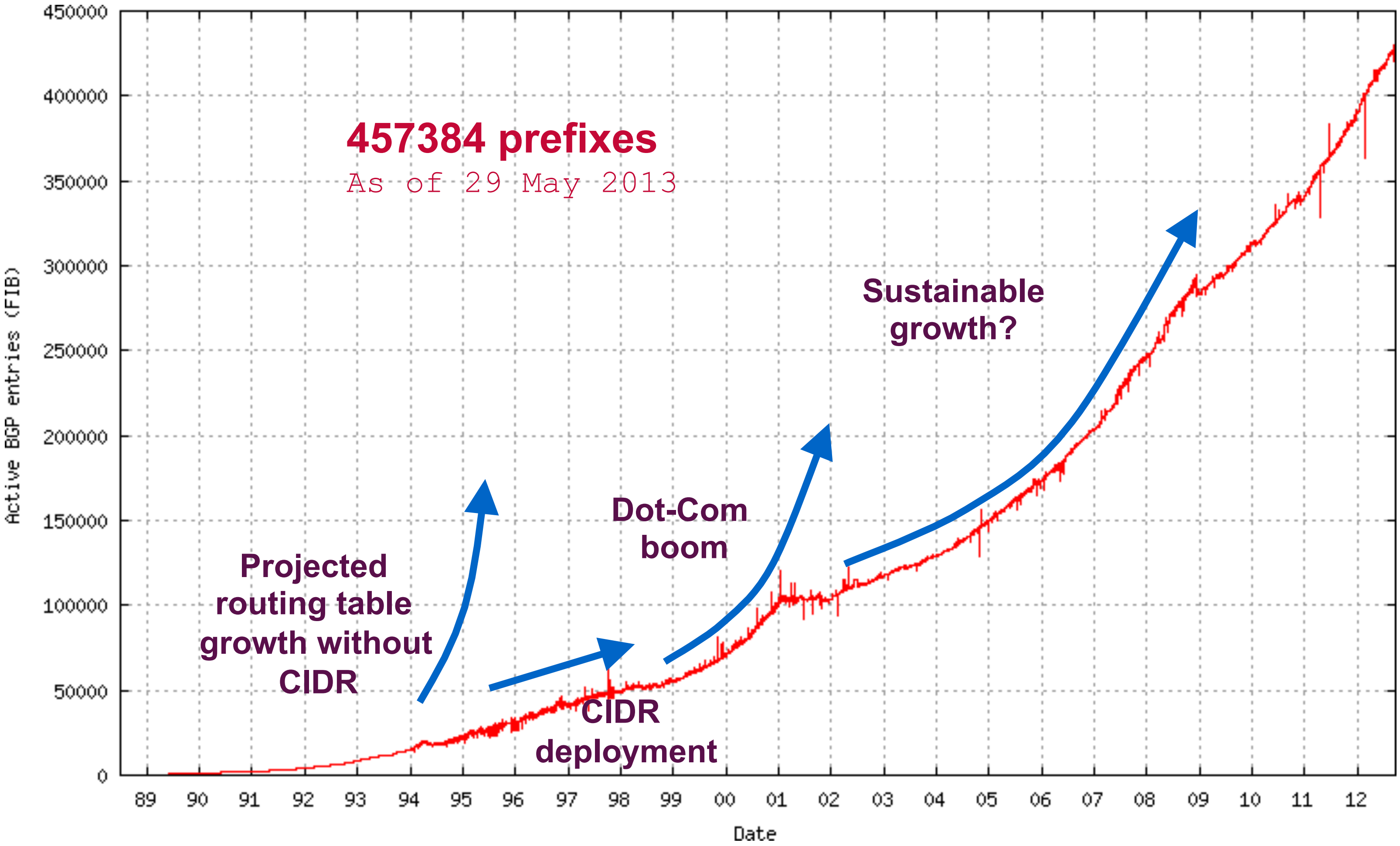
- August 1990
  - First wakeup call by Solensky in IETF on IPv4 address exhaustion
- December 1994
  - IPng area were formed within IETF to manage IPng effort [RFC1719]
  - List of technical criteria was defined to choose IPng [RFC1726]
- January 1995
  - IPng director recommendation to use 128 bit address [RFC1752]
- December 1995
  - First version of IPv6 address specification [RFC1883]
- December 1998
  - Updated version changing header format from 1st version [RFC2460]

Motivation Behind IPv6 Protocol

- Plenty of address space (Mobile Phones, Tablet Computers, Car Parts, etc. ☺)
- Solution of very complex hierarchical addressing need, which IPv4 is unable to provide
- End to end communication without the need of NAT for some real time application (i.e online transaction)
- Ensure security, reliability of data and faster processing of protocol overhead
- Stable service for mobile network (i.e Internet in airline, trains)

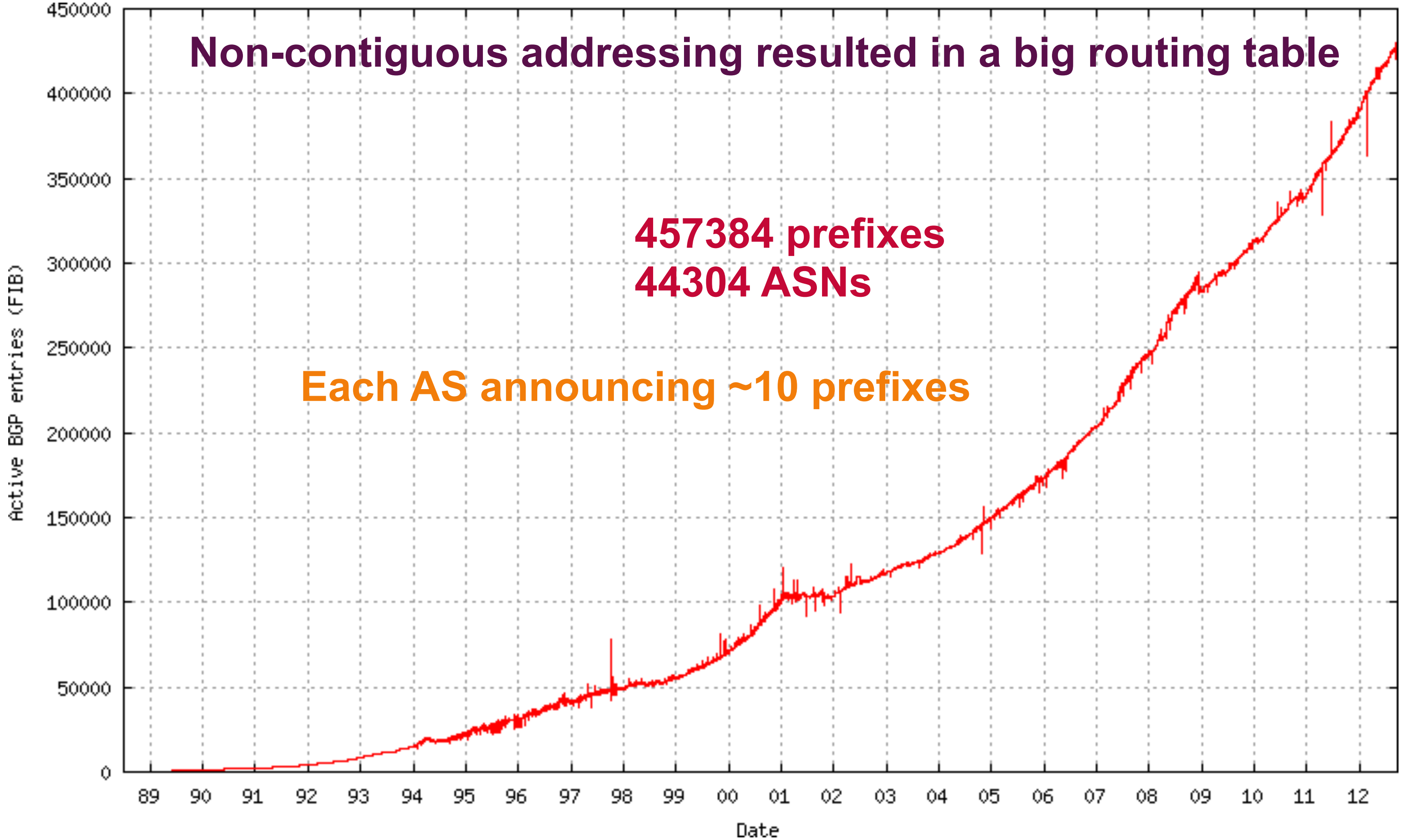
Growth of the Global Routing Table

http://bgp.potaroo.net/as1221/bgp-active.html

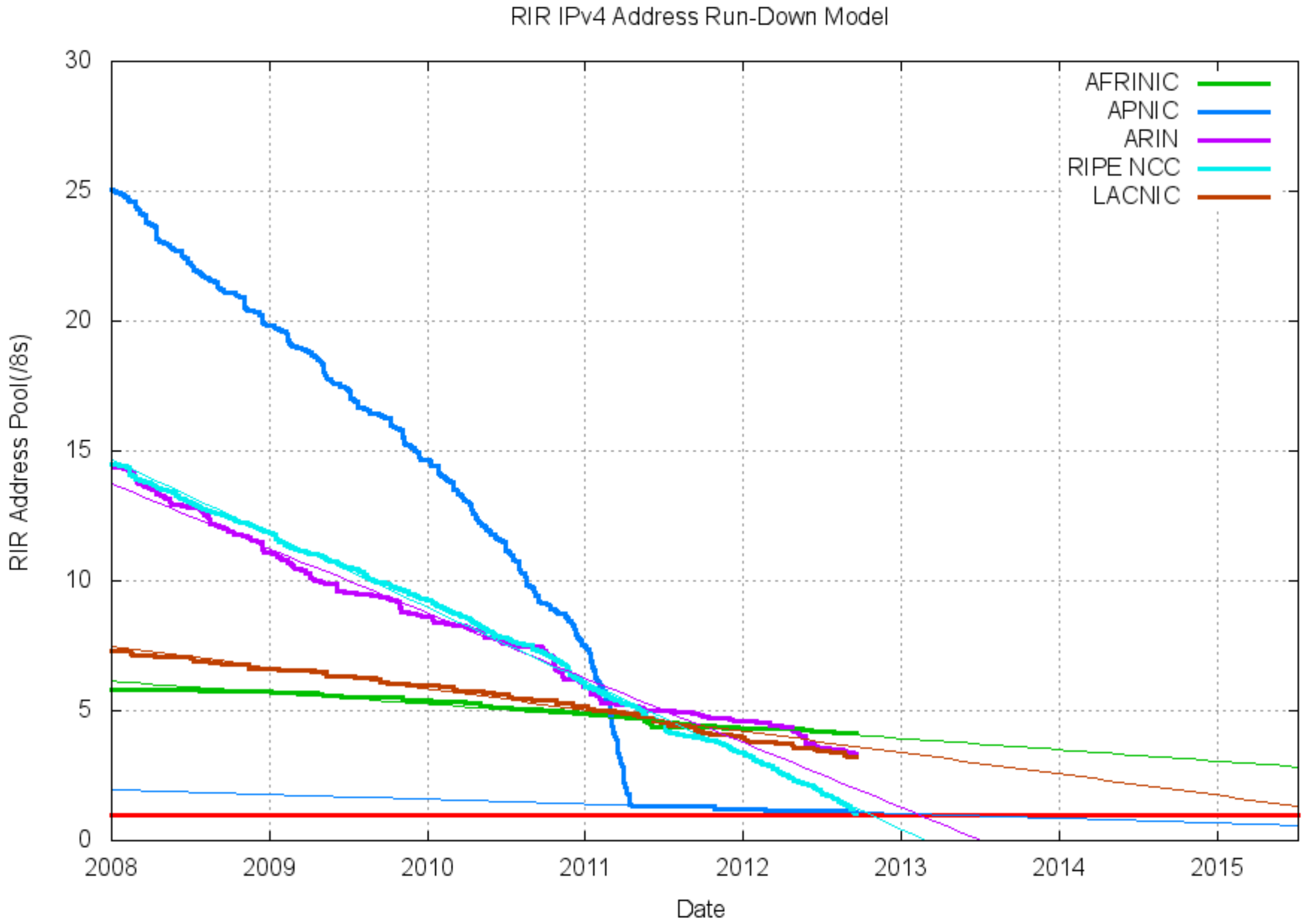


IPv4 BGP Table

http://bgp.potaroo.net/as1221/bgp-active.html



IPv4 Exhaustion



New Functional Improvement

- Address Space
  - Increase from 32-bit to 128-bit address space
- Management
  - Stateless autoconfiguration means no more need to configure IP addresses for end systems, even via DHCP
- Performance
  - Fixed header size (40 bytes) and 64-bit header alignment mean better performance from routers and bridges/switches
- No hop-by-hop segmentation
  - Path MTU discovery

Source: http://www.opus1.com/ipv6/whatisipv6.html

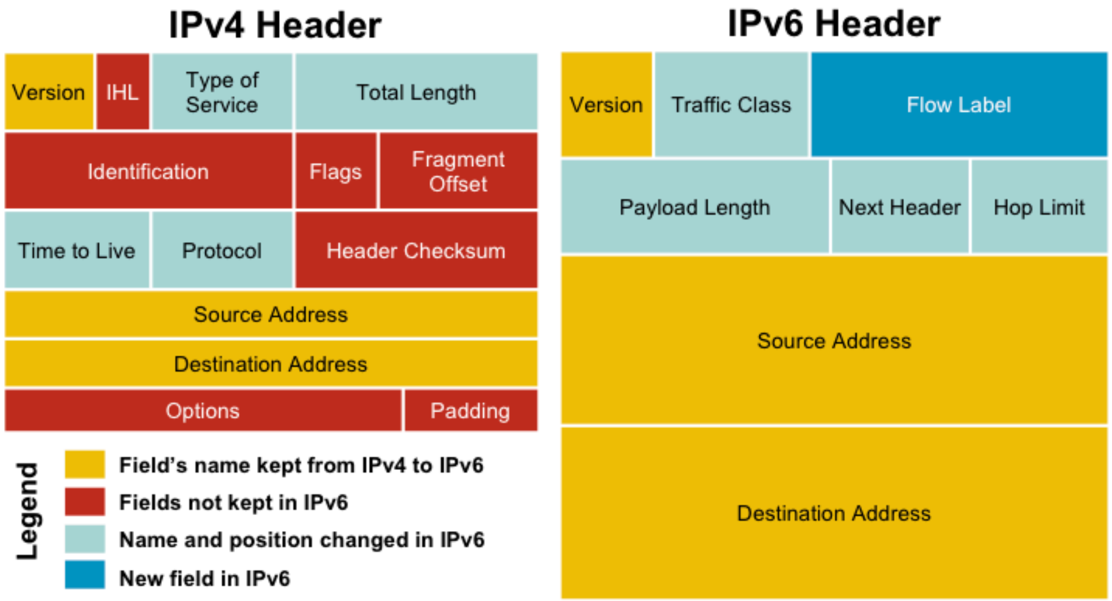
New Functional Improvement

- Multicast/Multimedia
  - Built-in features for multicast groups, management, and new "anycast" groups
- Mobile IP
  - Eliminate triangular routing and simplify deployment of mobile IP-based systems
- Virtual Private Networks
  - Built-in support for ESP/AH encrypted/ authenticated virtual private network protocols;
- Built-in support for QoS tagging
- No more broadcast

Overview

- Introduction to IPv6
- **IPv6 Protocol Architecture**
- Mobile IPv6 Operation
- IPv6 Security Features
- IPv6 Addressing and Subnetting
- IPv4 to IPv6 Transition Technologies
- IPv6 Services

Protocol Header Comparison



- IPv4 contains 10 basic header field
- IPv6 contains 6 basic header field
- IPv6 header has 40 octets in contrast to the 20 octets in IPv4
- So a smaller number of header fields and the header is 64-bit aligned to enable fast processing by current processors

Diagram Source: www.cisco.com

IPv6 Protocol Header Format

- The IPv6 header fields:
- **Version**
  - A 4-bit field, same as in IPv4. It contains the number 6 instead of the number 4 for IPv4
- **Traffic class**
  - A 8-bit field similar to the type of service (ToS) field in IPv4. It tags packet with a traffic class that it uses in differentiated services (DiffServ). These functionalities are the same for IPv6 and IPv4.
- **Flow label**
  - A completely new 20-bit field. It tags a flow for the IP packets. It can be used for multilayer switching techniques and faster packet-switching performance

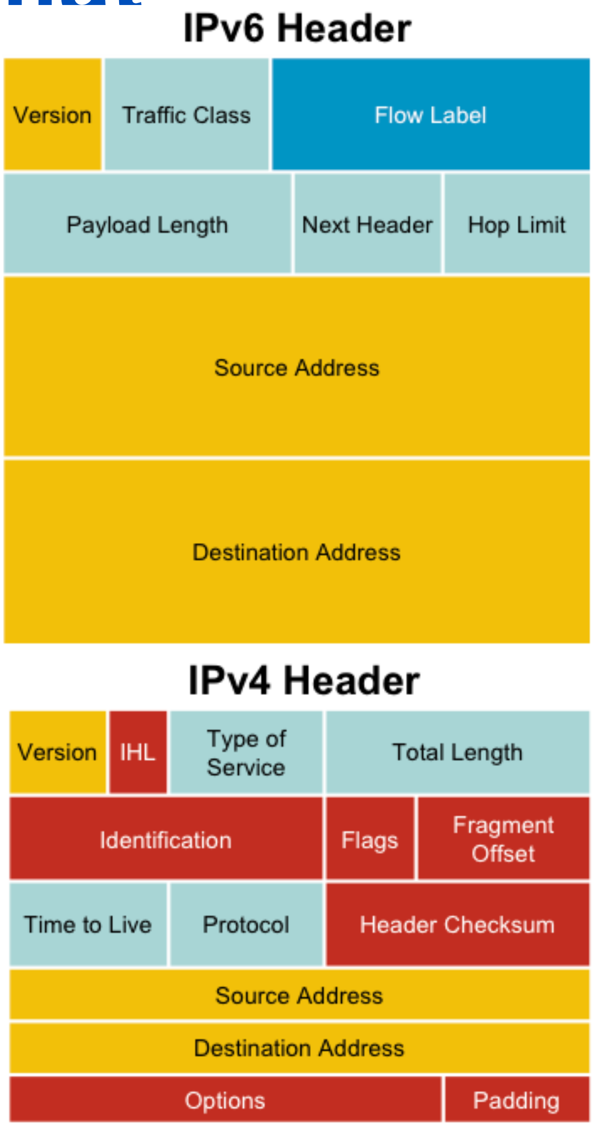
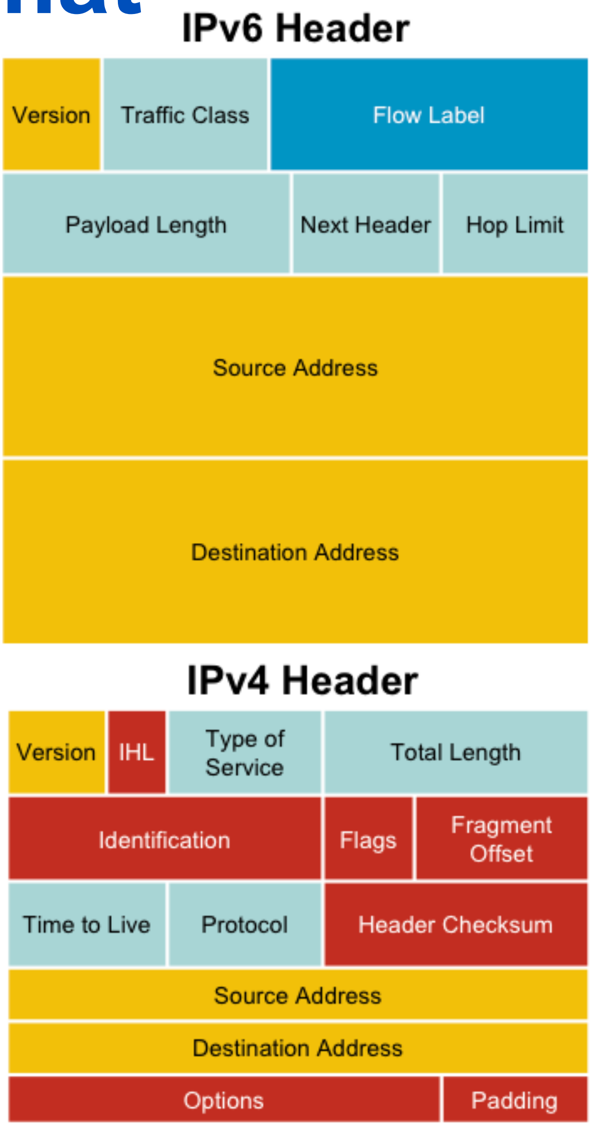


Diagram Source: www.cisco.com

IPv6 Protocol Header Format

- **Payload length**
  - This 16-bit field is similar to the IPv4 Total Length Field, except that with IPv6 the Payload Length field is the length of the data carried after the header, whereas with IPv4 the Total Length Field included the header. 216 = 65536 Octets.
- **Next header**
  - The 8-bit value of this field determines the type of information that follows the basic IPv6 header. It can be a transport-layer packet, such as TCP or UDP, or it can be an extension header. The next header field is similar to the protocol field of IPv4.
- **Hop limit**
  - This 8-bit field defines by a number which count the maximum hops that a packet can remain in the network before it is destroyed. With the IPv4 TLV field this was expressed in seconds and was typically a theoretical value and not very easy to estimate.



IPv6 Extension Header

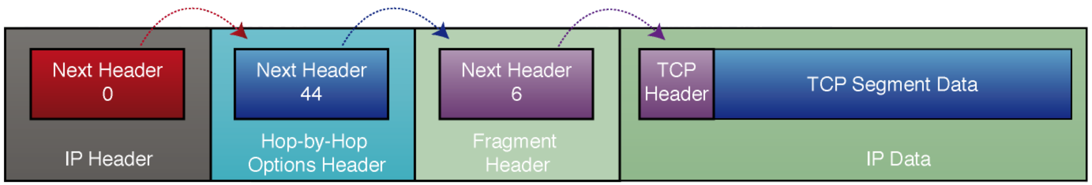
- Adding an optional Extension Header in IPv6 makes it simple to add new features in IP protocol in future without a major re-engineering of IP routers everywhere
- The number of extension headers are not fixed, so the total length of the extension header chain is variable
- The extension header will be placed in between main header and payload in an IPv6 packet

IPv6 Extension Header

- If the Next Header field value (code) is 6, it determines that there is no extension header and the next header field is pointing to TCP header which is the payload of this IPv6 packet
- Code values of Next Header field:
  - 0 Hop-by-hop option
  - 2 ICMP
  - 6 TCP
  - 17 UDP
  - 43 Source routing
  - 44 Fragmentation
  - 50 Encrypted security payload
  - 51 Authentication
  - 59 Null (No next header)
  - 60 Destination option

Link listed Extension Header

IPv6 Datagram With No Extension Headers Carrying TCP Segment



IPv6 Datagram With Two Extension Headers Carrying TCP Segment

- Link listed extension header can be used by simply using next header code value
- Above example use multiple extension header creating link list by using next header code value i.e 0 44 6
- The link list will end when the next header point to transport header i.e next header code 6

Order Of Extension Header

- Source node follow the order:
  - 1. Hop-by-hop
  - 2. Routing
  - 3. Fragment
  - 4. Authentication
  - 5. Encapsulating security payload
  - 6. Destination option
  - 7. Upper-layer
- Order is important because:
  - Only hop-by-hop has to be processed by every intermediate nodes
  - Routing header need to be processed by intermediate routers
  - At the destination fragmentation has to be processed before others
  - This is how it is easy to implement using hardware and make faster processing engine

Fragmentation Handling In IPv6

- Routers handle fragmentation in IPv4 which cause variety of processing performance issues
- IPv6 routers no longer perform fragmentation. IPv6 host use a discovery process [Path MTU Discovery] to determine most optimum MTU size before creating end to end session
- In this discovery process, the source IPv6 device attempts to send a packet at the size specified by the upper IP layers [i.e TCP/Application].
- If the device receives an ICMP packet too big message, it informs the upper layer to discard the packet and to use the new MTU.
- The ICMP packet too big message contains the proper MTU size for the pathway.
- Each source device needs to track the MTU size for each session.

Source: www.cisco.com

MTU Size Guideline

- MTU for IPv4 and IPv6
  - MTU is the largest size datagram that a given link layer technology can support [i.e HDLC]
  - Minimum MTU 68 Octet [IPv4] 1280 Octet [IPV6]
  - Most efficient MTU 576 [IPv4] 1500 [IPv6]
- Important things to remember:
  - Minimum MTU for IPv6 is 1280
  - Most efficient MTU is 1500
  - Maximum datagram size 64k
  - With IPv6 in IPv4 tunnel 1560 [Tunnel Source Only]

IPv6 Header Compression

- IPv6 header size is double then IPv4
- Some time it becomes an issue on limited bandwidth link i.e Radio
- **Robust Header Compression [RoHC]** standard can be used to minimize IPv6 overhead transmission in limited bandwidth link
- RoHC is IETF standard for IPv6 header compression

Overview

- Introduction to IPv6
- IPv6 Protocol Architecture
- **Mobile IPv6 Operation**
- IPv6 Security Features
- IPv6 Addressing and Subnetting
- IPv4 to IPv6 Transition Technologies
- IPv6 Services

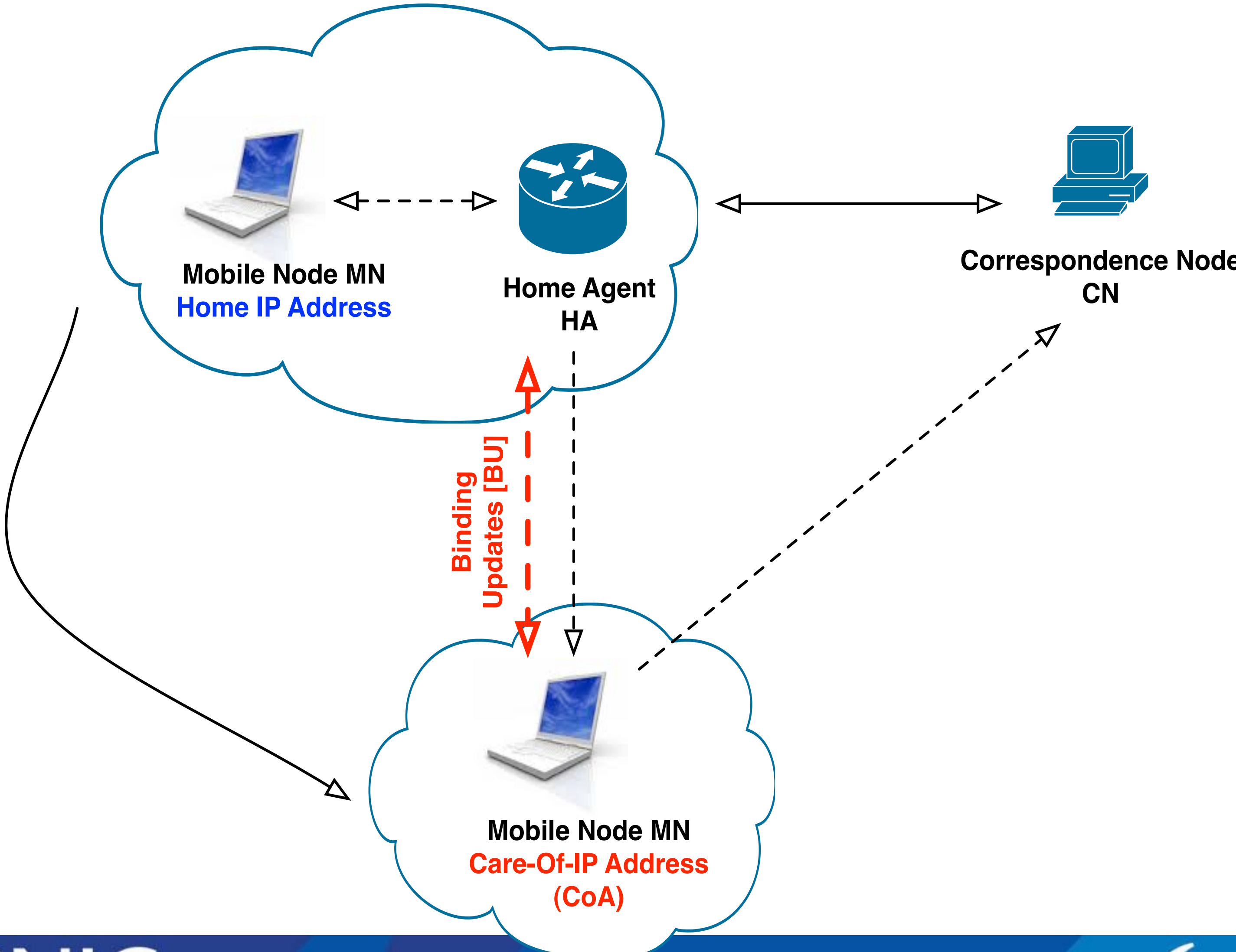
IP Address Mobility

- IP address mobility is a mechanism that will sustain the IP connection even when the IP address change if the device move from one location to other location (subnet)
- IP address mobility is achieved by using Mobile IP
- Mobile IP is designed to work with both IPv4 [RFC3344] and IPv6 [RFC3775]
- Mobile IP operation is optimized for IPv6

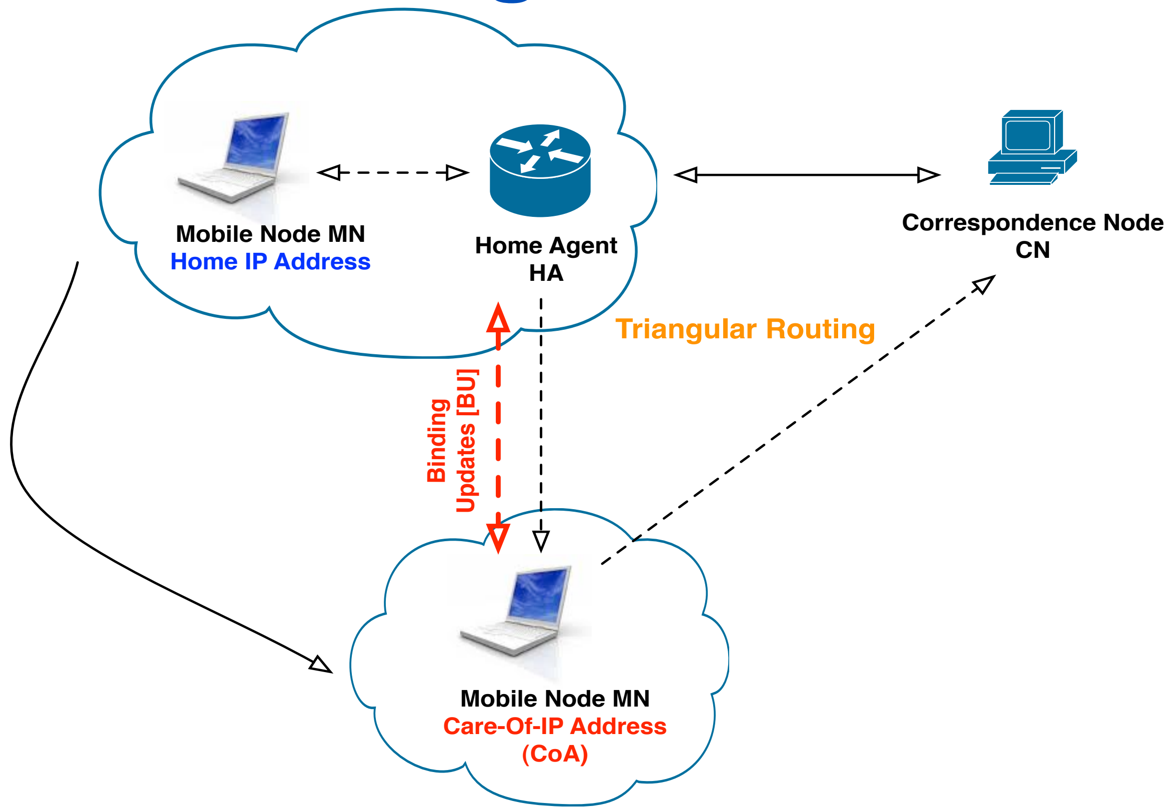
IP Address Mobility Terminology

- Mobile Node [MN]
  - Is the mobile user
- Correspondent Node [CN]
  - Fixed [or may be mobile] user
- Home Agent [HA]
  - Usually a router in home representing MN
- Home IP Address
  - Primary (fixed) IP address of MN
- Care-Of-Address [CoA]
  - Secondary (variable) IP address of MN
- Binding Update [BU]
  - Process to register new IP address to HA [some time CN]

Basic Mobile IP Operation

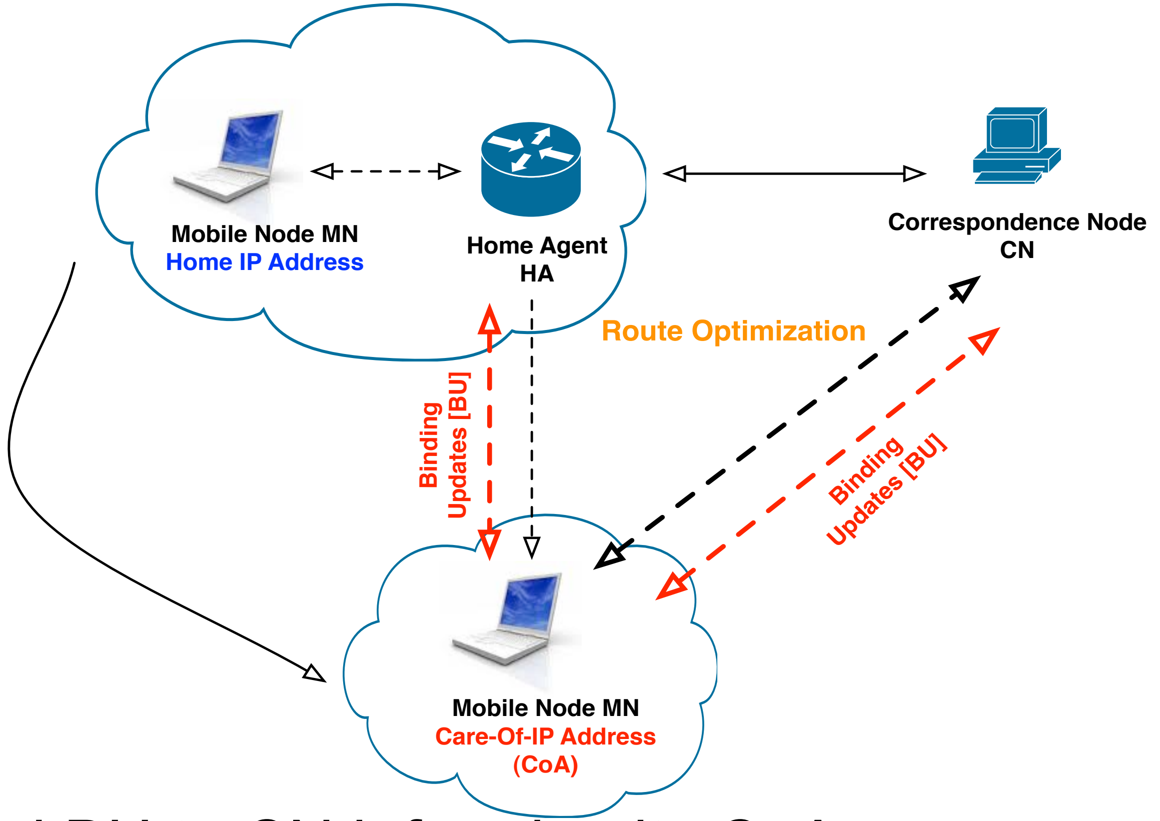


Triangular Routing Issue



- Triangular routing creates delay which affects real time application i.e VoIP, streaming etc

Route Optimization



- MN send BU to CN informing its CoA
- Direct data communication will start between MN and CN

Overview

- Introduction to IPv6
- IPv6 Protocol Architecture
- Mobile IPv6 Operation
- **IPv6 Security Features**
- IPv6 Addressing and Subnetting
- IPv4 to IPv6 Transition Technologies
- IPv6 Services

IPv6 Security Features

- IPsec is mandatory in IPv6
- Since IPsec become part of the IPv6 protocol all node can secure their IP traffic if they have required keying infrastructure
- In build IPsec does not replace standard network security requirement but introduce added layer of security with existing IP network

IPsec Transport and Tunnel Mode

- IPsec has two mode of encapsulation
  - Transport mode

    Provide end to end security between two end station
  - Tunnel mode

    Provide secure connection between two gateway (router). Unencrypted data from end system go through encrypted tunnel provided by the source and destination gateways

IPsec Transport and Tunnel Mode

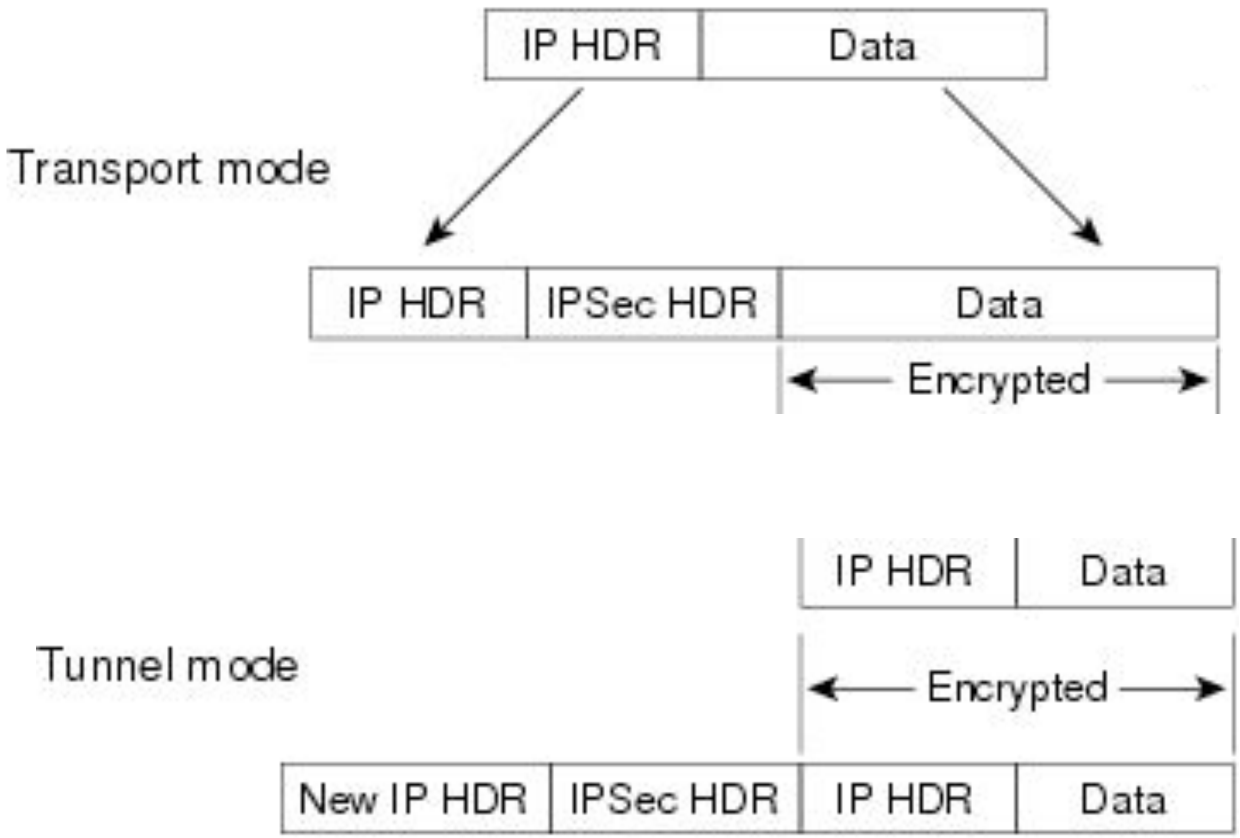


Diagram Source: www.cisco.com

Overview

- Introduction to IPv6
- IPv6 Protocol Architecture
- Mobile IPv6 Operation
- IPv6 Security Features
- **IPv6 Addressing and Subnetting**
- IPv4 to IPv6 Transition Technologies
- IPv6 Services

Allocation And Assignment

- Allocation
  - "A block of address space held by an IR (or downstream ISP) for subsequent allocation or assignment"
    - Not yet used to address any networks
- Assignment
  - "A block of address space used to address an operational network"
    - May be provided to ISP customers, or used for an ISP's infrastructure ('self-assignment')

Allocation and Assignment

APNIC
Allocates
to APNIC Member

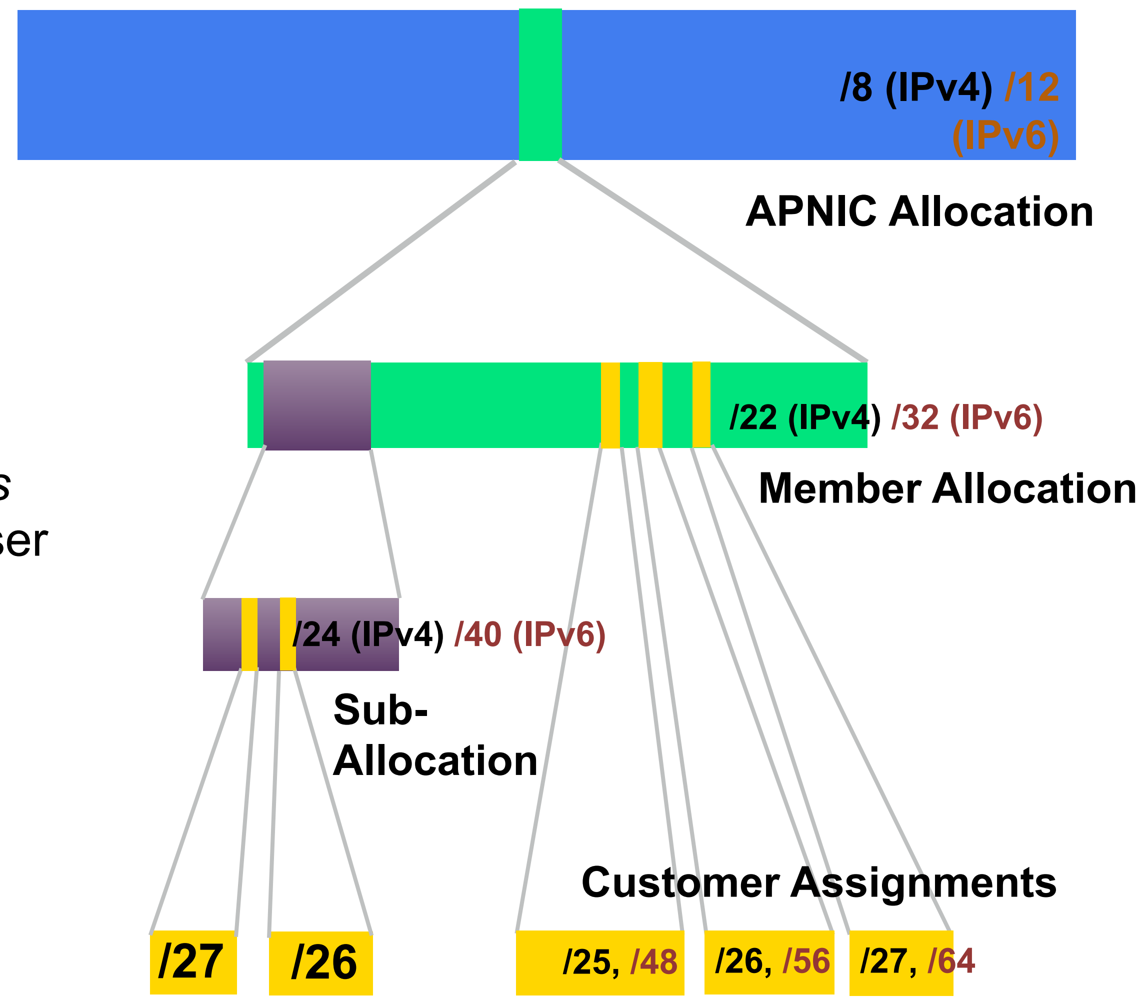
APNIC Member

Allocates
to downstream

Downstream
Assigns
to end-user

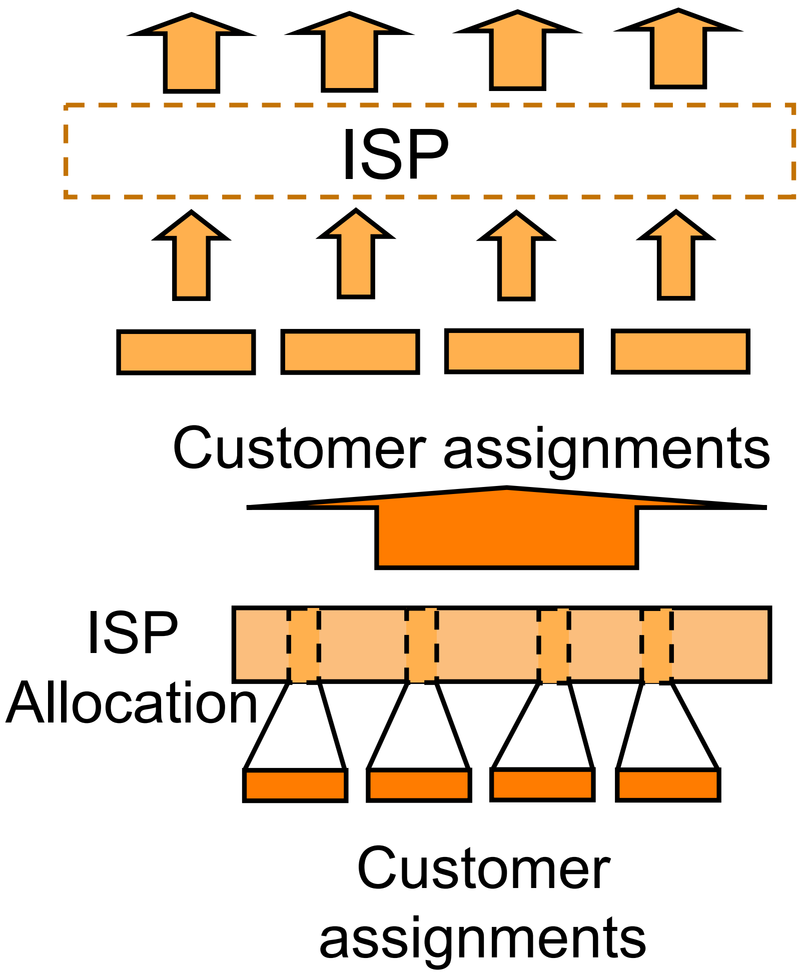
Assigns
to end-user

Customer / End User

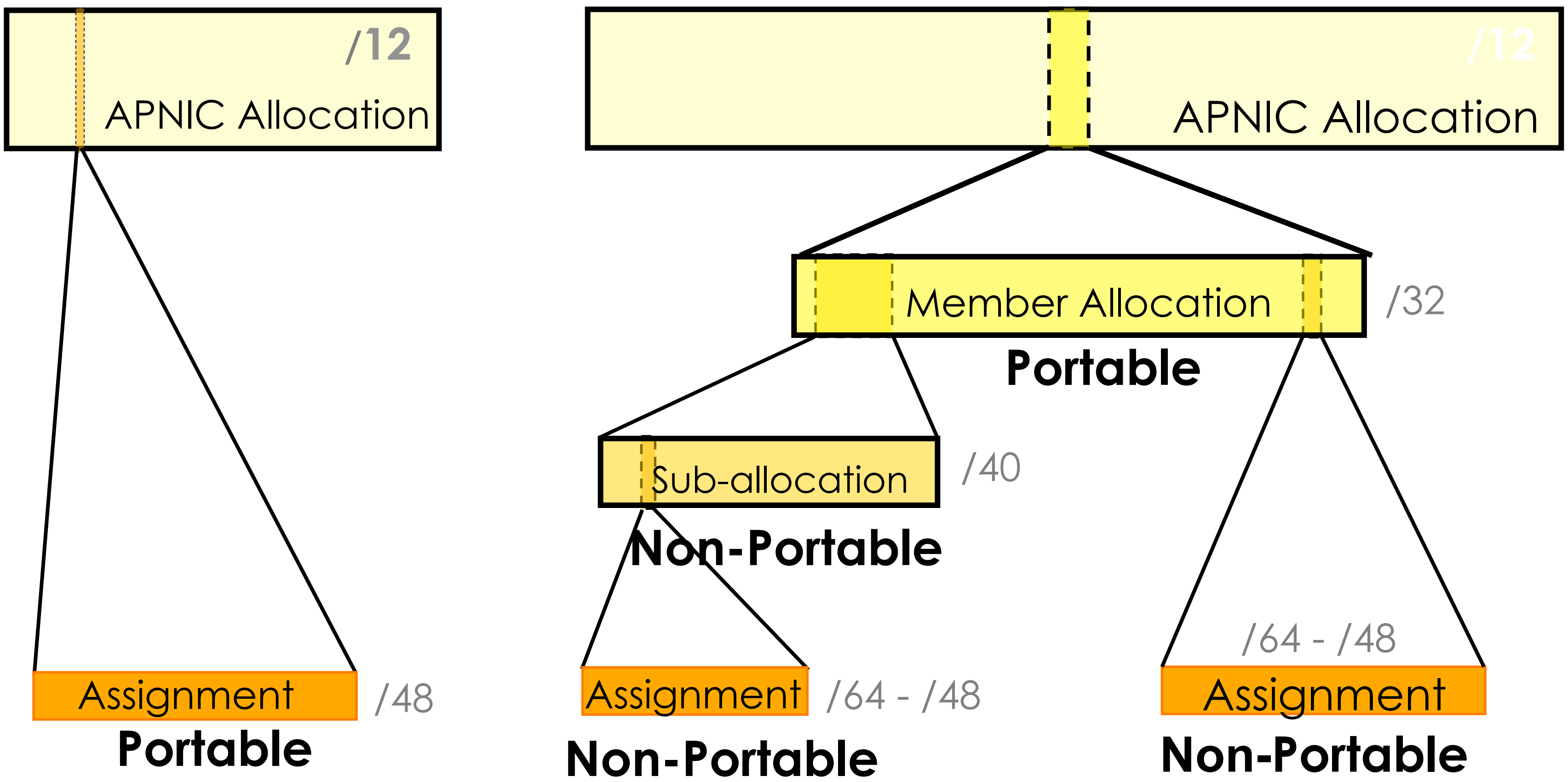


Portable & non-portable

- Portable Assignments
  - Customer addresses independent from ISP
    - Keeps addresses when changing ISP
  - Bad for size of routing tables
  - Bad for QoS: routes may be filtered, flap-dampened
- Non-portable Assignments
  - Customer uses ISP's address space
    - Must renumber if changing ISP
  - Only way to effectively scale the Internet
- Portable allocations
  - Allocations made by APNIC/NIRs



Address Management Hierarchy



Describes "portability" of the address space

Internet Resource Management Objectives

Conservation

- Efficient use of resources
- Based on demonstrated need

Aggregation

- Limit routing table growth
- Support provider-based routing

Registration

- Ensure uniqueness
- Facilitate trouble shooting

Uniqueness, fairness and consistency

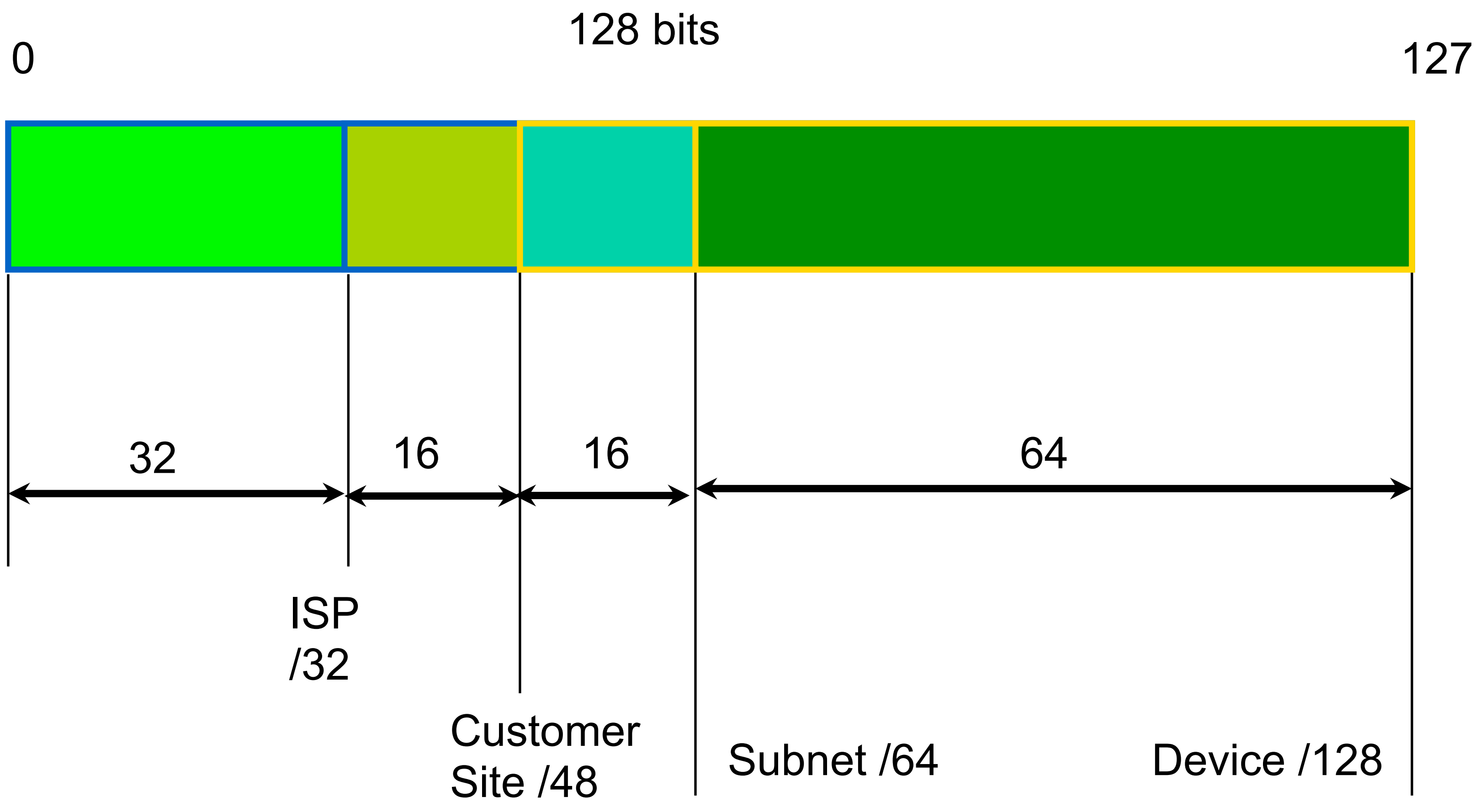
Initial IPv6 Allocation

- To qualify for an initial allocation of IPv6 address space, an organization must:
  - Not be an end site (must provide downstream services)
  - Plan to provide IPv6 connectivity to organizations to which it will make assignments
- Meet one of the two following criteria:
  - Have a plan for making at least 200 assignments to other organizations within two years OR
  - Be an existing ISP with IPv4 allocations from an APNIC or an NIR, which will make IPv6 assignments or sub-allocations to other organizations and announce the allocation in the inter-domain routing system within two years

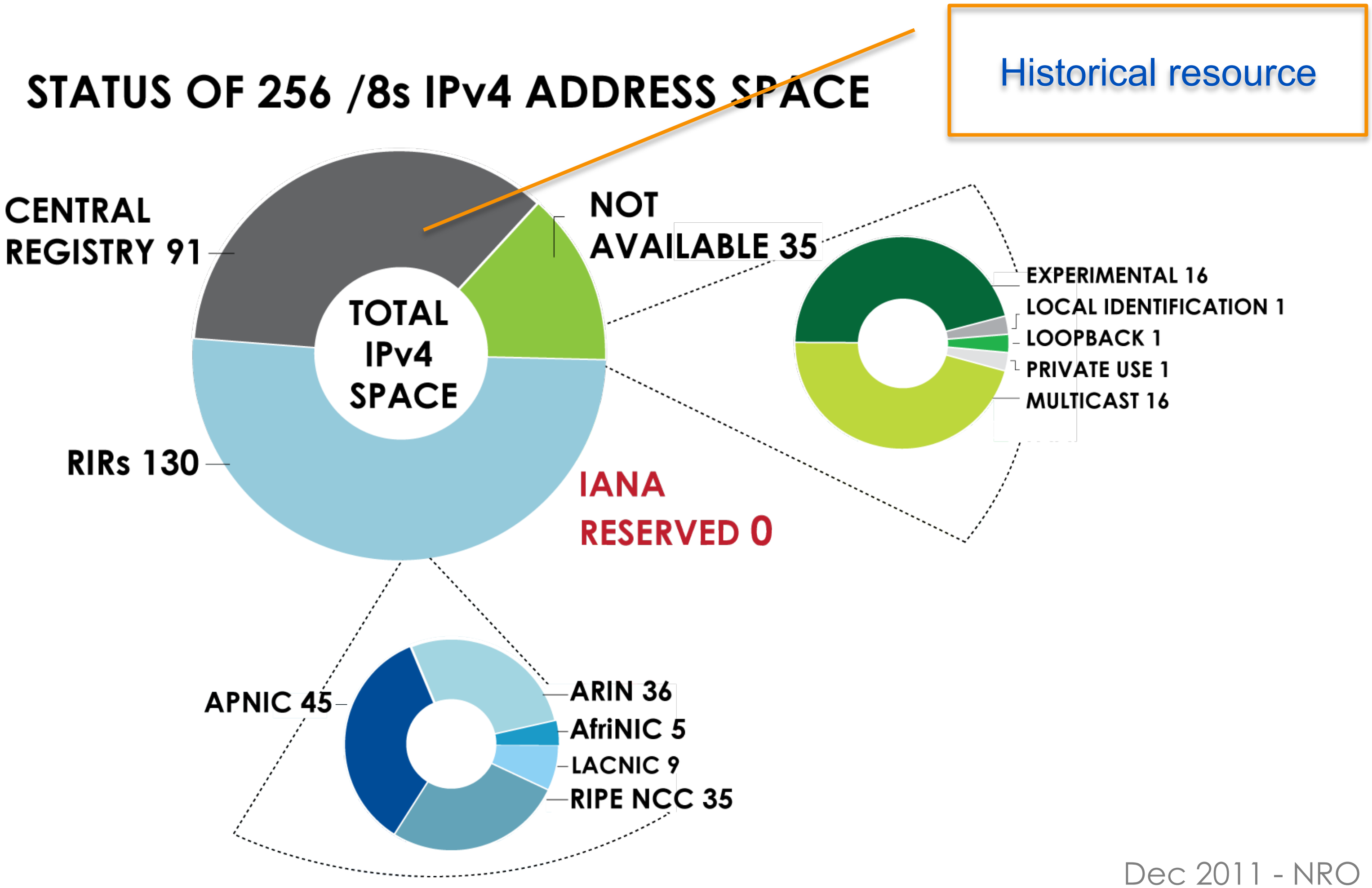
“One Click” IPv6 Policy

- Members with IPv4 holdings can click the button in MyAPNIC to instantly receive their IPv6 block
  - No forms to fill out!
  - “Get your IPv6 addresses” icon in the main landing page at MyAPNIC
- A Member that has an IPv4 allocation is eligible for a /32
- A Member that has an IPv4 assignment is eligible for a /48

IPv6 Addressing Structure

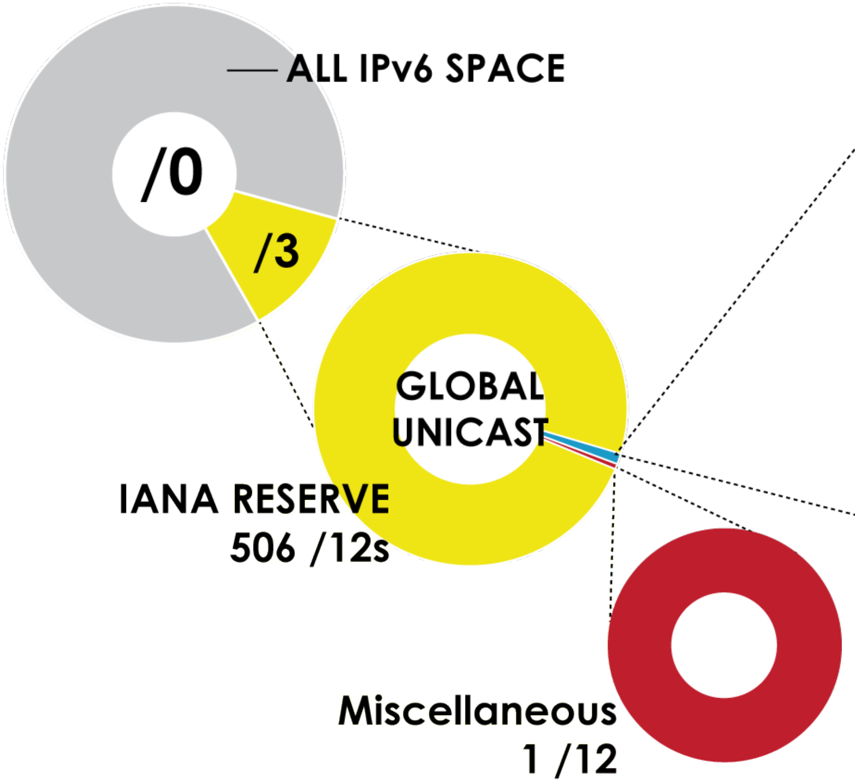


Historical Resources



Dec 2011 - NRO

IPv6 Address Space

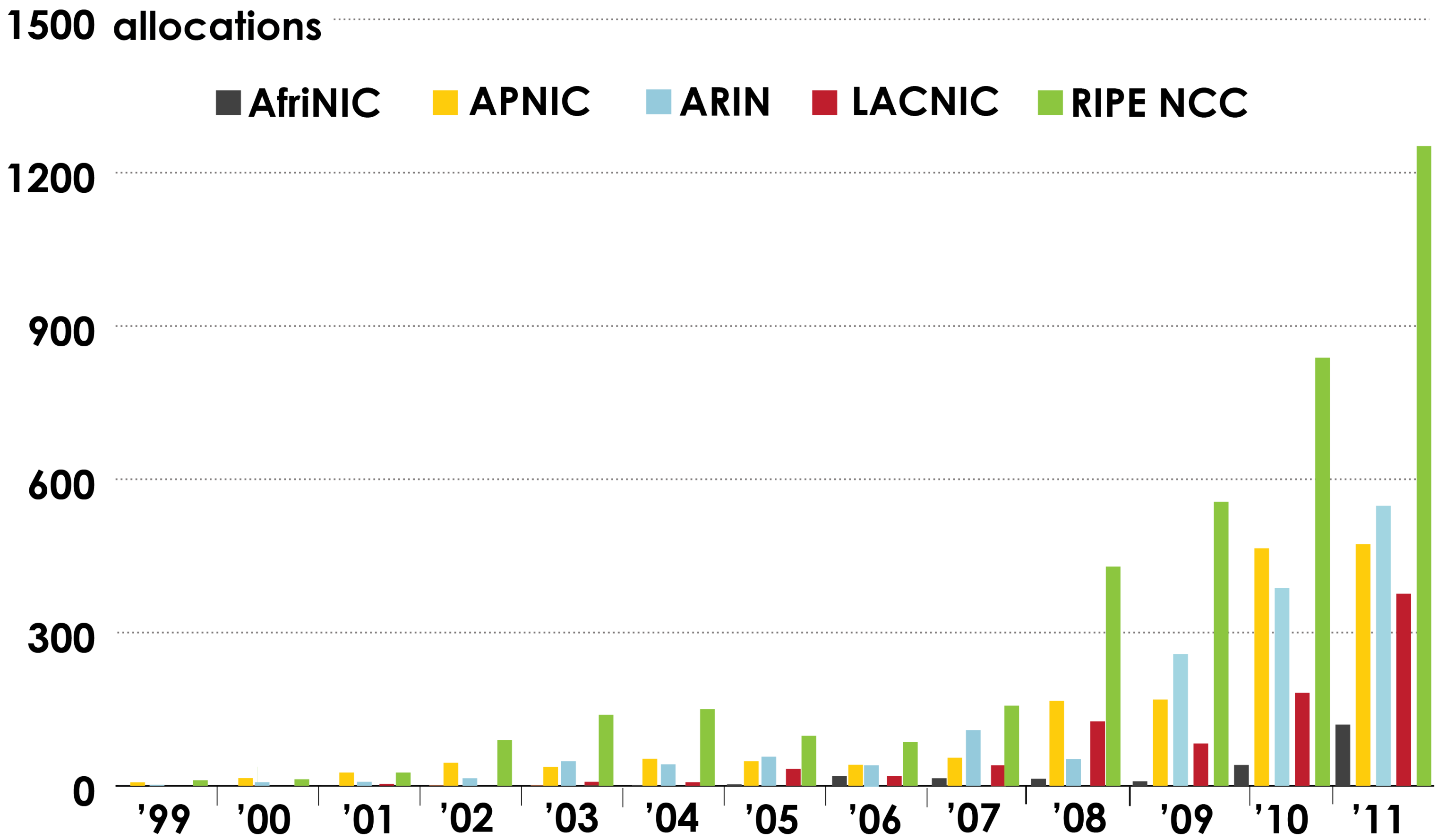


RIRs 5 /12s (October 2006)

RIR	IPv6 ADDRESS
AfriNIC	2C00:0000::/12
APNIC	2400:0000::/12
ARIN	2600:0000::/12
LACNIC	2800:0000::/12
RIPE NCC	2A00:0000::/12

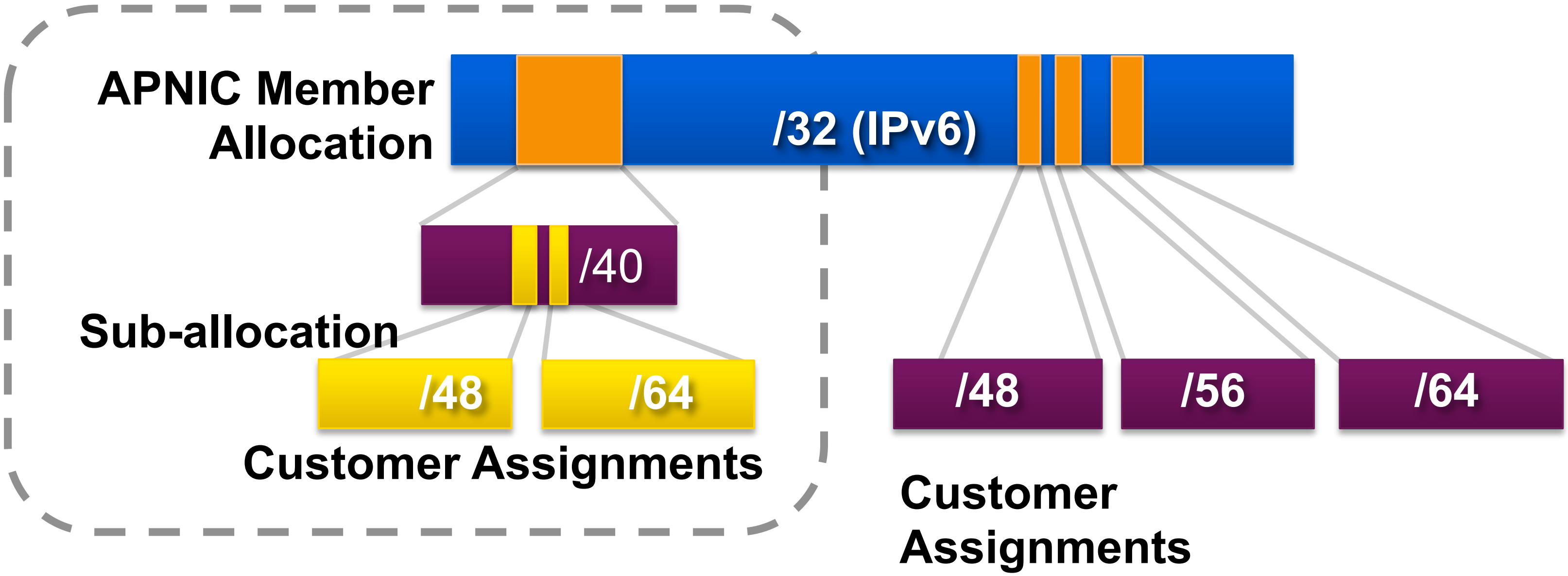
Dec 2011 - NRO

IPv6 Allocations RIRs to LIRs



Dec 2011 - NRO

Sub-allocations



- No specific policy for LIRs to allocate space to subordinate ISPs
- All /48 assignments to end sites must be registered
- Second Opinion applies
  - Must submit a second opinion request for assignments more than /48

Sub-allocation Guidelines

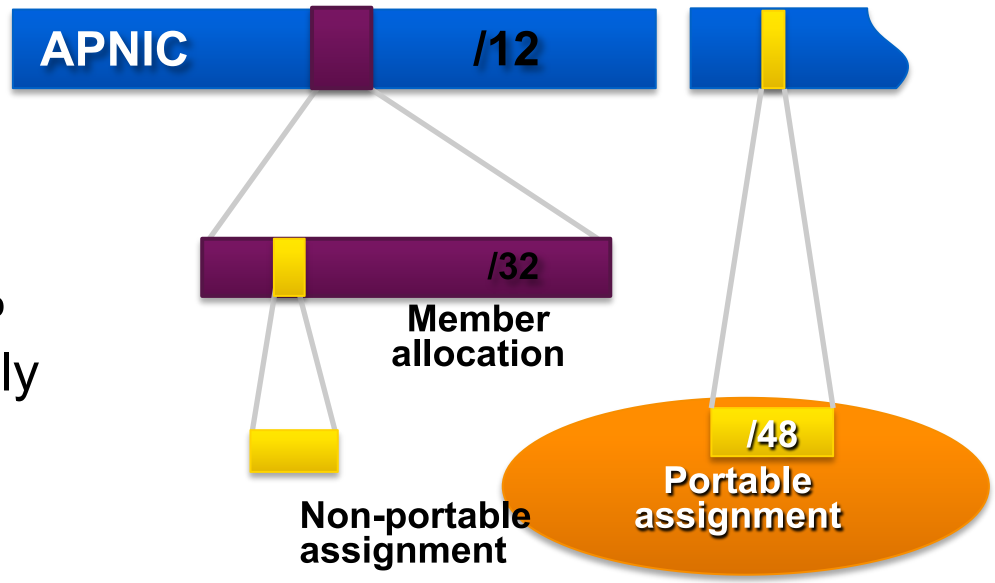
- Sub-allocate cautiously
  - Only allocate or assign what the customer has demonstrated a need for
  - Seek APNIC advice if in doubt
- Efficient assignments
  - Member is responsible for overall utilisation
- Database registration (WHOIS Db)
  - Sub-allocations & assignments must be registered in the whois db

IPv6 Assignment Policy

- Assignment address space size
  - Minimum of /64 (only 1 subnet), Normal maximum of /48, Larger end-site assignment can be justified
- In typical deployments today
  - Several ISPs gives small customers a /56 or a /60 and Single LAN end sites a /64, e.g.,
    - /64 if end-site will ever only be a LAN
    - /60 for small end-sites (e.g. consumer)
    - /56 for medium end-sites (e.g. small business)
    - /48 for large end-sites
- Assignment of multiple /48s to a single end site
  - Documentation must be provided
  - Will be reviewed at the RIR/NIR level
- Assignment to operator's infrastructure
  - /48 per PoP as the service infrastructure of an IPv6 service operator

Portable Assignments for IPv6

- For (small) organisations who require a portable assignment for multi-homing purposes
  - The current policy allows for IPv6 portable assignment to end-sites
  - Size: /48, or a shorter prefix if the end site can justify it
  - To be multi-homed within 1 month
  - Demonstrate need to use 25% of requested space immediately and 50% within a year



IXP IPv6 Assignment Policy

- Criteria
  - Demonstrate ‘open peering policy’
  - 3 or more peers
- Portable assignment size: /48
  - All other needs should be met through normal processes
  - /64 holders can “upgrade” to /48
    - Through NIRs/ APNIC
    - Need to return /64

Portable Critical Infrastructure Assignments

- What is Critical Internet Infrastructure?
  - Domain Registry Infrastructure
    - Operators of Root DNS, gTLD, and ccTLD
  - Address Registry Infrastructure
    - IANA, RIRs & NIRs
- Why a specific policy ?
  - Protect stability of core Internet function
- Assignment sizes:
  - IPv6: /32

IPv6 Utilisation

- Utilisation determined from end site assignments
  - ISP responsible for registration of all /48 assignments
  - Intermediate allocation hierarchy not considered
- Utilisation of IPv6 address space is measured differently from IPv4
  - Use HD ratio to measure
- Subsequent allocation may be requested when IPv6 utilisation requirement is met

Subsequent Allocation

- Must meet **HD = 0.94** utilisation requirement of previous allocation (subject to change)
- Other criteria to be met
  - Correct registrations (all /48s registered)
  - Correct assignment practices etc
- Subsequent allocation results in a doubling of the address space allocated to it
  - Resulting in total IPv6 prefix is 1 bit shorter
  - Or sufficient for 2 years requirement

HD Ratio

- The HD ratio threshold is
  - HD = log (/56 units assigned) / log (16,777,216)
  - 0.94 = 6,183,533 x /56 units
- Calculation of the HD ratio
  - Convert the assignment size into equivalent /56 units
    - Each /48 end site = 256 x /56 units
    - Each /52 end site = 16 x /56 units
    - Each /56 end site = 1 x /56 units
    - Each /60 end site = 1/16 x /56 units
    - Each /64 end site = 1/256 x /56 units

IPv6 utilisation (HD = 0.94)

- Percentage utilisation calculation

IPv6 Prefix	Site Address Bits	Total site address in /56s	Threshold (HD ratio 0.94)	Utilisation %
/42	14	16,384	9,153	55.9%
/36	20	1,048,576	456,419	43.5%
/35	21	2,097,152	875,653	41.8 %
/32	**24**	**16,777,216**	**6,185,533**	**36.9%**
/29	27	134,217,728	43,665,787	32.5 %
/24	32	4,294,967,296	1,134,964,479	26.4 %
/16	40	1,099,511,627,776	208,318,498,661	18.9 %

RFC 3194: “In a hierarchical address plan, as the size of the allocation increases, the density of assignments will decrease.”

IPv6 Addressing

- An IPv6 address is 128 bits long
- So the number of addresses are **2^128** = 340282366920938463463374607431768211455
- In hex, 4 bits (also called a 'nibble') is represented by a hex digit

2001:DC0:A910::

nibbles

1010 | 1001 | 0001 | 0000

IPv6 Addressing

2001:0DB8:DEAD:BEEF:1AB6:503F:A804:71D9

0010 0000 0000 0001	0000 1101 1011 1000
1101 1110 1010 1101	1011 1110 1110 1111
0001 1010 1011 0110	1001 0000 0011 1111
1010 1000 0000 0100	0111 0001 1101 1001

128 bits is reduced down to 32 hex digits

IPv6 Address Representation

RFC 5952

- Hexadecimal values of eight 16 bit fields
  - X:X:X:X:X:X:X:X (X=16 bit number, ex: A2FE)
  - 16 bit number is converted to a 4 digit hexadecimal number
  - Case insensitive
- Example:
  - FE38:DCE3:124C:C1A2:BA03:6735:EF1C:683D
  - Abbreviated form of address

    FE80:0023:**0000:0000:0000**:036E:1250:2B00

    →FE80:23:**0:0:0**:36E:1250:2B00

    →FE80:23:**::**36E:1250:2B00

    (Null value can be used only once)

Leading zeroes

Groups of zeroes

Double colons

IPv6 Address Representation (2)

- Double colons (::) representation
  - RFC5952 recommends that the rightmost set of :0: be replaced with :: for consistency
    - 2001:db8:0:2f::5 rather than 2001:db8::2f:0:0:0:5
- In a URL, it is enclosed in brackets (RFC3986)
  - http://[2001:db8:4f3a::206:ae14]:8080/index.html
  - Cumbersome for users, mostly for diagnostic purposes
  - Use fully qualified domain names (FQDN)
- Prefix Representation
  - Representation of prefix is just like IPv4 CIDR
  - In this representation, you attach the prefix length
  - IPv6 address is represented as:
    - 2001:db8:12::/40

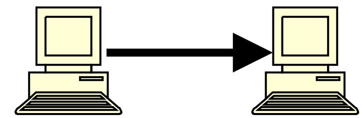
Exercise

1. 2001:0db8:0000:0000:0000:0000:0000:0000
2. 2001:0db8:0000:0000:d170:0000:1000:0ba8
3. 2001:0db8:0000:0000:00a0:0000:0000:10bc
4. 2001:0db8:0fc5:007b:ab70:0210:0000:00bb

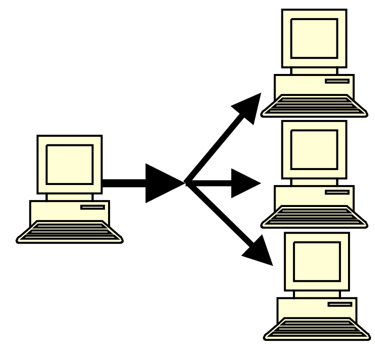
IPv6 Addressing Model

RFC 4291

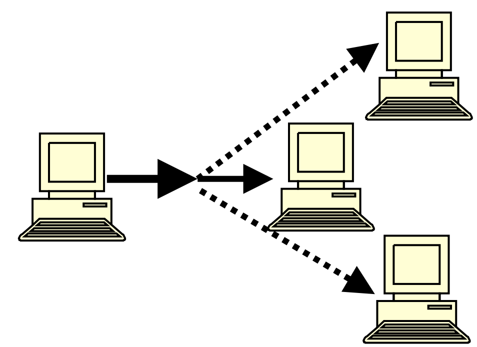
- Unicast
  - An identifier for a single interface



- Multicast
  - An identifier for a group of nodes



- Anycast
  - An identifier for a set of interfaces



Unicast address

- Address given to interface for communication between host and router
  - Global unicast address currently delegated by IANA

001 FP 3bits	Global routing prefix 45 bits	Subnet ID 16 bits	Interface ID 64 bits

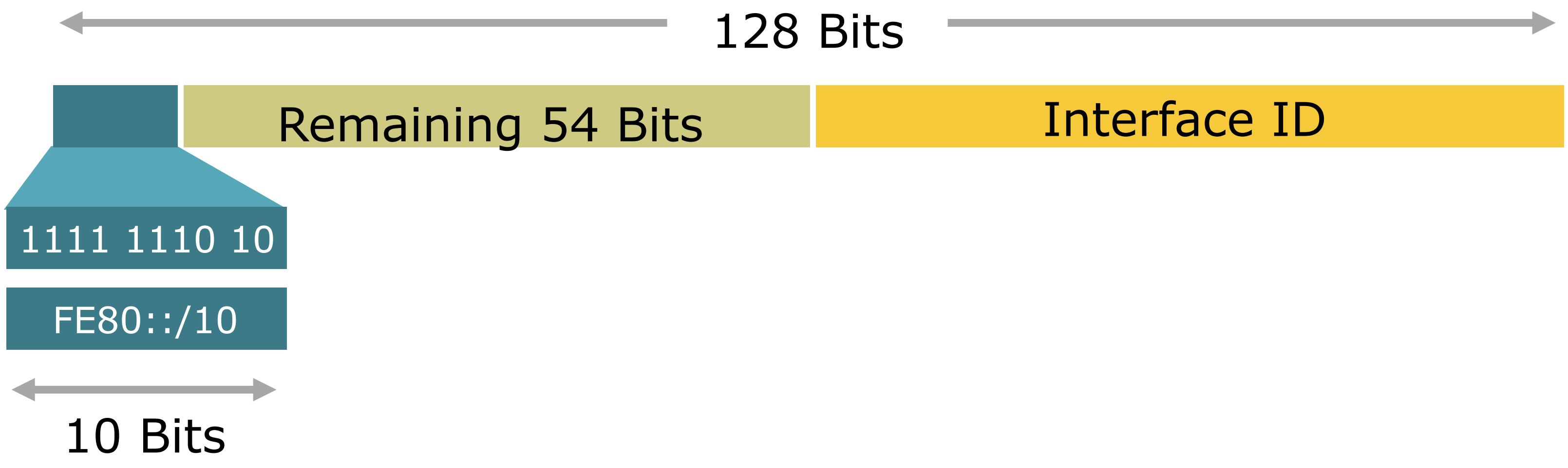
  - Local use unicast address
    - Link-local address (starting with FE80::)

1111111010 10 bits	000…….0000 54 bits	Interface ID 64 bits

Local Addresses With Network Prefix

- Link Local Address
  - A special address used to communicate within the local link of an interface (i.e. anyone on the link as host or router)
  - The address in the packet destination would never pass through a router (local scope)
  - Mandatory address - automatically assigned as soon as IPv6 is enabled
  - **fe80::/10**

Local Addresses With Network Prefix



- Remaining 54 bits could be Zero or any manual configured value

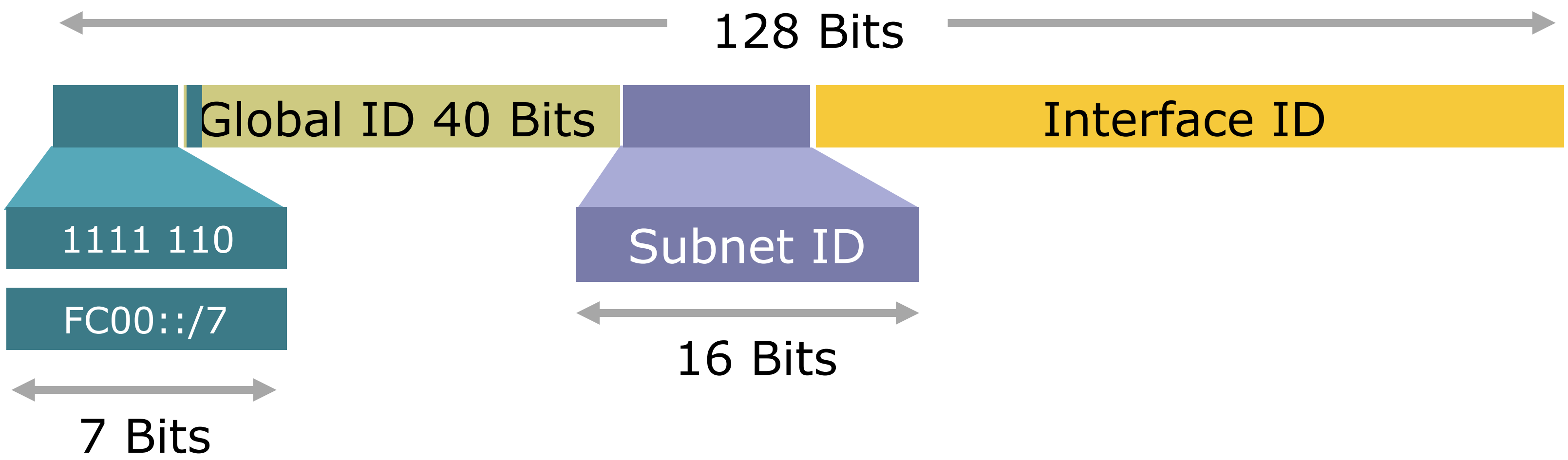
Local Addresses With Network Prefix

- Site Local Address
  - Addresses similar to the RFC 1918 / private address like in IPv4
  - **fec0::/10**
- This address type is now deprecated by RFC 3879 because of lack of uniqueness
- Still used in test lab

Local Addresses With Network Prefix

- Unique Local IPv6 Unicast Address
  - Addresses similar to the RFC 1918 (private address) in IPv4
  - Ensures uniqueness
  - A part of the prefix (40 bits) are generated using a pseudo-random algorithm and it's improbable that two generated ones are equal
  - **fc00::/7**
  - Example webtools to generate ULA prefix
    - http://www.sixxs.net/tools/grh/ula/
    - http://www.goebel-consult.de/ipv6/createLULA
  - RFC 4193

Local Addresses With Network Prefix



- Unique-Local Addresses Used For:
  - Local communications & inter-site VPNs
  - Local devices such as printers, telephones, etc
  - Site Network Management systems connectivity
- Not routable on the Internet

Global Addresses With Network Prefix

- IPV6 Global Unicast Address
  - Global Unicast Range: 0010 2000::/3
    0011 3FFF:FFF:…:FFFF/3
  - All five RIRs are given a /12 from the /3 to further distribute within the RIR region

APNIC	2400:0000::/12
ARIN	2600:0000::/12
AfriNIC	2C00:0000::/12
LACNIC	2800:0000::/12
Ripe NCC	2A00:0000::/12

Global Addresses With Network Prefix

- 6to4 Addresses
  - **2002::/16**
  - Designed for a special tunneling mechanism [RFC 3056] to connect IPv6 Domains via IPv4 Clouds
  - Automatic tunnel transition Mechanisms for IPv6 Hosts and Routers
  - Need 6to4 relay routers in ISP network

Examples and Documentation Prefix

- Two address ranges are reserved for examples and documentation purpose by RFC 3849
  - For example 3fff:ffff::/32
  - For documentation 2001:0DB8::/32

Special addresses

- The unspecified address
  - A value of 0:0:0:0:0:0:0:0 (::)
  - It is comparable to 0.0.0.0 in IPv4
- The loopback address
  - It is represented as 0:0:0:0:0:0:0:1 (::1)
  - Similar to 127.0.0.1 in IPv4

Addresses Without a Network Prefix

• Loopback	::1/128
• Unspecified Address	::/128
• IPv4-mapped IPv6 address	::ffff/96 [a.b.c.d]
• IPv4-compatible IPv6 address	::/96 [a.b.c.d]

IPv6 Address Space

IPv6 Prefix	Allocation	RFC
0000::/8	Reserved by IETF	RFC 4291
2000::/3	Global Unicast	RFC 4291
FC00::/7	Unique Local Address	RFC 4193
FE80::/10	Link Local Unicast	RFC 4291
FEC0::/10	Reserved by IETF	RFC 3879
FF00::/8	Multicast	RFC 4291
2002::/16	6to4	RFC3056

http://www.iana.org/assignments/ipv6-address-space/ipv6-address-space.xml

Subnetting

- Network engineers must have a solid understanding of subnetting
  - Important for address planning
- IPv6 subnetting is similar (if not exactly the same) as IPv4 subnetting
- Note that you are working on hexadecimal digits rather than binary
  - 0 in hex = 0000 in binary
  - 1 in hex = 0001 in binary

Subnetting (Example)

- Provider A has been allocated an IPv6 block

  2001:DB8::/32

- Provider A will delegate /48 blocks to its customers
- Find the blocks provided to the first 4 customers

Subnetting (Example)

Original block: **2001:0DB8::/32**

Rewrite as a /48 block: **2001:0DB8:0000:/48**

This is your network prefix!

How many /48 blocks are there in a /32?

$$\frac{/32}{/48} = \frac{2^{128-32}}{2^{128-48}} = \frac{2^{96}}{2^{80}} = 2^{16}$$

Find only the first 4 /48 blocks…

Subnetting (Example)

Start by manipulating the LSB of your network prefix – write in BITS

2001:0DB8:0000::/48

In bits

2001:0DB8:	0000 0000 0000 0000	::/48	2001:0DB8:0000::/48
2001:0DB8:	0000 0000 0000 0001	::/48	2001:0DB8:0001::/48
2001:0DB8:	0000 0000 0000 0010	::/48	2001:0DB8:0002::/48
2001:0DB8:	0000 0000 0000 0011	::/48	2001:0DB8:0003::/48

Then write back into hex digits

Exercise 1.1: IPv6 subnetting

- Identify the first four /64 address blocks out of 2001:DB8:0::/48
  1. ______________________
  2. ______________________
  3. ______________________
  4. ______________________

Exercise 1.2: IPv6 subnetting

- Identify the first four /36 address blocks out of 2406:6400::/32
    1. ______________________
    2. ______________________
    3. ______________________
    4. ______________________

Exercise 1.3: IPv6 subnetting

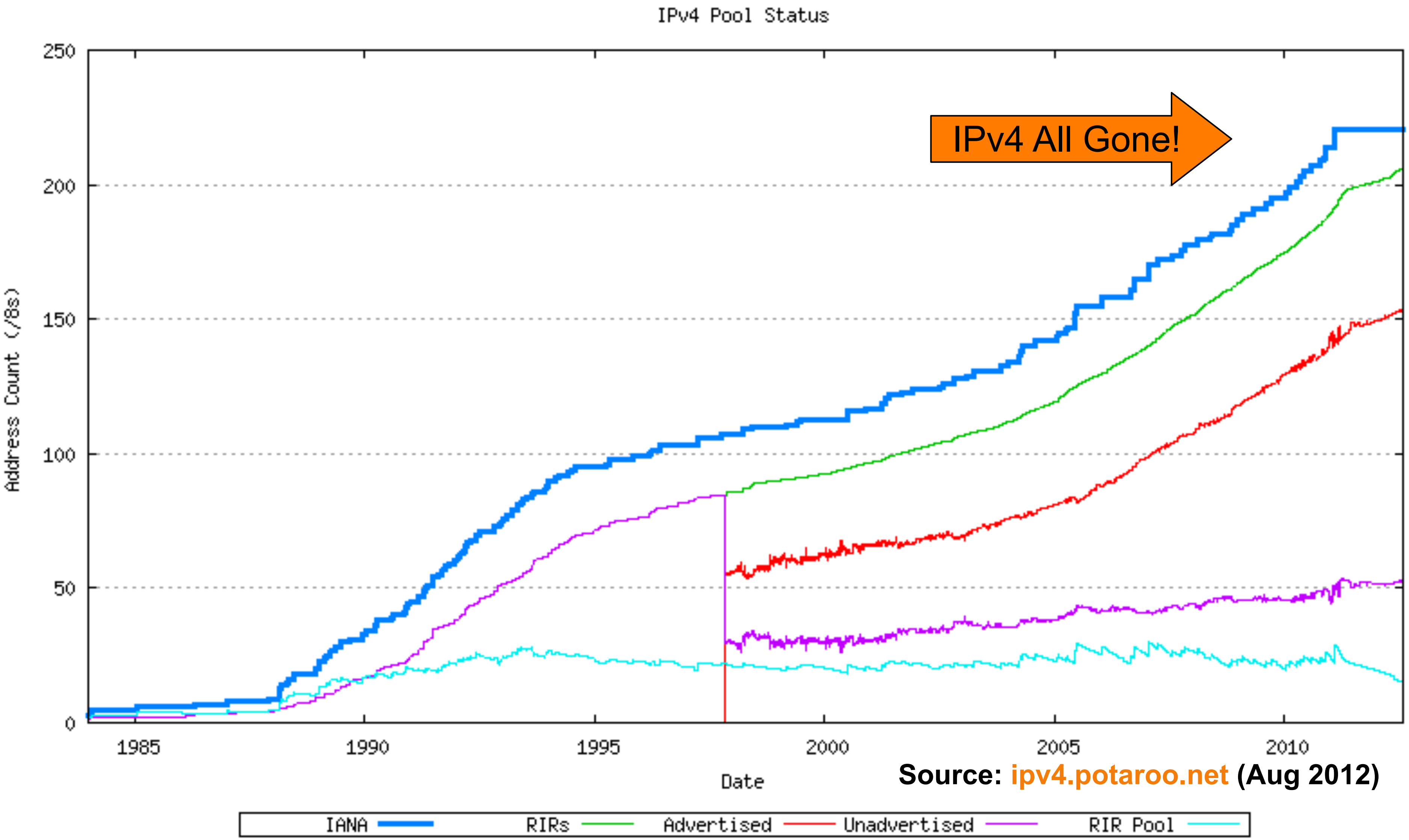
- Identify the first six /35 address blocks out of 2406:6400::/32

1. ______________________
2. ______________________
3. ______________________
4. ______________________
5. ______________________
6. ______________________

Overview

- Introduction to IPv6
- IPv6 Protocol Architecture
- Mobile IPv6 Operation
- IPv6 Security Features
- IPv6 Addressing and Subnetting
- **IPv4 to IPv6 Transition Technologies**
- IPv6 Services

“The times, They are a’ changin’”

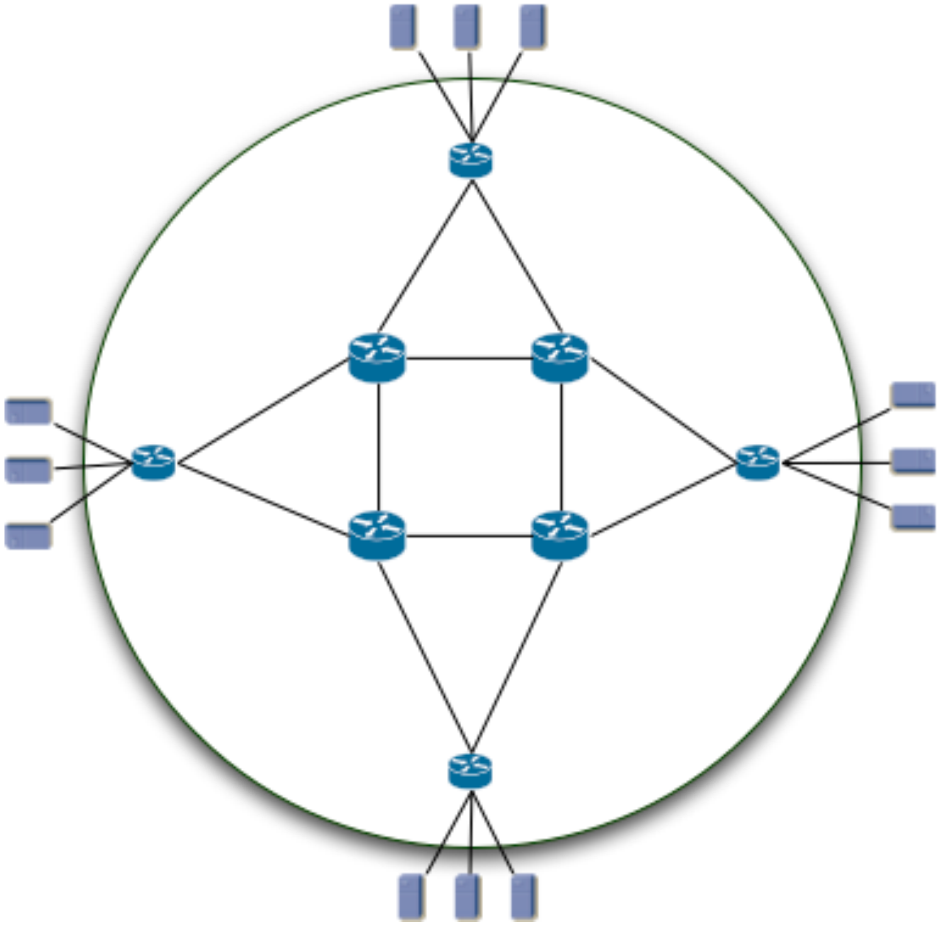


Source: ipv4.potaroo.net (Aug 2012)

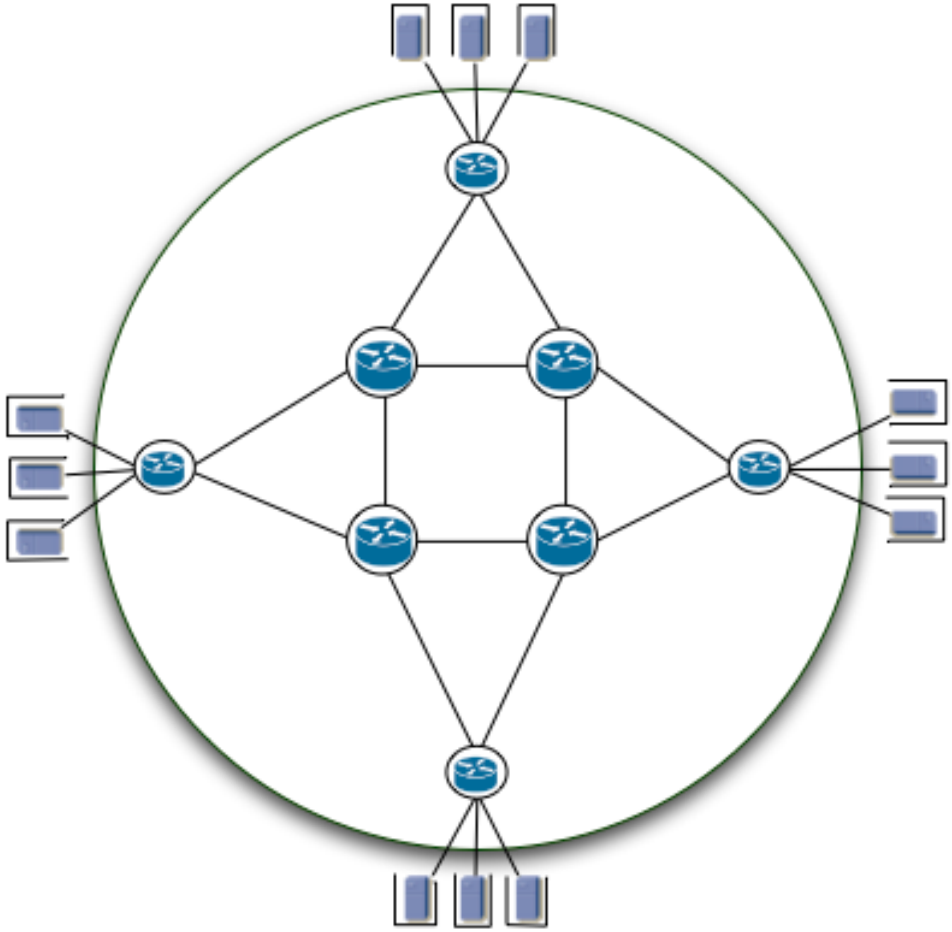
IETF Working Groups

- "v6ops"
  - Define the processes by which networks can be transitioned from IPv4 to IPv6
  - www.ietf.org/dyn/wg/charter/v6ops-charter.html
- "behave"
  - Designs solutions for the IPv4 to IPv6 translations scenarios
  - www.ietf.org/dyn/wg/charter/behave-charter.html
- "softwires"
  - Specifies the standardisation of discovery, control and encapsulation methods for connecting IPv4 networks across IPv6 networks and IPv6 networks across IPv4 networks in a way that will encourage multiple, inter-operable implementations
  - www.ietf.org/dyn/wg/charter/softwire-charter.html

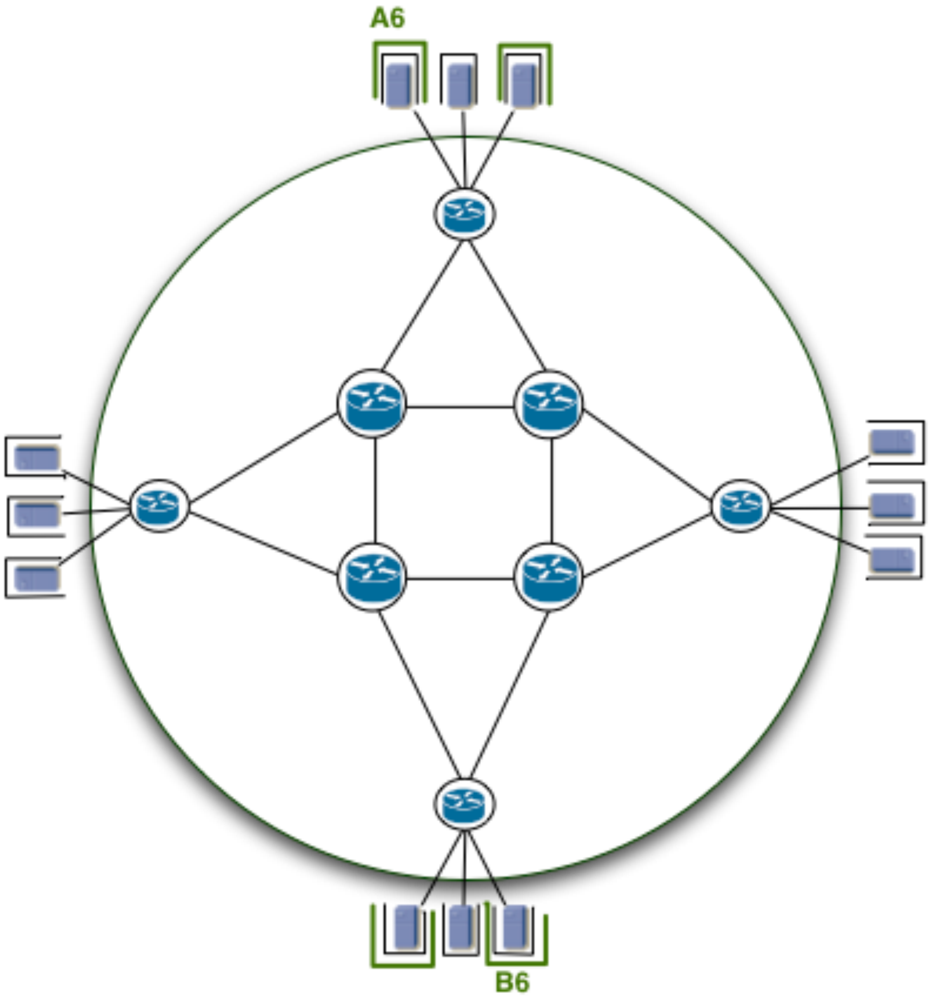
Transition Concept



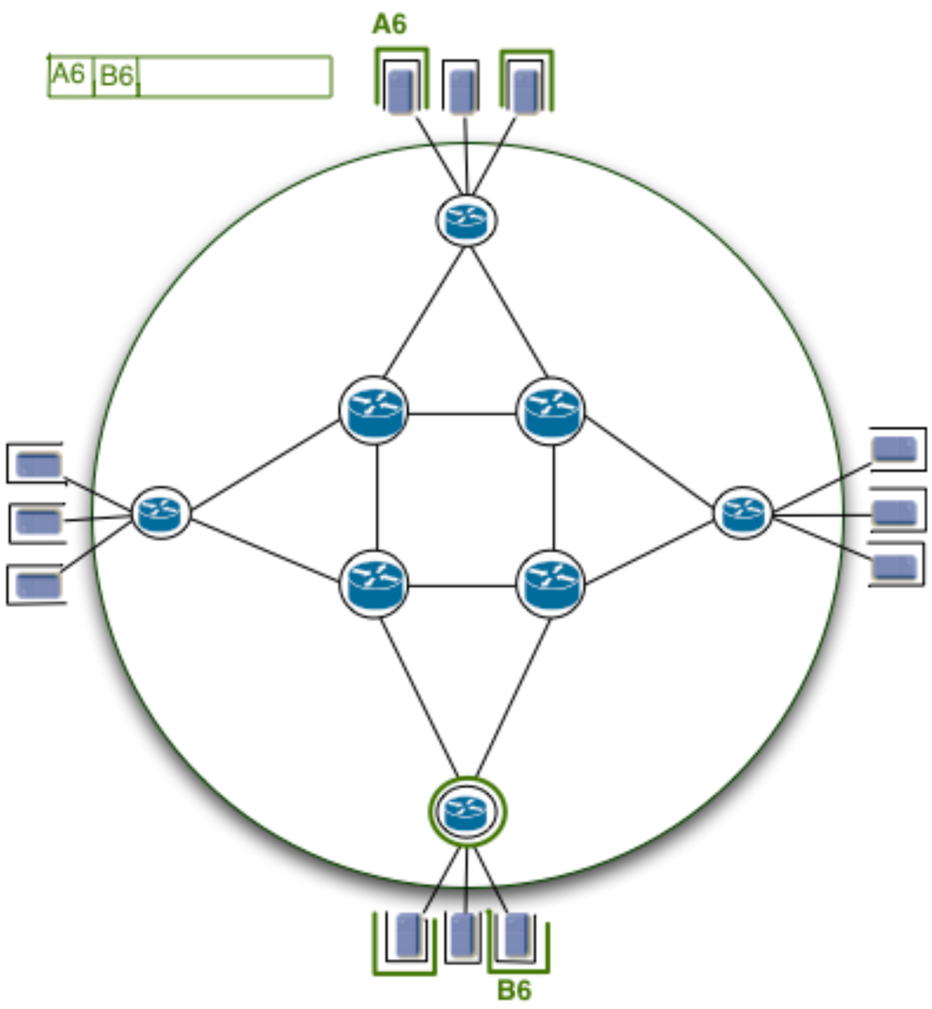
Transition Concept



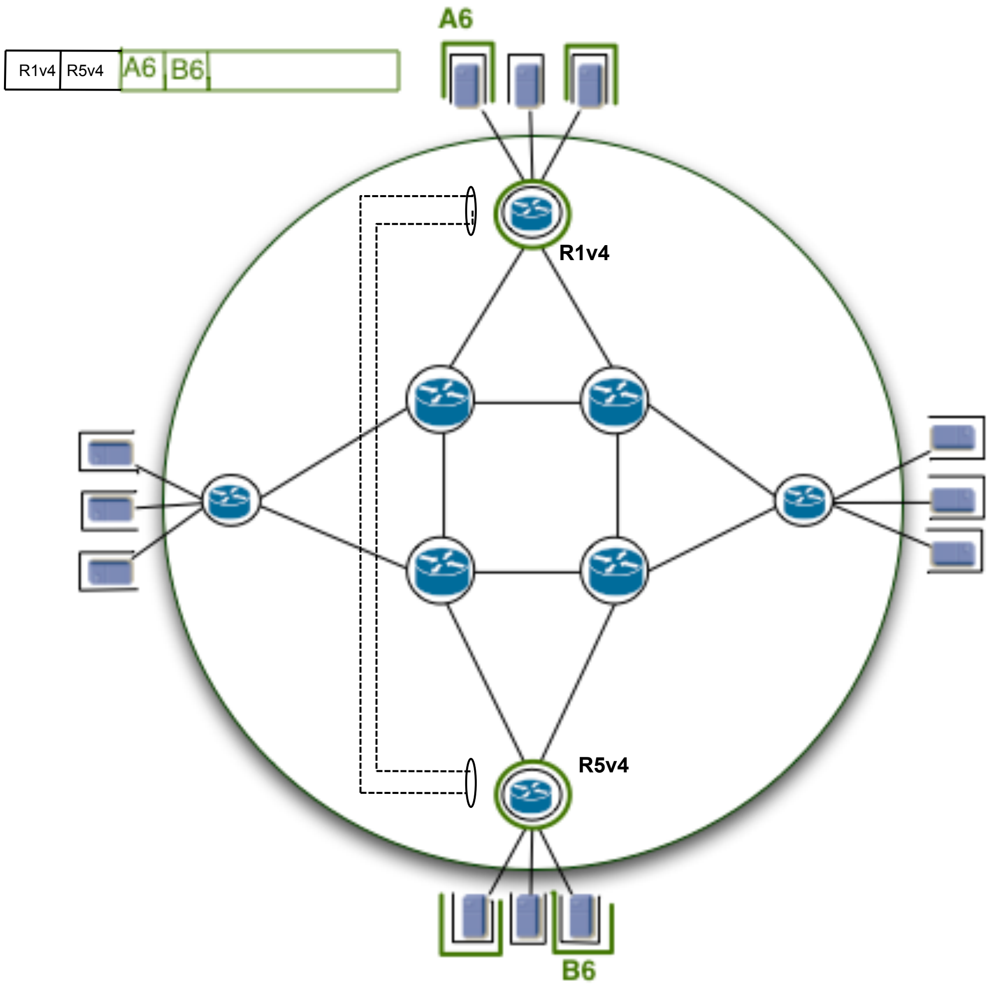
Transition Concept



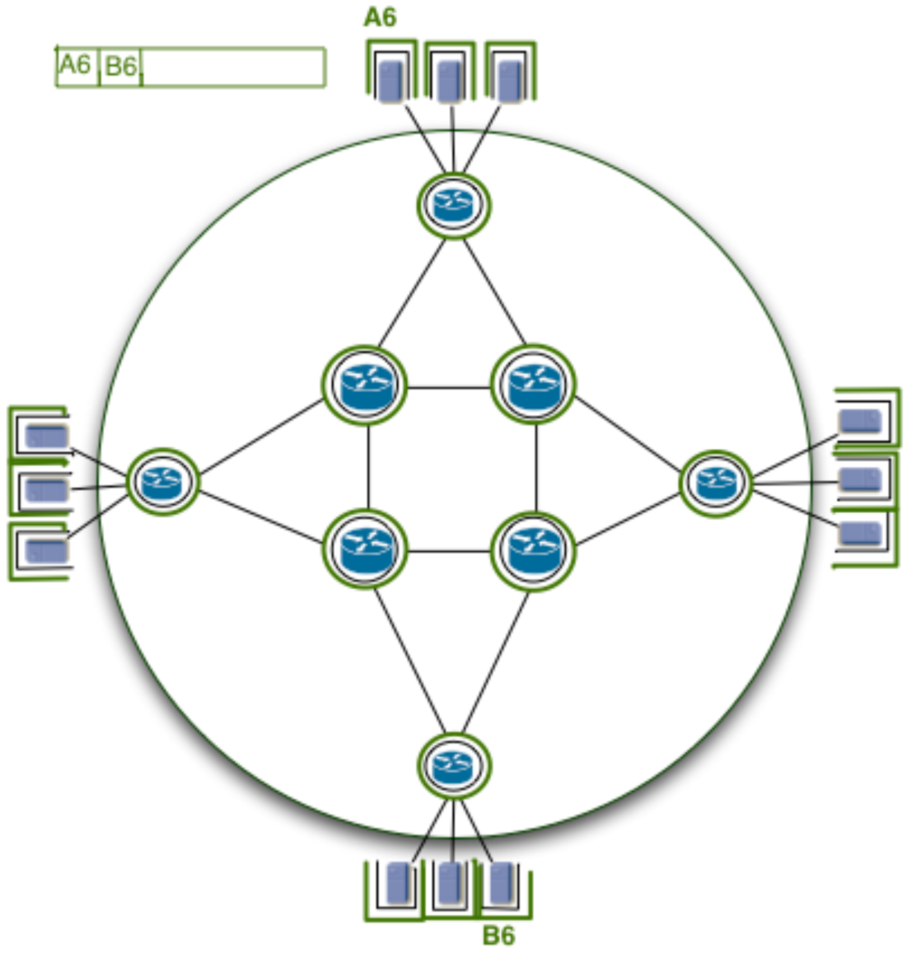
Transition Concept



Transition Concept



Transition Concept



IPv4 to IPv6 Transition

- Implementation rather than transition
  - No fixed day to convert
- The key to successful IPv6 transition
  - Maintaining compatibility with IPv4 hosts and routers while deploying IPv6
    - Millions of IPv4 nodes already exist
    - Upgrading every IPv4 nodes to IPv6 is not feasible
    - No need to convert all at once
    - Transition process will be gradual

Strategies available for Service Providers

- Do nothing
  - Wait and see what competitors do
  - Business not growing, so don’t care what happens
- Extend life of IPv4
  - Force customers to NAT
  - Buy IPv4 address space on the marketplace
- Deploy IPv6
  - Dual-stack infrastructure
  - IPv6 and NATed IPv4 for customers
  - 6rd (Rapid Deploy) with native or NATed IPv4 for customers
  - Or various other combinations of IPv6, IPv4 and NAT

Dual-Stack Networks

- Both IPv4 and IPv6 have been fully deployed across all the infrastructure
  - Routing protocols handle IPv4 and IPv6
  - Content, application, and services available on IPv4 and IPv6
- End-users use dual-stack network transparently:
  - If DNS returns IPv6 address for domain name query, IPv6 transport is used
  - If no IPv6 address returned, DNS is queried for IPv4 address, and IPv4 transport is used instead
- It is envisaged that the Internet will operate dual-stack for many years to come

IP in IP Tunnels

- A mechanism whereby an IP packet from one address family is encapsulated in an IP packet from another address family
  – Enables the original packet to be transported over network of another address family
- Allows ISP to provide dual-stack service prior to completing infrastructure deployment
- Tunnelling techniques include:
  – IPinIP, GRE, 6to4, Teredo, ISATAP, 6rd, MPLS

Address Family Translation (AFT)

- Refers to translation of an IP address from one address family into another address family
  - e.g. IPv6 to IPv4 translation (sometimes called NAT64)
  - Or IPv4 to IPv6 translation (sometimes called NAT46)

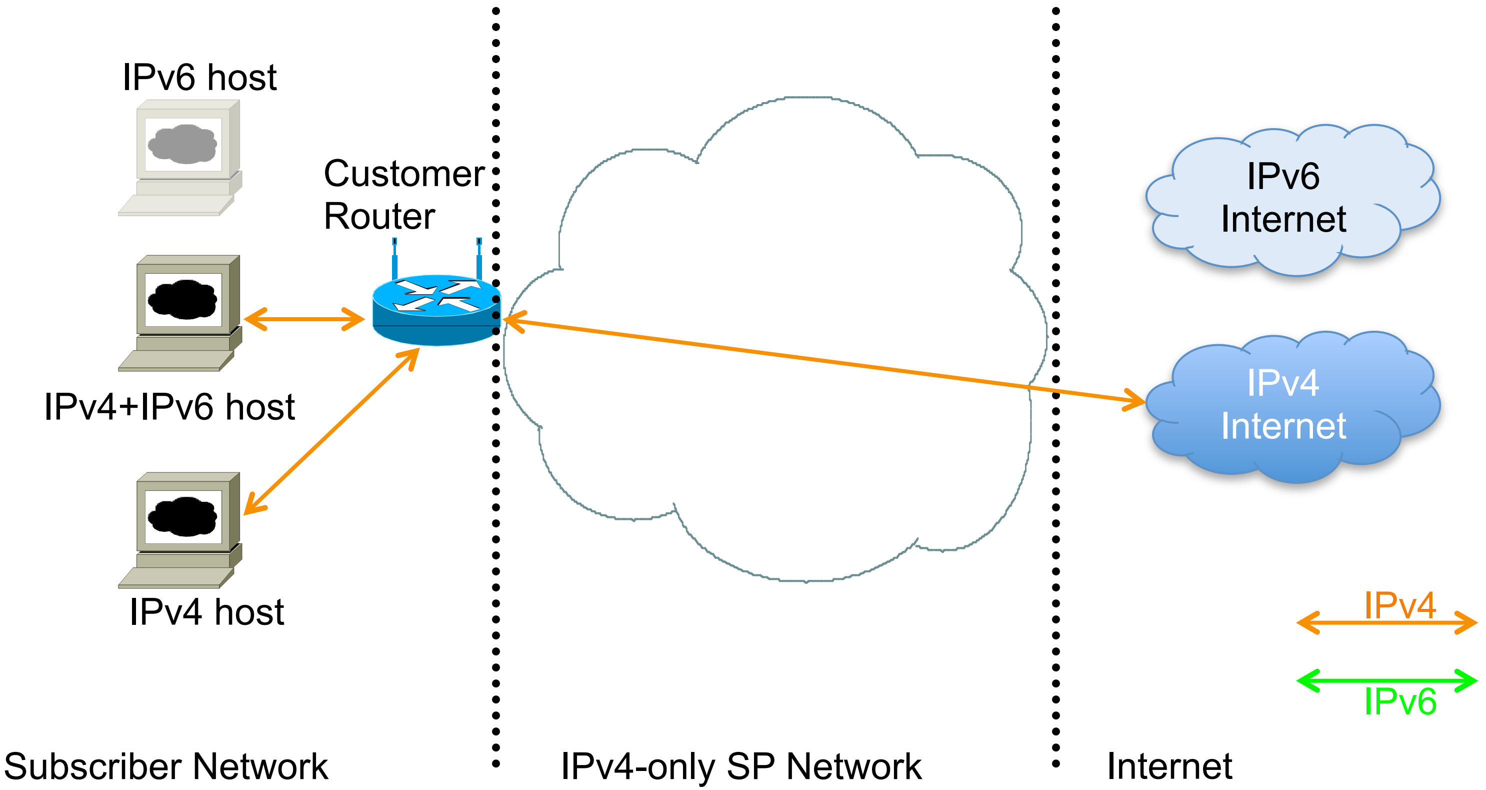
Network Address Translation (NAT)

- NAT is translation of one IP address into another IP address
- NAPT (Network Address & Port Translation) translates multiple IP addresses into one other IP address
  – TCP/UDP port distinguishes different packet flows
- NAT-PT (NAT – Protocol Translation) is a particular technology which does protocol translation in addition to address translation
  – NAT-PT is has now been made obsolete by the IETF
  – http://tools.ietf.org/html/rfc4966

Carrier Grade NAT (CGN)

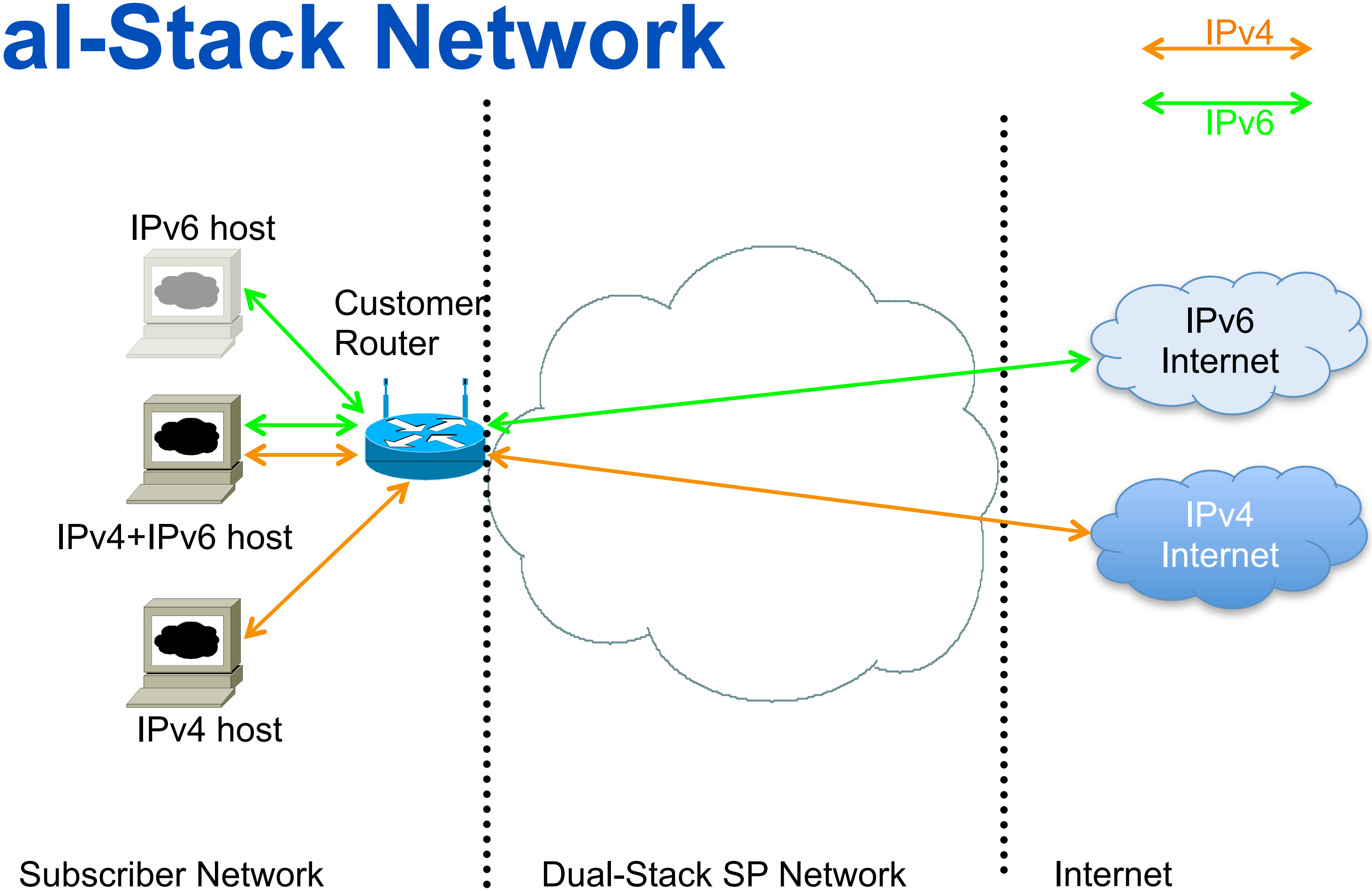
- ISP version of subscriber NAT
  - Subscriber NAT can handle only hundreds of translations
  - ISP NAT can handle millions of translations
- Not limited to just translation within one address family, but does address family translation as well
- Often referred to as Large Scale NAT (LSN)

IPv4 only Network



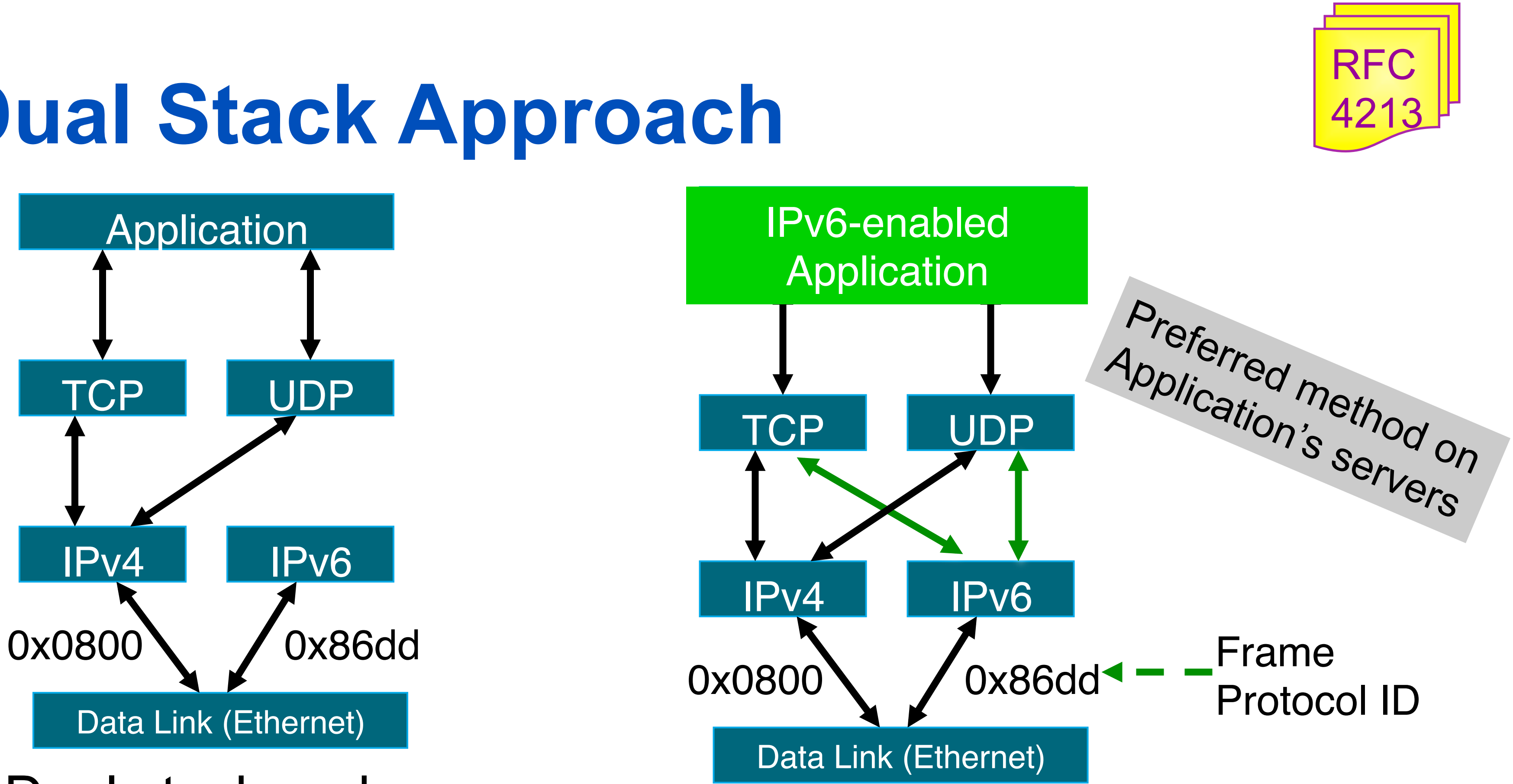
- The situation for many SPs today:
  - No IPv6 for consumer
  - IPv4 scaling lasts as long as IPv4 addresses are available

Dual-Stack Network



- The original transition scenario, but dependent on:
  - IPv6 being available all the way to the consumer
  - Sufficient IPv4 address space for the consumer and SP core

Dual Stack Approach

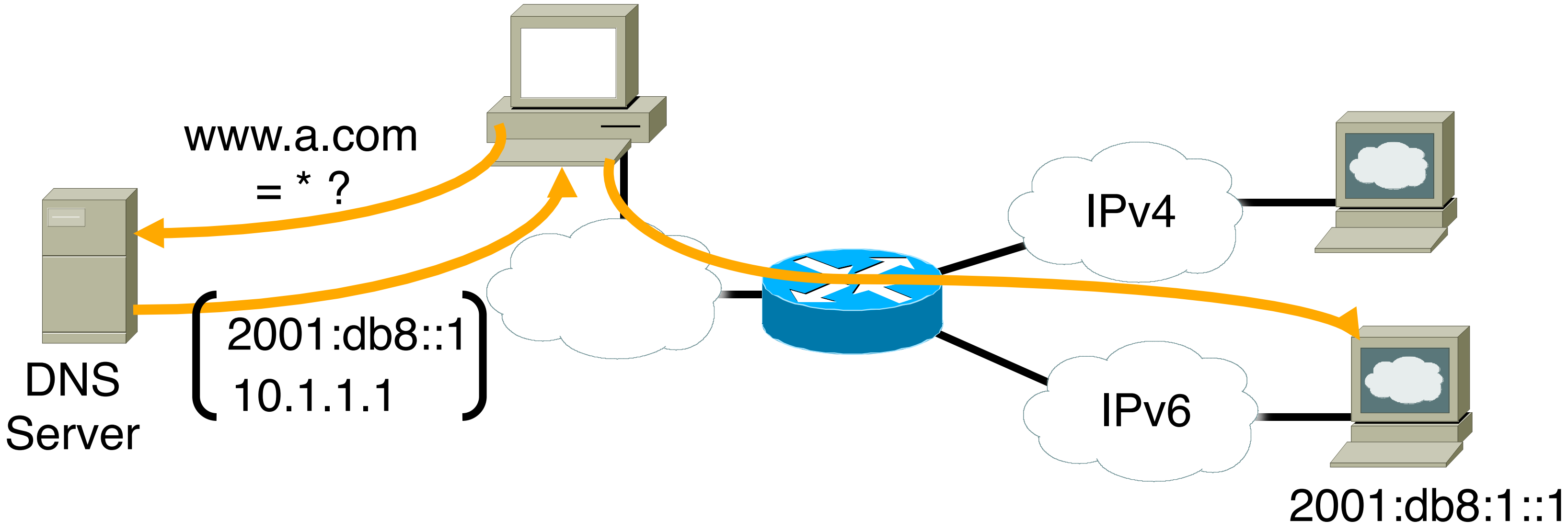


- Dual stack node means:
  - Both IPv4 and IPv6 stacks enabled
  - Applications can talk to both
  - Choice of the IP version is based on name lookup and application preference

Dual Stack Challenges

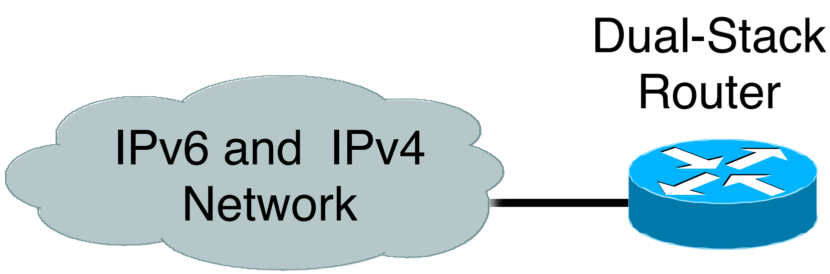
- Compatible software
  - Eg. If you use OSPFv2 for your IPv4 network you need to run OSPFv3 in addition to OPSFv2
- Transparent availability of services
- Deployment of servers and services
- Content provision
- Business processes
- Traffic monitoring
- End user deployment

Dual Stack Approach & DNS



- In a dual stack case, an application that:
  - Is IPv4 and IPv6-enabled
  - Asks the DNS for all types of addresses
  - Chooses one address and, for example, connects to the IPv6 address

A Dual Stack Configuration



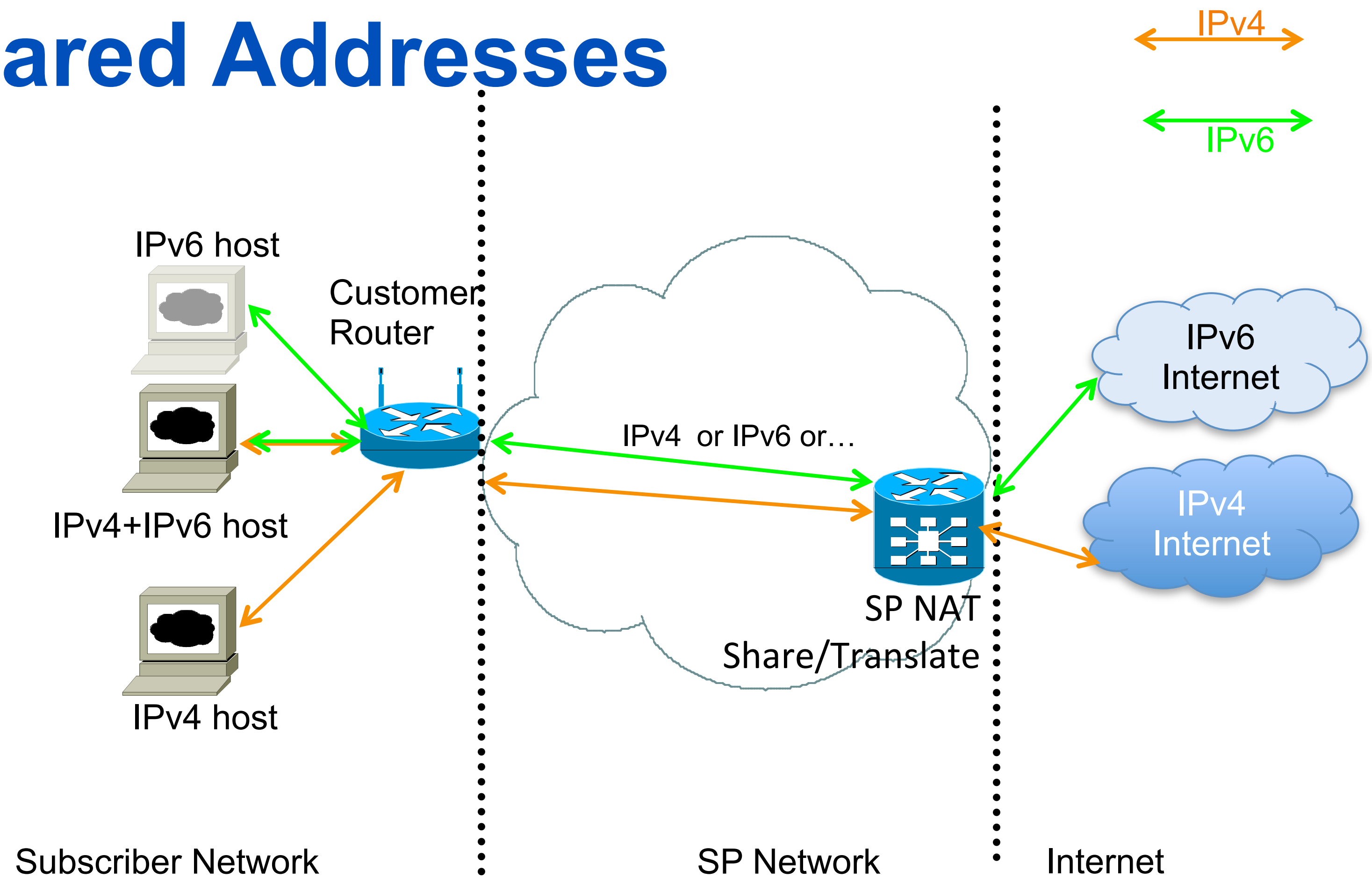

```
router#
ipv6 unicast-routing

interface Ethernet0
 ip address 192.168.99.1 255.255.255.0
 ipv6 address 2001:db8:213:1::1/64
```

IPv4: 192.168.99.1

IPv6: 2001:db8:213:1::1/64

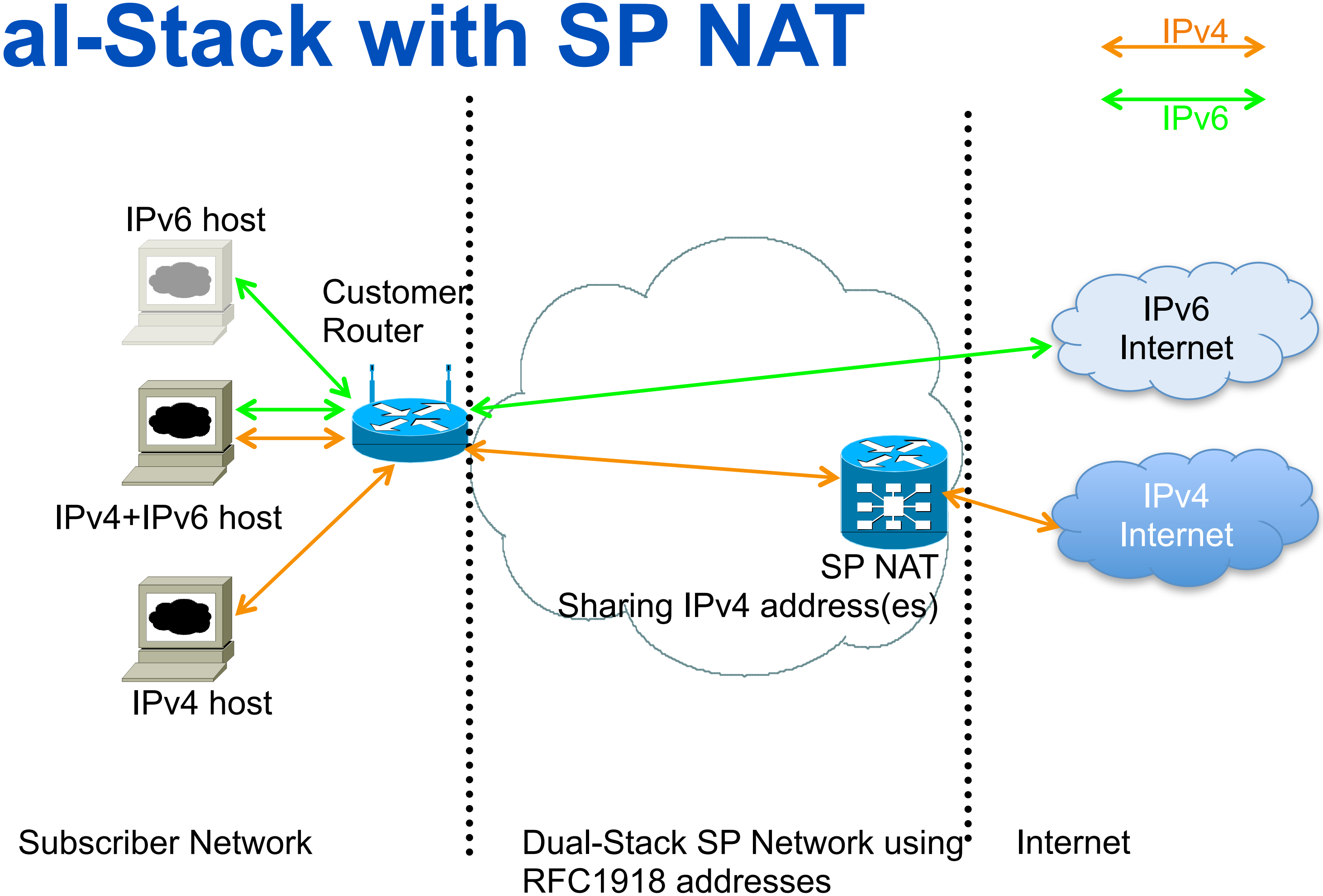
- IPv6-enabled router
  - If IPv4 and IPv6 are configured on one interface, the router is dual-stacked
  - Telnet, Ping, Traceroute, SSH, DNS client, TFTP,…

Shared Addresses



- SP shares globally routable IPv4 addresses amongst customers:
  - Customer could have IPv6, or IPv4, or a mixture
  - SP NAT device does necessary sharing and translation to access IPv4 and IPv6 Internets

Dual-Stack with SP NAT



- More likely scenario:
  - IPv6 being available all the way to the consumer
  - SP core and customer has to use IPv4 NAT due to v4 depletion

Using Tunnels for IPv6 Deployment

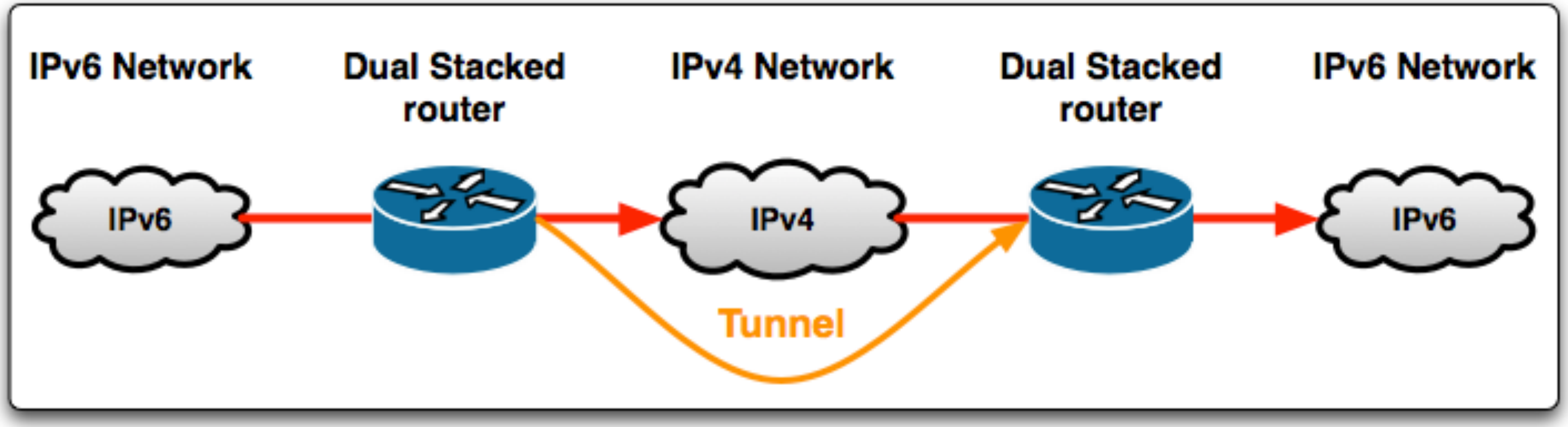
- Many techniques are available to establish a tunnel:
  - Manually configured
    - Manual Tunnel (RFC 2893)
    - GRE (RFC 2473)
  - Semi-automated
    - Tunnel broker
  - Automatic
    - 6to4 (RFC 3056)
    - 6rd

Tunnels

- Part of a network is IPv6 enabled
  - Tunnelling techniques are used on top of an existing IPv4 infrastructure and uses IPv4 to route the IPv6 packets between IPv6 networks by transporting these encapsulated in IPv4
  - Tunnelling is used by networks not yet capable of offering native IPv6 functionality
  - It is the main mechanism currently being deployed to create global IPv6 connectivity
- Manual, automatic, semi-automatic configured tunnels are available

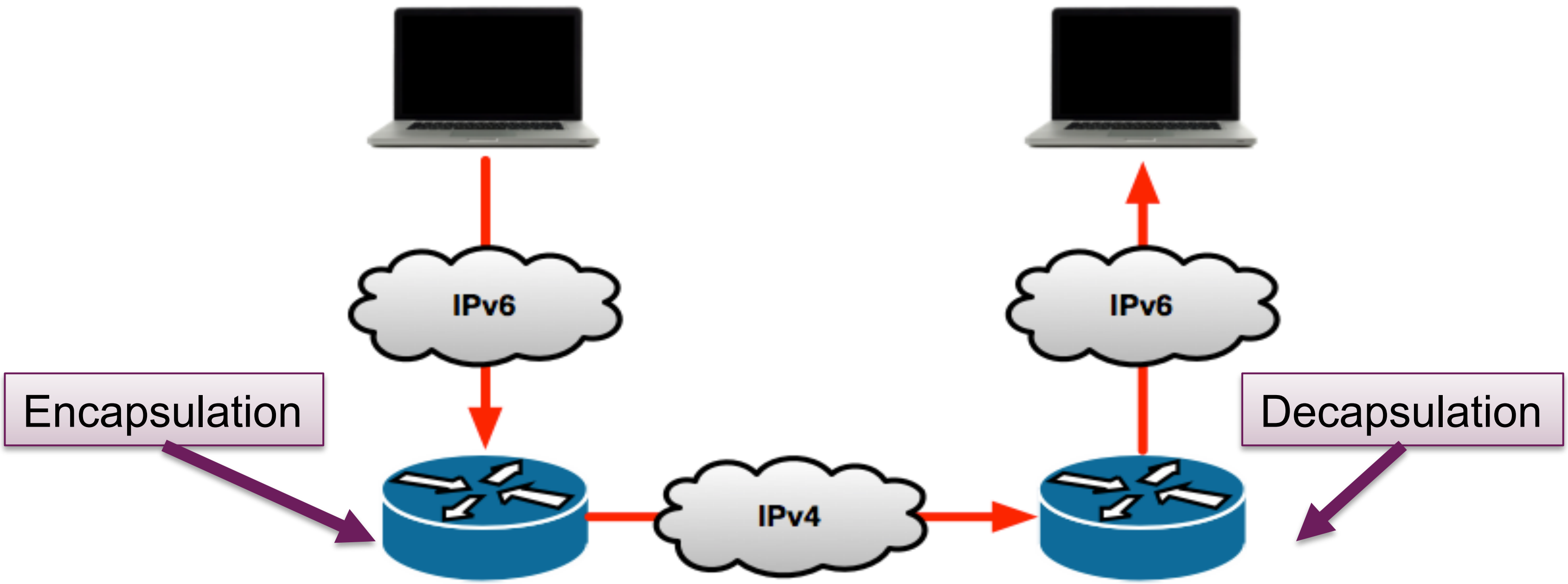
Tunneling – General Concept

- Tunneling can be used by routers and hosts
  - Tunneling is a technique by which one transport protocol is encapsulated as the payload of another.

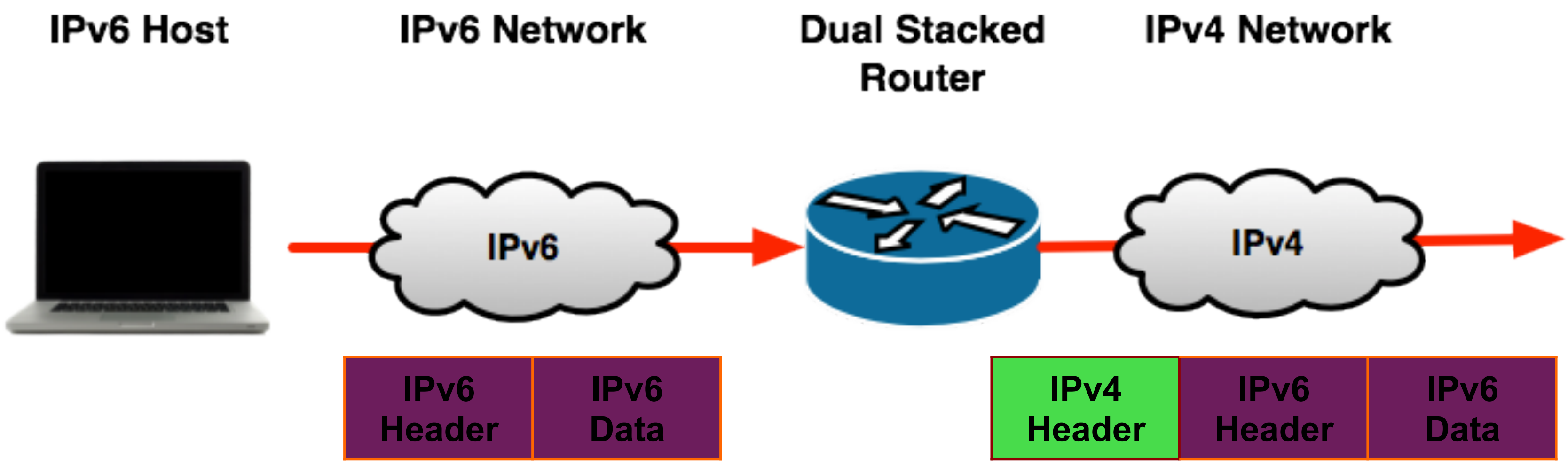


Tunneling – General Concept

- Two stepped process
  - Encapsulation of IPv6 packets to IPv4 packets
  - Decapsulation of IPv4 packets to IPv6 packets

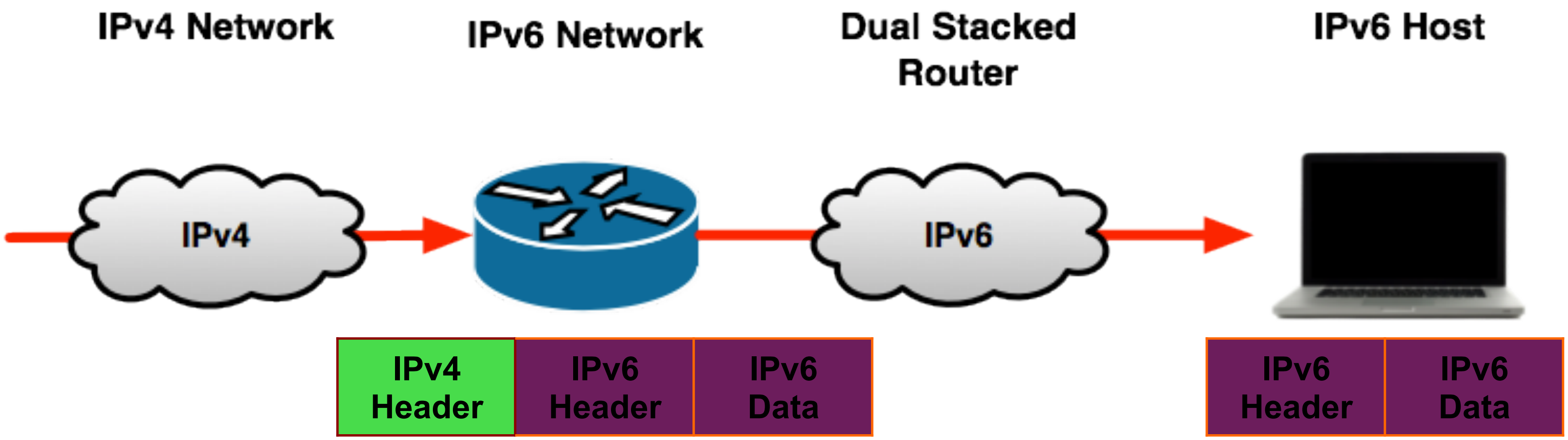


Tunnel Encapsulation



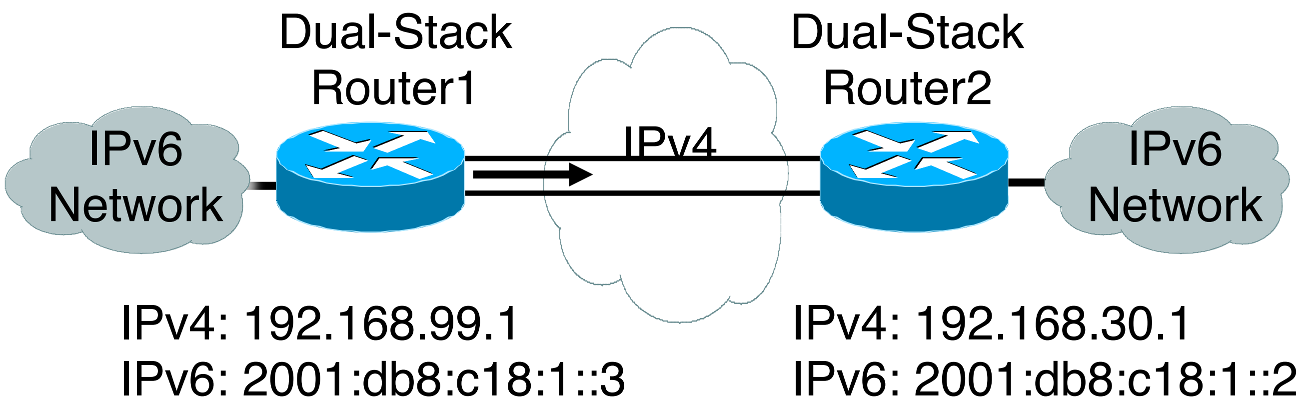
IPv6 essentials by Silvia Hagen, p258

Tunnel Decapsulation



IPv6 essentials by Silvia Hagen, p258

Manually Configured Tunnel (RFC4213)

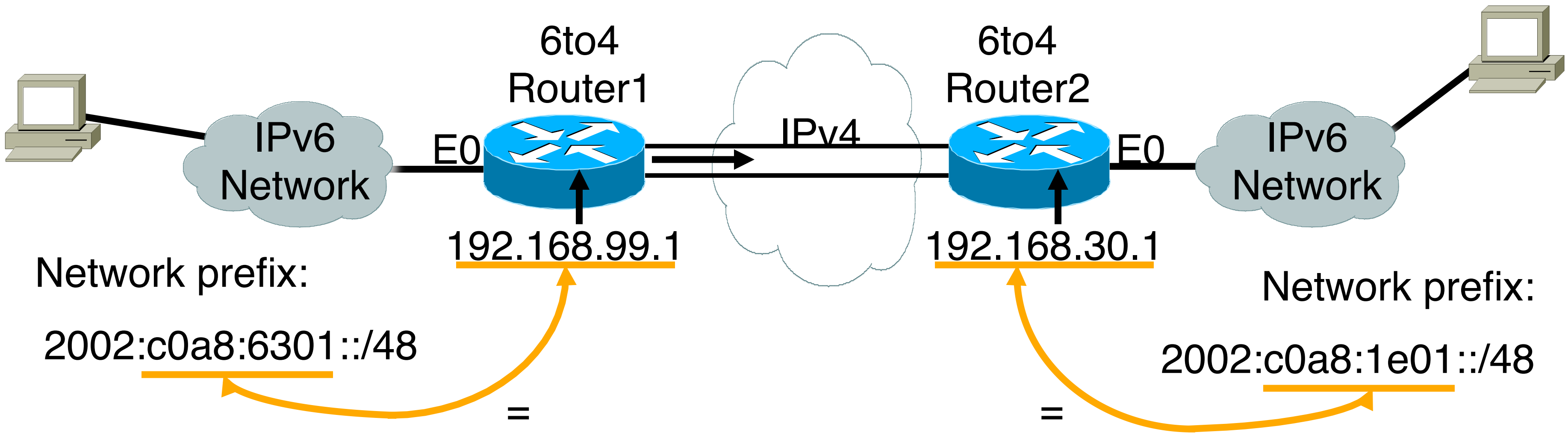



```
router1#

interface Tunnel0
 ipv6 address 2001:db8:c18:1::3/64
 tunnel source 192.168.99.1
 tunnel destination 192.168.30.1
 tunnel mode ipv6ip
```

```
router2#

interface Tunnel0
 ipv6 address 2001:db8:c18:1::2/64
 tunnel source 192.168.30.1
 tunnel destination 192.168.99.1
 tunnel mode ipv6ip
```

- Manually Configured tunnels require:
  - Dual stack end points
  - Both IPv4 and IPv6 addresses configured at each end

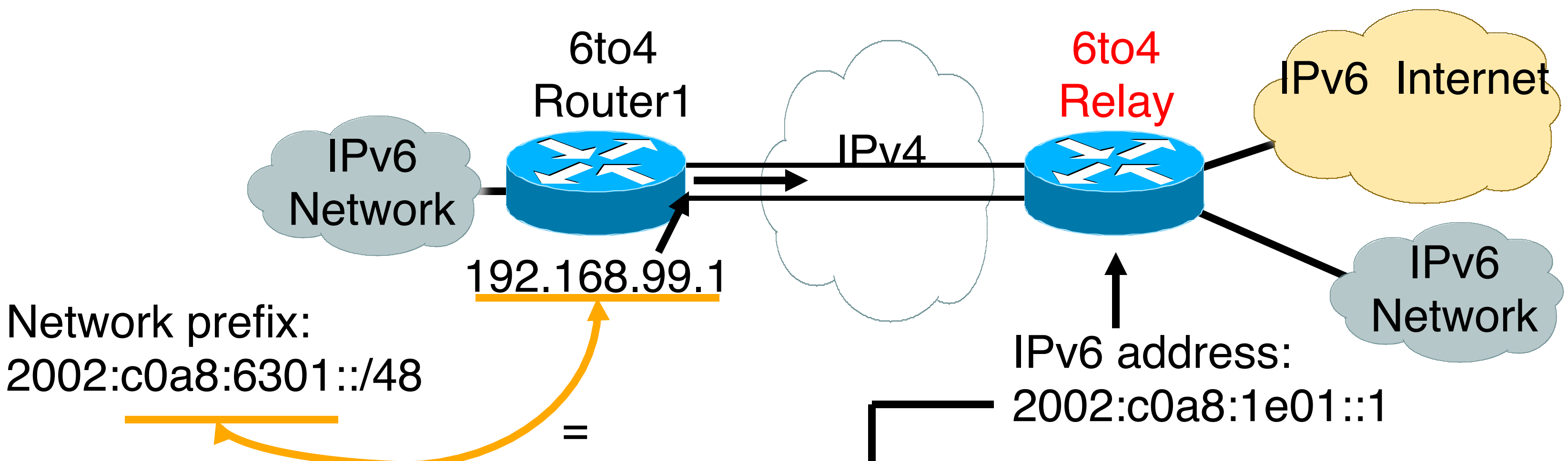
6to4 Tunnel (RFC 3056)



- 6to4 Tunnel:
  - Is an automatic tunnel method
  - Gives a prefix to the attached IPv6 network
  - 2002::/16 assigned to 6to4
  - Requires one global IPv4 address on each Ingress/Egress site

```
router2#
interface Loopback0
 ip address 192.168.30.1 255.255.255.0
 ipv6 address 2002:c0a8:1e01:1::/64 eui-64
interface Tunnel0
 no ip address
 ipv6 unnumbered Ethernet0
 tunnel source Loopback0
 tunnel mode ipv6ip 6to4

ipv6 route 2002::/16 Tunnel0
```

6to4 Relay




```
router1#
interface Loopback0
 ip address 192.168.99.1 255.255.255.0
 ipv6 address 2002:c0a8:6301:1::/64 eui-64
interface Tunnel0
 no ip address
 ipv6 unnumbered Ethernet0
 tunnel source Loopback0
 tunnel mode ipv6ip 6to4

ipv6 route 2002::/16 Tunnel0
ipv6 route ::/0 2002:c0a8:1e01::1
```

- 6to4 relay:
  - Is a gateway to the rest of the IPv6 Internet
  - Default router
  - Anycast address (RFC 3068) for multiple 6to4 Relay

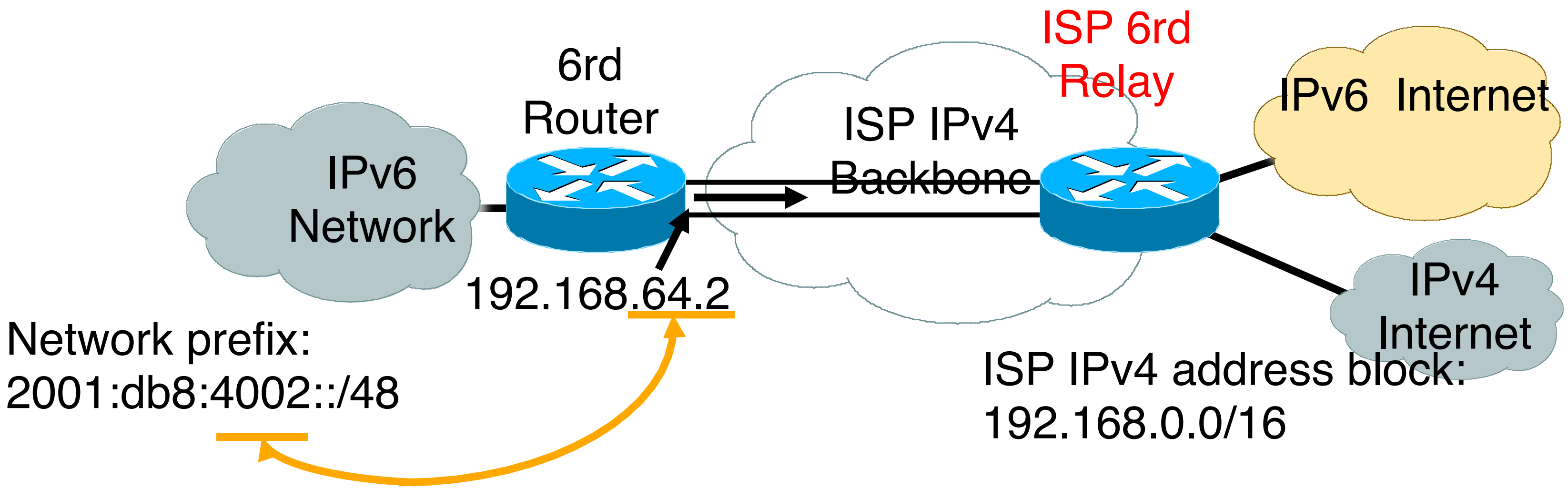
6to4 in the Internet

- 6to4 prefix is 2002::/16
- 192.88.99.0/24 is the IPv4 anycast network for 6to4 routers
- 6to4 relay service
  - An ISP who provides a facility to provide connectivity over the IPv4 Internet between IPv6 islands
    - Is connected to the IPv6 Internet and announces 2002::/16 by BGP to the IPv6 Internet
    - Is connected to the IPv4 Internet and announces 192.88.99.0/24 by BGP to the IPv4 Internet
  - Their router is configured with local IPv4 address of 192.88.99.1 and local IPv6 address of 2002:c058:6301::1

6to4 in the Internet
Relay Router Configuration

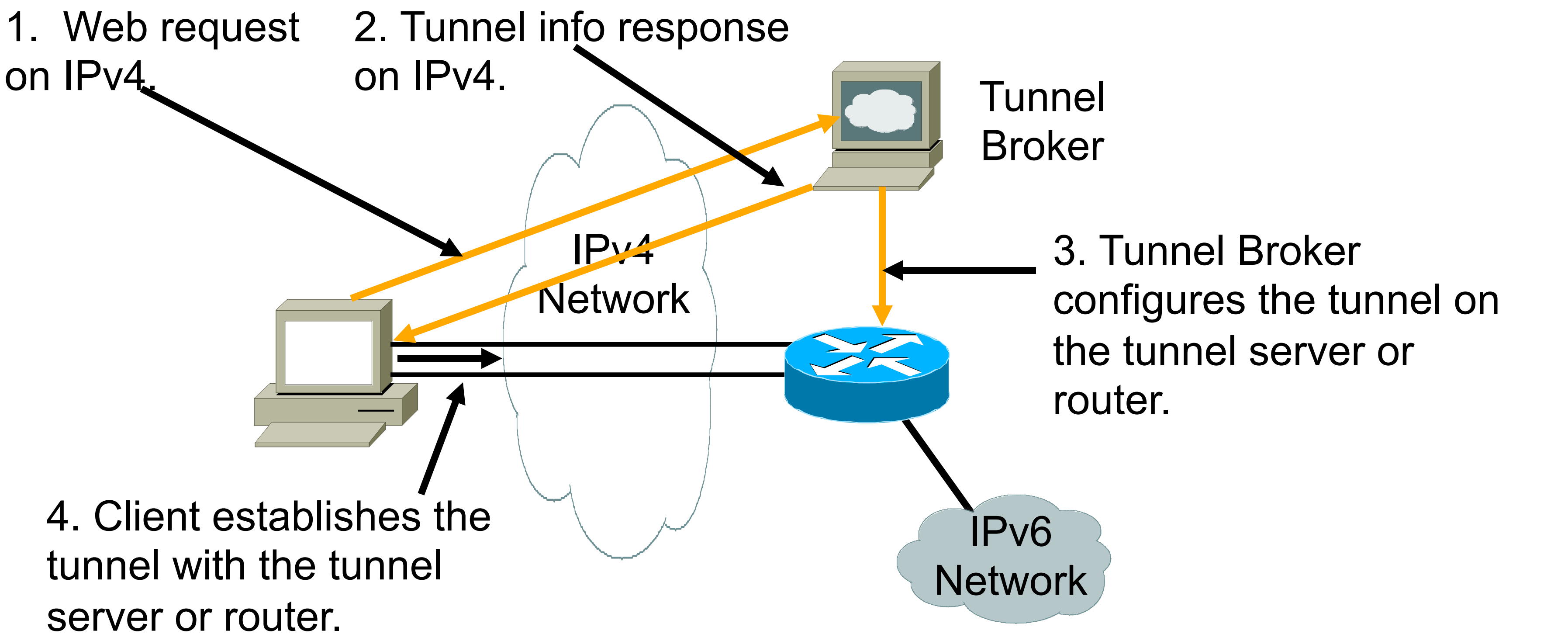
```
interface loopback0
 ip address 192.88.99.1
   255.255.255.255
 ipv6 address 2002:c058:6301::1/128
!
interface tunnel 2002
 no ip address
 ipv6 unnumbered Loopback0
 tunnel source Loopback0
 tunnel mode ipv6ip 6to4
 tunnel path-mtu-discovery
!
interface FastEthernet0/0
 ip address 105.3.37.1 255.255.255.0
 ipv6 address 2001:db8::1/64
!
router bgp 100
 address-family ipv4
  neighbor <v4-transit> remote-as 101
  network 192.88.99.0 mask
   255.255.255.0.
 address-family ipv6
  neighbor <v6-transit> remote-as 102
  network 2002::/16
!
ip route 192.88.99.0 255.255.255.0
   null0 254
ipv6 route 2002::/16 tunnel2002
```

6rd Tunnel



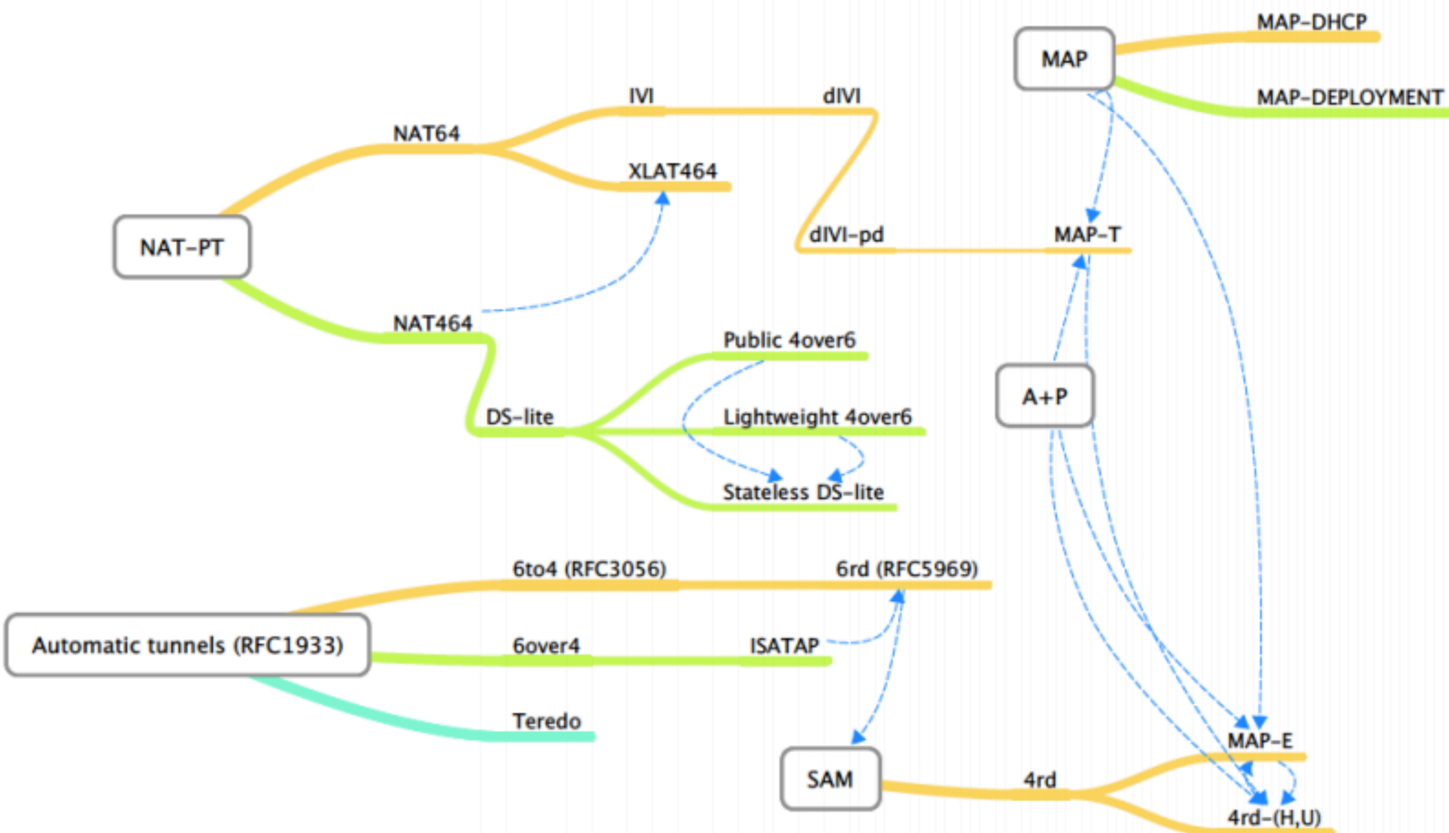
- 6rd (example):
  - ISP has 192.168.0.0/16 IPv4 address block
  - ISP has 2001:db8::/32 IPv6 address block
  - Final 16 bits of IPv4 address used on customer point-to-point link to create customer /48 → customer uses 2001:db8:4002::/48 address space
  - IPv6 tunnel to ISP 6rd relay bypasses infrastructure which cannot handle IPv6

Tunnel Broker



- Tunnel broker:
  – Tunnel information is sent via http-ipv4

Evolution of IPv6 Transition Technologies as of March 2013 (IETF83)



Source: IETF 83 Softwires

Conclusions
Potential Techniques

Scenario	Potential Techniques
Content and Applications move to IPv6	IPv6 only network; Dual-Stack, 6rd and DS-lite as migration techniques
Content and Applications on IPv4 and IPv6	Dual-Stack (if enough IPv4) or 6rd; SP IPv4-NAT; DS-lite (for greenfield) *
Users are IPv6 only	IPv6 only network; Dual-Stack, 6rd and DS-lite as migration techniques
No change (double NAT)	SP IPv4-NAT *
No change (no double NAT)	Do nothing *

* Transfer Market applicable

Recommendations

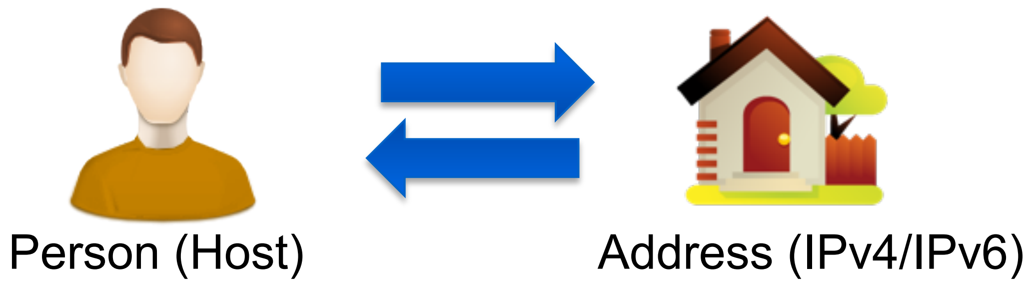
- Start deploying IPv6 as long term strategy
- Evaluate current addressing usage to understand if IPv4 to IPv4 NAT is sufficient for transition period
- Prepare a translation mechanism from the IPv4 Internet to the IPv6 Internet
- Educate your user base on IPv6 introduction, the use cases and troubleshooting

Overview

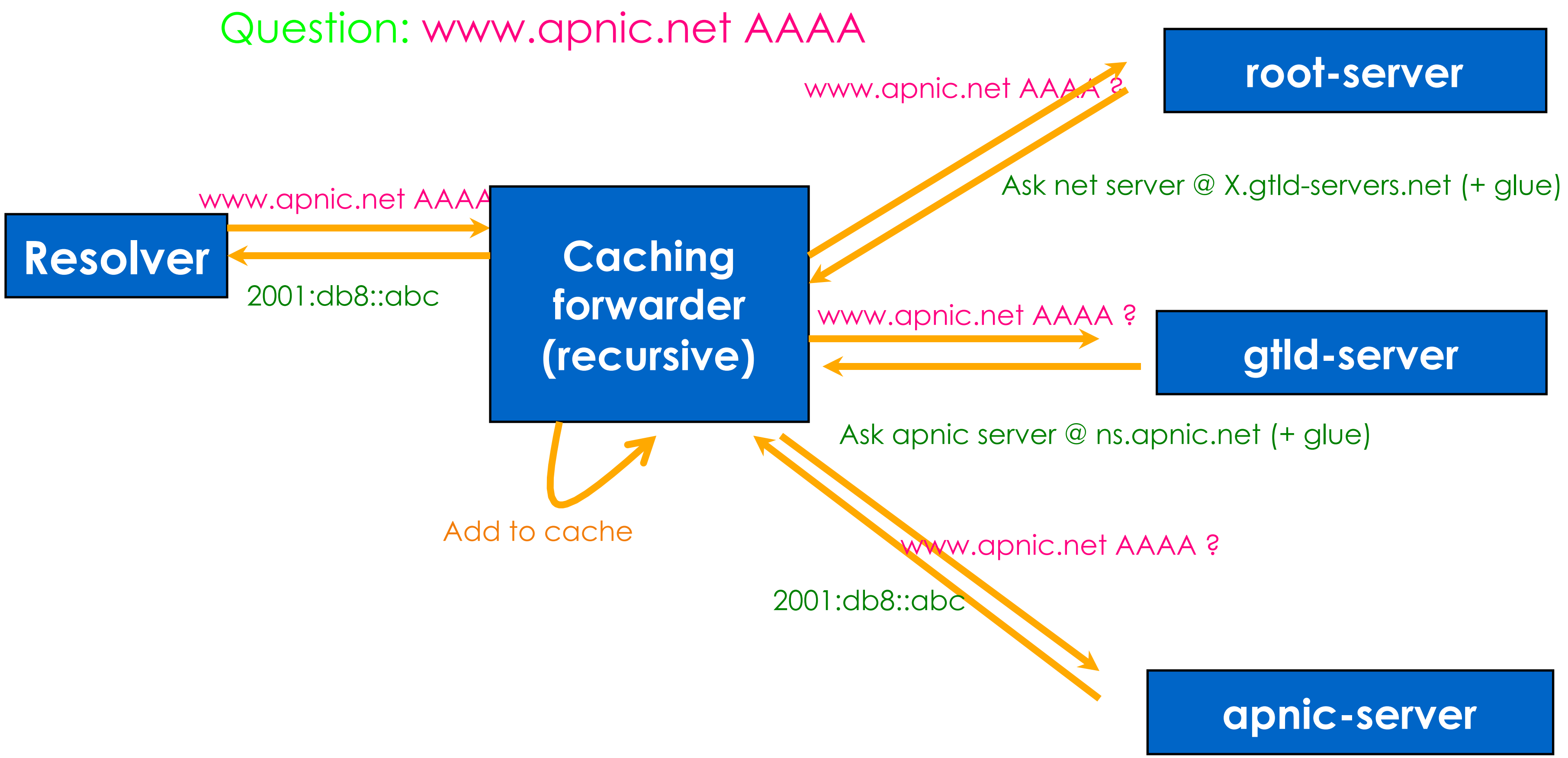
- Introduction to IPv6
- IPv6 Protocol Architecture
- Mobile IPv6 Operation
- IPv6 Security Features
- IPv6 Addressing and Subnetting
- IPv4 to IPv6 Transition Technologies
- **IPv6 Services**

DNS Basics

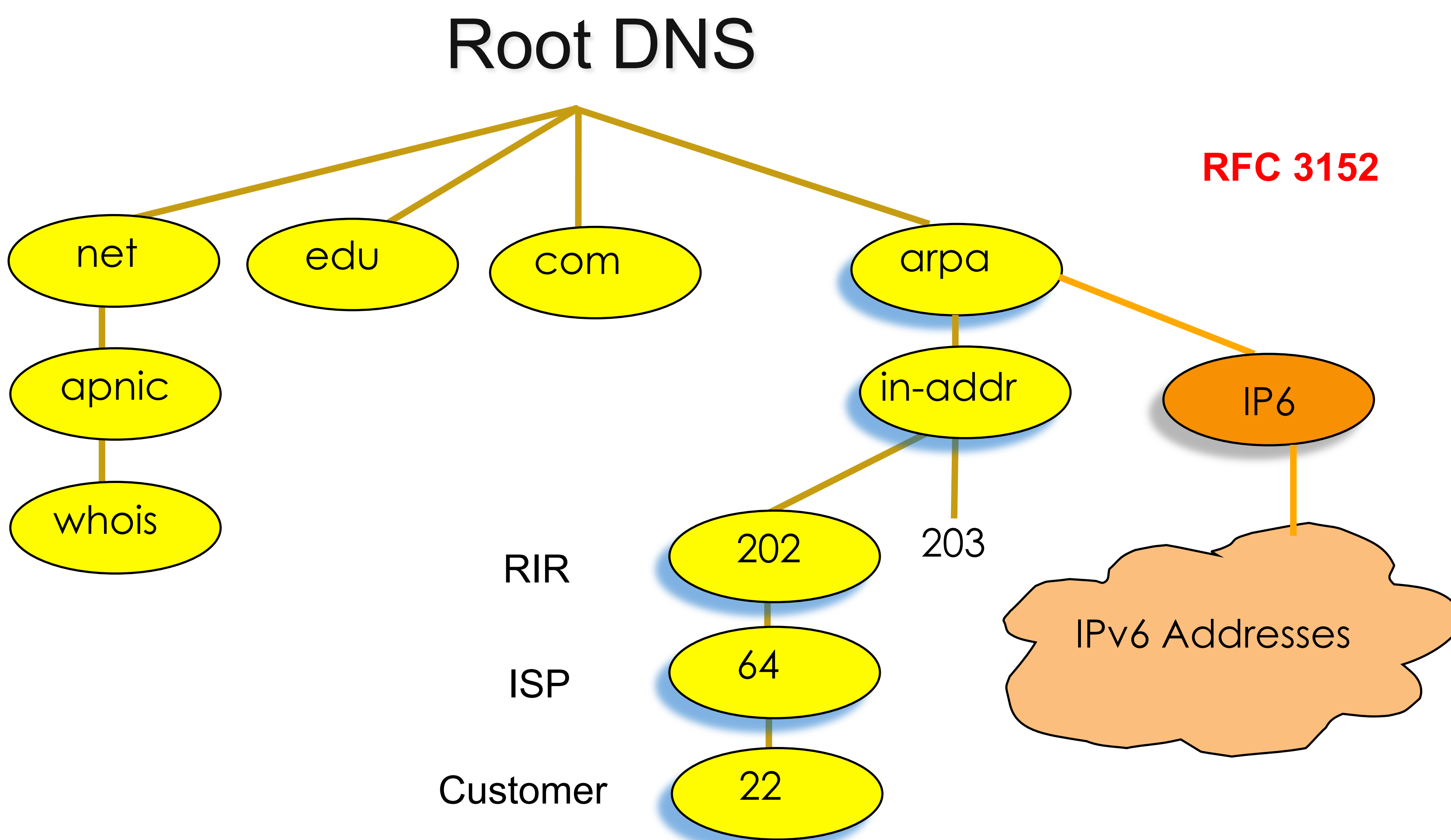
- DNS maps one resource to another resource
  - IP address to hostname (and vice versa)
  - Useful for long addresses (such as IPv6)
- Globally distributed, hierarchical tree structure
- Three components: namespace, resolvers, servers
- Resource records are the actual mappings
  - RR Types: A, AAAA, PTR, CNAME, etc



DNS Overview (Lookup)



IPv6 Reverse DNS tree



RFCs

- RFC 3596 – DNS Extensions to Support IPv6
  - Introduced AAAA record
  - IP6.ARPA domain
  - Updates RFC1886 (uses IP6.INT domain)
- RFC 3152 – Delegation of IP6.ARPA
  - Used for reverse mapping
  - IP6.ARPA is analogous to IN-ADDR.ARPA zone for IPv4
- RFC 3901 – DNS IPv6 Transport Operational Guidelines
  - As a Best Common Practice

IPv6 Representation in the DNS

- Forward lookup support: Multiple RR records for name to number
  - AAAA (Similar to A RR for IPv4)
- Reverse lookup support:
  - Reverse nibble format for zone ip6.arpa
- Multiple addresses are possible for any given name
  - Ex: in a multi-homed situation
- Can assign A records and AAAA records to a given name/ domain
- Can also assign separate domains for IPv6 and IPv4

Sample Forward Lookup File

```
apnic.net. 7200 IN        SOA      ns.apnic.net. admin.apnic.net.
                        (
                          2010020901      ; Serial
                          12h   ; Refresh 12 hours
                          4h   ; Retry 4 hours
                          4d ; Expire 4 days
                          2h ; Negative cache 2 hours )

apnic.net.              7200   IN     NS       ns.apnic.net.
server1.apnic.net.      3600   IN     A        193.0.1.162
                        3600   IN     AAAA     2001:0db8:1230::ABC:1
```

IPv6 Reverse Lookups – PTR records

- Similar to the IPv4 reverse record

```
b.a.9.8.7.6.5.0.4.0.0.0.3.0.0.0.2.0.0.0.1.0.0.0.0.0.0.0.1.2.3.4.ip6.arpa.
          IN      PTR  test.ip6.example.com.
```

- Example: reverse name lookup for a host with address

  3ffe:8050:201:1860:42::1

```
$ORIGIN 0.6.8.1.1.0.2.0.0.5.0.8.e.f.f.3.ip6.arpa.
1.0.0.0.0.0.0.0.0.0.0.0.2.4.0.0  14400  IN PTR host.example.com.
```

Sample Reverse Lookup File

```
$ORIGIN 0.0.0.0.4.3.2.1.8.B.D.0.1.0.0.2

apnic.net. 7200 IN      SOA     ns.apnic.net. admin.apnic.net.
                      (
                        2010020901      ; Serial
                        12h   ; Refresh 12 hours
                        4h   ; Retry 4 hours
                        4d ; Expire 4 days
                        2h ; Negative cache 2 hours )

apnic.net.              7200   IN    NS      ns.apnic.net.
1.C.B.A.0.0.0.0.0.0.0.0.0.0.0.0 3600   IN    PTR server1.apnic.net.
```

IPv6 in the Root Servers

- http://www.internic.net/zones/named.root
- 9 of 13 root servers have IPv6 AAAA records
  - C, E, G root servers don’t have IPv6 capability yet
  - root.hints file contains the IP address of the root servers

IPv6 in TLDs

- (as of 28 March 2013)
- Total number of TLDs: 317
- TLDs with IPv6: 276 (87%)
- Registered domains with AAAA records: 4,809,569
  - COM: 1,686,543 of 108,698,290 domains
  - NET: 324,983 of 15,040,549 domains

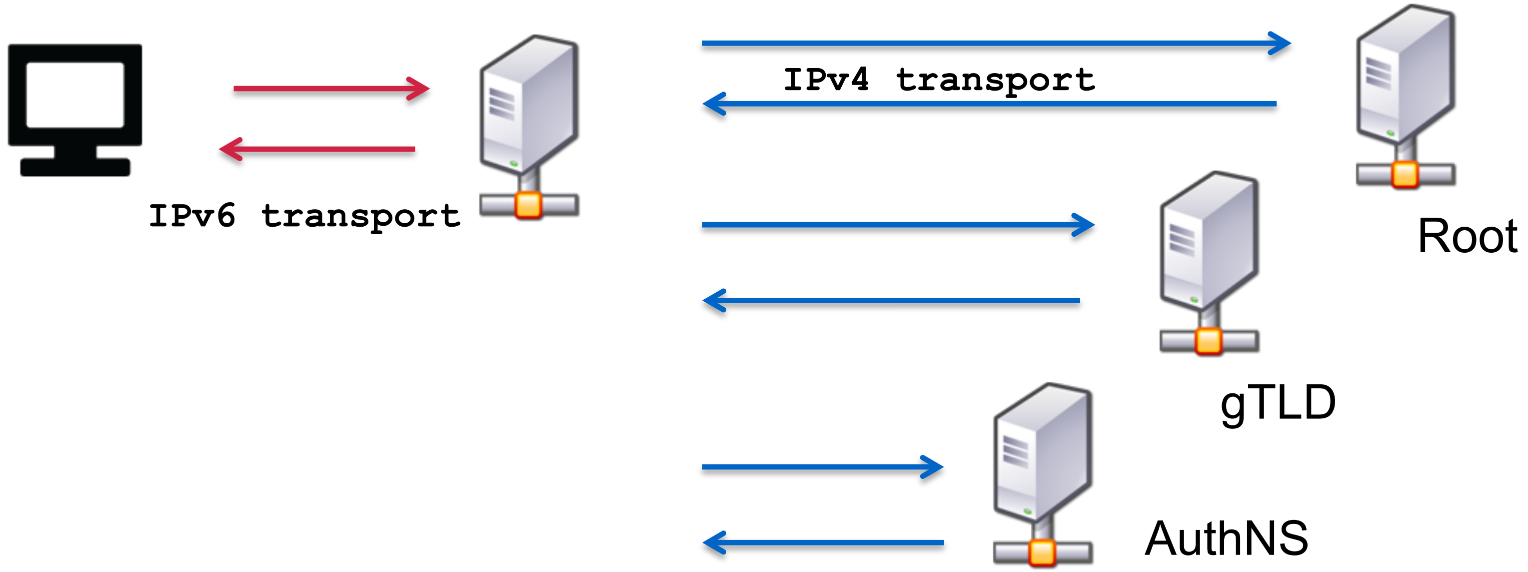
Source: Global IPv6 Deployment Progress Report
http://bgp.he.net/ipv6-progress-report.cgi

Anycast DNS Servers

- Benefits of Anycast DNS
  - Increased reliability
  - Load balancing
  - Improved performance
  - Enhanced security
  - Localized impact of DoS attacks
  - Simplified client configuration
  - Increased availability

DNS Infrastructure

- All DNS servers must be dual-stack because not all DNS Infrastructure supports IPv6 yet



Using BIND with IPv6

- BIND options for IPv6
  - `Listen-on-v6 { };`
  - `Query-source-v6 { };`
  - `Use-v6-udp-ports or avoid-v6-udp-ports`
  - `Transfer-source-v6`
- AAAA records
- PTR records
  - In named.conf

```
Zone "1.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa" {
        Type master;
        File "ipv6ptr.zone";
};
```

  - In zone file

```
        4.3.2.1.0.0.0.1.0.0.0.0. IN PTR www.example.com
```

Forward and Reverse DNS

- Populating the DNS is an often omitted piece of an ISP operation
  – Unfortunately it is extremely vital, both for connectivity and for troubleshooting purposes
- Forward DNS for IPv6
  – Simply a case of including suitable AAAA records alongside the corresponding A records of a host
- Reverse DNS for IPv6
  – Requires getting the /32 address block delegated from the RIR, and then populating the ip6.arpa fields

Forward DNS

- Operators typically access the router by connecting to loopback interface address
- Setting up the IPv6 entries means adding a quad-A record beside each A record:

```
r1.pop1    A      192.168.1.1
           AAAA   2001:db8::1:1
r2.pop1    A      192.168.1.2
           AAAA   2001:db8::1:2
gw1.pop1   A      192.168.1.3
           AAAA   2001:db8::1:10
```

Forward DNS

- Completing the infrastructure zone file as per the example is sufficient
  - Update the SOA record
  - Reload the nameserver software
  - All set
- If connecting from an IPv6 enabled client
  - IPv6 transport will be chosen before the IPv4 transport
  - For all connections to IPv6 enabled devices which have entries in the forward DNS zones

Reverse DNS

- First step is to have the /32 address block delegated by the RIR
- Prepare the local nameservers to handle the reverse zone, for example in BIND:

```
zone ″8.b.d.0.1.0.0.2.ip6.arpa" in {
        type master;
        file "ip6.arpa-zones/db.2001.0db8;
        allow-transfer {"External"; "NOC-NET";};
};
```

- And then “create and populate the zone file”

Reverse DNS

- The db.2001.0db8 zone file heading:

```
$TTL 86400
@     IN    SOA    ns1.isp.net. hostmaster.isp.net. (
                   2008111000        ;serial
                   43200             ;refresh
                   3600              ;retry
                   608400            ;expire
                   7200)             ;minimum

        NS         ns1.isp.net.
        NS         ns2.isp.net.
;Hosts are list below here
```

Creating the reverse zone file

- IPv6 addresses are 128 bits long
  - Bits are grouped in 4 and represented in by a hexadecimal digit
  - Therefore and IPv6 address has 32 hexadecimal digits in it
  - Each one gets a field in IPv6's reverse DNS
- 2001:db8::1:1 is the loopback address for cr1.pop1
  - We can omit leading zeros and padding zeros are replaced with a set of ::
  - This cannot be done in Reverse DNS ip6.arpa zone files
- Equivalent reverse value would be:
  - 1.0.0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.b.d.
    0.1.0.0.2.ip6.arpa

Creating the reverse zone file

- Major task is filling up the zone file with entries such as
  - 1.0.0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.d.b.0.1.0.0.2.ip6.arpa
- Strategy needed!
  - Otherwise serious errors would result, reverse DNS wouldn't function,
  - Missing out a single "0" will have consequences
- Possible strategies:
  - Delegate infrastructure /48 to a separate zone file
  - Delegate PtP link /48 to a separate zone file
  - Each customer /48 is delegated to a separate zone file
  - Etc…

Creating the reverse zone file

- Reverse zone for the /32 could read like:

```
; header as previously
;
; Infrastructure /48
0.0.0.0   NS      ns1.isp.net.
0.0.0.0   NS      ns2.isp.net.
; Customer PtP link /48
1.0.0.0   NS      ns1.isp.net.
1.0.0.0   NS      ns2.isp.net.
; Customer One /48
2.0.0.0   NS      ns1.isp.net.
2.0.0.0   NS      ns2.isp.net.
; etc - fill in as we grow
f.f.f.f   NS      ns1.isp.net.
f.f.f.f   NS      ns2.isp.net.
```

Infrastructure reverse zone

- And now we have a /48 reverse zone delegated for infrastructure
  - How do we populate this file?? Entries could still be like this:

```
1.0.0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 PTR    cr1.pop1.isp.net.
```

- Suggestion 1:
  - Delegate loopbacks to their own /64
  - Keeps the loopback zone file separate, and perhaps easier to manage
- Suggestion 2:
  - Make use of the $ORIGIN directive

Example Infrastructure Reverse Zone

```
; Point to Point links
;
$ORIGIN 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.1.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa.
1        PTR     ge0-1.cr1.pop1.isp.net.
2        PTR     ge0-0.br1.pop1.isp.net.
$ORIGIN 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.1.1.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa.
1        PTR     ge0-1.cr1.pop1.isp.net.
2        PTR     ge0-1.br2.pop1.isp.net.
$ORIGIN 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2.1.1.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa.
1        PTR     ge0-1.cr2.pop1.isp.net.
2        PTR     ge0-1.br1.pop1.isp.net.
$ORIGIN 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.3.1.1.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa.
1        PTR     ge0-1.cr2.pop1.isp.net.
2        PTR     ge0-0.br2.pop1.isp.net.
```

Example Loopback Reverse Zone

```
; PoP1
;
$ORIGIN 0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa.
1.0             PTR     cr1.pop1.isp.net.
2.0             PTR     cr2.pop1.isp.net.
3.0             PTR     br1.pop1.isp.net.
4.0             PTR     br2.pop1.isp.net.
0.1             PTR     gw1.pop1.isp.net.
1.1             PTR     gw2.pop1.isp.net.
2.1             PTR     gw3.pop1.isp.net.
3.1             PTR     gw4.pop1.isp.net.
; etc
```

- Note again the use of $ORIGIN and how it keeps the actual lines with the PTR value **simple** for each loopback interface in the PoP

IPv6 DNS

- Previous examples show how to build forward and reverse DNS zone files
  - Forward is easy
  - Reverse can be troublesome unless care is applied and there is a good strategy in place
- There may well be tools out there which help build reverse DNS zone files from IPv6 address databases
  - Long term that will be a better approach!

Dual Stack DNS Conf

- Both Master & Slave
  - DNS software bind-9.7.3.tar.gz [source ftp.isc.org/isc/bind9/9.7.3]
  - BIND root directory [/var/named/chroot] conf file path: /etc/sysconfig/named
  - [named.conf] file path: /var/named/chroot/etc/
  - Zone file path for master zone: /var/named/chroot/var/named/master/
  - Zone file path for slave zone: /var/named/chroot/var/named/slave/
  - Binary executable path: /usr/sbin/
  - Doc file path: /usr/share/doc/bind-9.7*

Dual Stack DNS Conf

- #vi named.conf

```
options
{
        directory "/var/named";
        dump-file               "data/cache_dump.db";
        statistics-file         "data/named_stats.txt";
        memstatistics-file      "data/named_mem_stats.txt";
        listen-on-v6 { any; };
};
acl "slave-server-list" {
        203.176.189.29; 2001:0df0:a:100::1e;
        };
```

Dual Stack DNS Conf

- Split DNS configuration:
  - 3 view need to configure
    - View "localhost_resolver
    - view "internal"
    - view "external"

Dual Stack DNS Conf

- View "localhost_resolver

  view "localhost_resolver"

```
{
match-clients          { localhost; };
match-destinations     { localhost; };
recursion yes;
include "/etc/named.root.hints";
include "/etc/named.rfc1912.zones";
};
```

- * rfc1912zones i.e. localhost, localdomain, 0.0.127 arpa, ::1 ipv6.arpa, 255 arpa, 0 arpa *

Dual Stack DNS Conf

- view "internal”

```
view "internal"
{
match-clients           { localnets; };
match-destinations      { localnets; };
recursion yes;
include "/etc/named.root.hints";
```

Dual Stack DNS Conf

- view "internal”

```
zone "romlab.net" {
type master;
file "master/romlab.net.db";
allow-update        { none; };
allow-transfer { slave-server-list; };
};
```

Dual Stack DNS Conf

- view "internal”

```
zone "189.176.203.in-addr.arpa" {
type master;
file "master/189.176.203.in-addr.arpa.db";
allow-update        { none; };
allow-transfer { slave-server-list; };
};
```

Dual Stack DNS Conf

- view "internal”

```
zone " a.0.0.0.0.f.d.0.1.0.0.2.ip6.arpa" {
type master;
file " master/a.0.0.0.0.f.d.0.1.0.0.2.ip6.arpa.db";
allow-update        { none; };
allow-transfer { slave-server-list; };
};
            };
```

Dual Stack DNS Conf

- view ”external”

```
view "external"
{
    match-clients         { any; };
    match-destinations    { any; };
    recursion no;
allow-query-cache { none; };
```

Dual Stack DNS Conf

- view ”external”

```
zone "romlab.net" {
type master;
file "master/romlab.net.db";
allow-update        { none; };
allow-transfer { slave-server-list; };
    };
```

Dual Stack DNS Conf

- view ”external”

```
zone "189.176.203.in-addr.arpa" {
type master;
file "master/189.176.203.in-addr.arpa.db";
allow-update        { none; };
allow-transfer { slave-server-list; };
};
```

Dual Stack DNS Conf

- view ”external”

```
zone " a.0.0.0.0.f.d.0.1.0.0.2.ip6.arpa" {
type master;
file " master/a.0.0.0.0.f.d.0.1.0.0.2.ip6.arpa.db";
allow-update        { none; };
allow-transfer { slave-server-list; };
};
};
```

Dual Stack DNS Conf

- Zone file ”ipv6.arpa”

```
$TTL    86400
@               IN SOA  ns1.romlab.net. root.romlab.net. (
                                2011032801      ; serial
                                3H              ; refresh
                                15M             ; retry
                                1W              ; expiry
                                1D )            ; minimum
            IN NS       ns1.romlab.net.
            IN NS       ns2.romlab.net.
f.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0           IN PTR      ns1.romlab.net.
e.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0           IN PTR      ns2.romlab.net.
```

Configuring DHCPv6 on Linux

- Server Configuration [dhcp6s]
  - First need install DHCPv6 RPM on the server
    - # yum -y install dhcpv6
  - Enable IPv6 networking and IPv6 forwarding
    - # vi /etc/sysconfig/network

  NETWORKING_IPV6=yes

  IPV6FORWARDING=yes

Configuring DHCPv6 on Linux

- Configure IPv6 on interface
  - # vi /etc/sysconfig/network-scripts/ifcfg-eth0

  IPV6INIT=yes

  IPV6ADDR=”2406:6400:a000::1/64"
- Specify interface for DHCP server
  - # vi /etc/sysconfig/dhcp6s

  DHCP6SIF=eth0

  DHCP6SARGS=

Configuring DHCPv6 on Linux

– Edit the DHCPv6 server configuration file as follows:

  # cp /usr/share/doc/dhcpv6-*/dhcp6s.conf /etc/

  - # vi /etc/dhcp6s.conf

```
interface eth0 {
        server-preference 255;
        renew-time 60;
        rebind-time 90;
```

Configuring DHCPv6 on Linux

```
option dns_servers 2406:6400:800::2 example.com;
link AAA {
pool{
        range 2406:6400:800::20 to 2406:6400:800::40/64;
        prefix 2406:6400:800::/64;
                    };
        };
};
```

Start DHCPv6 server daemon:

```
# service network restart && service dhcp6s start && chkconfig dhcp6s on
```

Unix Webserver

- Apache 2.x supports IPv6 by default
- Simply edit the httpd.conf file
  - HTTPD listens on all IPv4 interfaces on port 80 by default
  - For IPv6 add:

    ```
    Listen [2001:db8:10::1]:80
    ```

    - So that the webserver will listen to requests coming on the interface configured with 2001:db8:10::1/64

Unix Sendmail

- Sendmail 8 as part of a distribution is usually built with IPv6 enabled
  - But the configuration file needs to be modified
- Then edit /etc/mail/sendmail.mc thus:
  - Remove the line which is for IPv4 only and enable the IPv6 line thus (to support both IPv4 and IPv6):
  - DAEMON_OPTIONS(‘Name=IPv4, Family=inet’ Addr=203.176.189.2’)dnl
  - DAEMON_OPTIONS(‘Name=IPv6, Family=inet6, Addr=3ffe:b00:1:1::1’)dnl
  - configuration files such as mailertable, access, and relay-domains
  - IPV6:3ffe:b00:1:1::1
  - Remake sendmail.cf, then restart sendmail

FTP Server

- Vsftpd is discussed here
  - Standard part of many Linux distributions now
- IPv6 is supported, but not enable by default
  - Need to run two vsftpd servers, one for IPv4, the other for IPv6
- IPv4 configuration file: /etc/vsftpd/vsftpd.conf

```
listen=YES
listen_address=<ipv4 addr>
```

- IPv6 configuration file: /etc/vsftpd/vsftpdv6.conf

```
listen=NO
listen_ipv6=YES
listen_address6=<ipv6 addr>
```

IPv6@APNIC

Your IP address:
203.119.42.199
Contact us | Press | Jobs | Site map
Search...
Go
Home
Services
Community
Events
Publications
About us
Login to MyAPNIC
Community
Print this page
Policy development
Participate
Working with the community
About the Internet community
IPv6@APNIC
Key IPv6 messages
IPv6 data and statistics
IPv6 Transition Stories
IPv6 Best Current Practices
IPv4 exhaustion
IPv6@APNIC
IPv6 is a top issue for the Asia Pacific Internet community. APNIC engages in activities throughout the region to help facilitate a smooth transition. The greater goal is to support the Asia Pacific in deploying IPv6 to maintain a scalable Internet for everyone.
APNIC reached the last /8 of IPv4 addresses in April 2011, and now delegates IPv4 resources according to the "last /8 policy". The scarcity of IPv4 makes IPv6 deployment critical for all networks and organizations in the Asia Pacific. Here's what APNIC is doing to support the community in achieving real and tangible IPv6 deployment:
Get your IPv6 addresses!
Distributing IPv6 addresses
Getting an IPv6 block is the first step in your transition, and the process is very simple.
Kickstart IPv6 - one click to IPv6
IPv6 training and education
Is your technical staff ready to deploy IPv6? Gaining technical knowledge does not happen overnight. Plan and implement training for your personnel. APNIC Training is constantly updating our IPv6 content, to reflect the industry's best current practices.
Upcoming training events
Related links
IPv6 news feed
IPv6 Info
Curated by APNIC
MicroNugget: 3 Basic Tasks For Building an IPv6 Network
Scoop.it!
IPv4 Exhaustion Counter
Present Status (RIR)
X-day and Reserved Blocks (Remaining /8)
AfriNIC
Jan 22, 2021 3.03
APNIC
Apr 15, 2011 0.89
ARIN
Jun 13, 2014 5.52
LACNIC
Oct 01, 2014 2.37
RIPE NCC

Wrapping up…

- Readings
  - Enterprise IPv6 Deployment Guidelines
  - http://tools.ietf.org/html/draft-ietf-v6ops-enterprise-incremental-ipv6-02
  - IPv6 Guidance for Internet Content Providers and Applicatoin Service Providers
  - http://tools.ietf.org/html/rfc6883

APNIC Helpdesk Chat

Questions?

Thank You