IPv6 overview

Florent Parent

Florent.Parent@viagenie.qc.ca

Viagénie

http://www.viagenie.qc.ca

LINKING CANADA'S INTERNET INNOVATORS | CARREFOUR DES INNOVATEURS CANADIENS DE L'INTERNET

Plan

- Who are we ?
- Features
- Header
- Neighbor discovery protocol
- Data links
- Transition mechanisms
- CA*net3
- Configuring IPv6 on Cisco routers

- Consulting firm specialized in:
  - Network security, audits, PKI,...
  - Advanced IP networks:
    - IPv6
    - OBGP
  - Voice/Video/Telephony over IP
  - Internationalisation
    - Co-chair of IETF internationalized domain names (idn working group)

IPv6 Features

Plan

- Larger address space
- Efficient IP header and datagram
- Mandatory features

Larger Address Space

- From 32bits to 128 bits addresses enables:
  - Global reachability:
    - no hidden networks, hosts.
    - All hosts can be reachable and be "servers".
    - End-to-End security can be used.
  - Flexibility
    - Multiple levels of hierarchy in the address space.
  - Autoconfiguration
    - Use of 64 bits for link-layer address encapsulation with warranty of uniqueness.

Larger Address Space (cont.)

- "Plug and play"
  - By autoconfiguration.
- Aggregation
  - Multiple prefixes for the same site enables multihoming without cutting holes in the aggregation.
- Multihoming
- Renumbering
  - By using autoconfiguration and multiple prefixes, renumbering becomes doable.

Efficient and Extensible IP Datagram

- Less number of fields enables:
  - Routing efficiency
  - Performance
  - Forwarding rate scalability
- Extensibility of header
  - Better handling of options
  - No checksum
- 64 bits aligned.
- Flow label.

Mandatory Features

- Security
- Mobility
  - Much more optimized in IPv6 than MobileIPv4.
- Multicast use:
  - No broadcast
  - Efficient use of the network and less interrupts on NICs.
  - Scoped groups.
- Transition richness
  - Seamless transition
  - Software change
  - Mechanisms and tools for IPv4-IPv6 interaction

Summary

- Larger address space enables many new features.
- Cleaner and more efficient header.
- Mandatory features.

References

- Internet Protocol, Version 6 (IPv6) Specification , RFC 2460, December 1998.

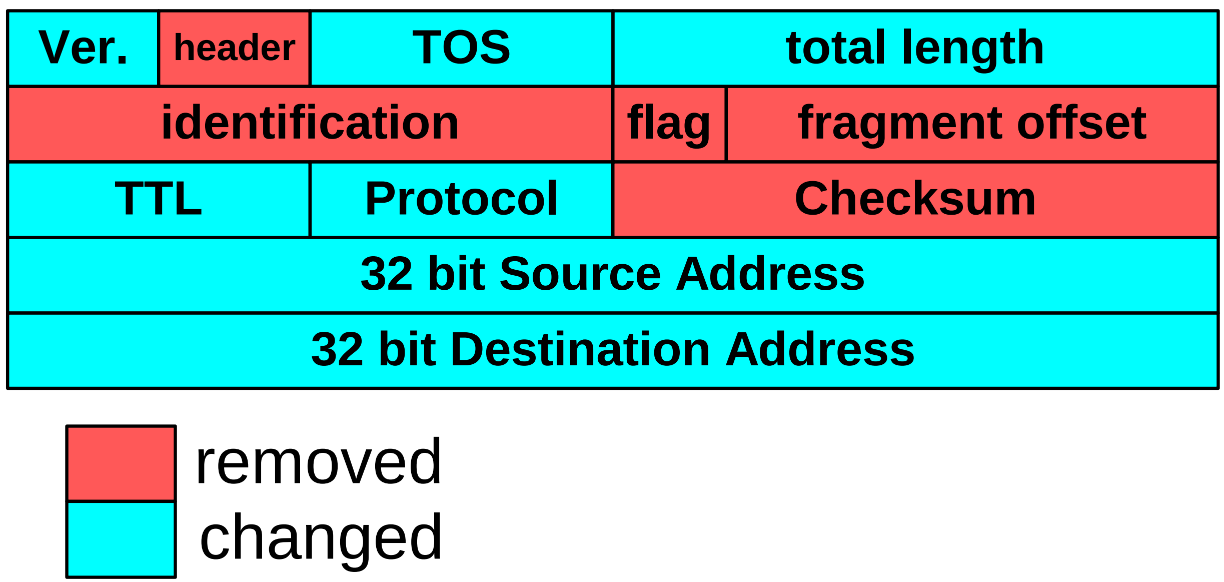
IPv6 Header

Plan

- IPv4 Header
- IPv6 Header
- Header Fields
- Extension Headers
- Routing Header

IPv4 Header

- IPv4 header = 20 bytes without options.



IPv6 Header

- IPv6 header = 40 bytes without extensions.

<table>
<tr><td>Ver.</td><td>TrafficClass</td><td colspan="3">Flow Label</td></tr>
<tr><td colspan="3">Payload Length</td><td>Next Header</td><td>Hop Limit</td></tr>
<tr><td colspan="5">128 bit
Source Address</td></tr>
<tr><td colspan="5">128 bit
Destination Address</td></tr>
</table>

Header Fields

- Version (4 bits)
  - 6 for IPv6
- Traffic Class (8 bits)
  - ~= TOS in IPv4
  - Identifies and distinguishes between different classes or priorities (diffserv)
- Flow Label (20 bits)
  - Use is not yet fully defined.
  - Used by a source node to label sequences of packets
- Payload Length
  - ~= Total length in IPv4

Next Header Field

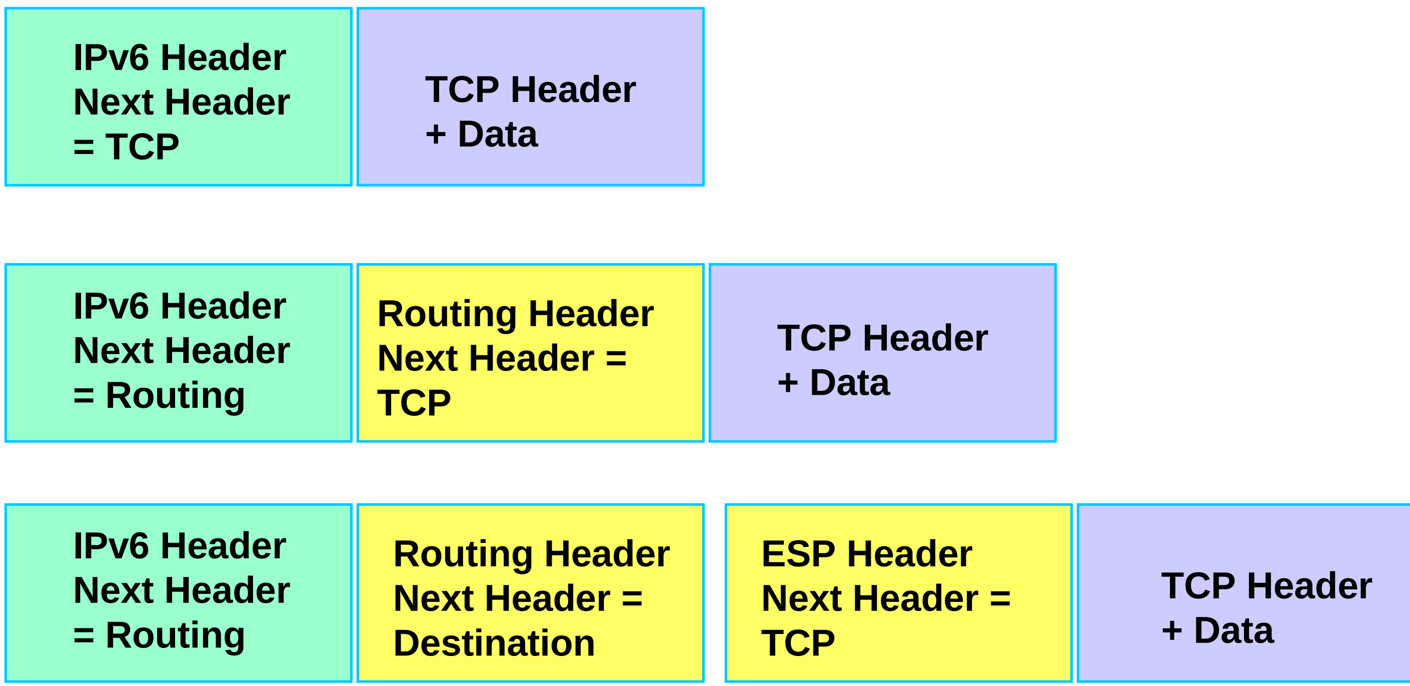
- Next Header (8 bits)
  - ~= Protocol field in IPv4
  - Used to identify the encapsulated protocol
    - TCP, UDP
    - ESP, AH (confidentiality and authentification in IPsec)
    - ICMPv6
    - Other extensions

Header Fields (cont.)

- Hop Limit ~= TTL in IPv4
- MTU must be at least 1280 bytes (1500+ recommended). Nodes should use Path MTU discovery.
- UDP checksum required

Extension Headers

- New way of doing options.
- Added after the basic IPv6 header



Extension Headers

- Daisy chained
  - Hop-by-hop options (0)
    - information that must be examined by every node along the path.
  - Routing (43)
    - similar to IPv4's Loose Source and Record Route option. Used also in mobileIPv6
  - Fragment (44)
    - used by source node (routers don’t fragment anymore !)
  - Destination options (60)
    - used to carry optional information that need to be examined only by a packet's destination node(s)
    - Used in mobileIPv6.

Summary

- Comparaison of IPv4 and IPv6 headers show a longer header, but less number of fields.
- Header processing is simpler.
- Options are handled by extension headers.

IPv6 Addressing

Plan

- IPv6 addresses
- Format
- Unicast
- Multicast
- Anycast
- Required Node Addresses
- Address Selection
- Addressing Architecture

Addresses

- IPv4 = 32 bits
- IPv6 = 128 bits.
  - This is not 4 times the number of addresses.
  - This is 4 times the number of bits.
  - ~3,4 * 10^{38} possible addressable nodes
  - 10^{30} addresses per person on the planet
  - Well, as with any numbering scheme, we will be using only a portion of the full address space.

Address Format

- x:x:x:x:x:x:x:x
  - Where x is a 16 bit hexadecimal field
    - 2001:0000:1234:0000:0000:C1C0:ABCD:0876
- Case insensitive
    - 2001:0000:1234:0000:0000:<u>c1c0</u>:<u>abcd</u>:0876
- Leading zeros in a field are optional:
    - 2001:<u>0</u>:1234:<u>0</u>:<u>0</u>:C1C0:ABCD:<u>876</u>

Address Format

- Successive fields of 0 are represented as ::, but only once in an address:
  - 2001:0:1234::C1C0:ABCD:876
  - Not valid: 2001::1234::C1C0:ABCD:876
- Other examples:
  - FF02:0:0:0:0:0:0:1 => FF02::1
  - 0:0:0:0:0:0:0:1 => ::1
  - 0:0:0:0:0:0:0:0 => ::

Addresses in URL

- In a URL, it is enclosed in brackets
  - http://[2001:1:4F3A::206:AE14]:8080/index.html
  - URL parsers have to be modified
  - Cumbersome for users
    - Mostly for diagnostic purposes
    - Should use Fully Qualified Domain Names (FQDN)

Address Types

- Unicast
  - Unspecified
  - Loopback
  - IPv4 compatible
  - Scoped addresses:
    - Link-local
    - Site-local
  - Aggregatable Global:
- Multicast
  - Broadcast: none in IPv6
- Anycast

Unspecified

- Used as a placeholder when no address available
  - initial DHCP request
  - Duplicate Address Detection (DAD)
- Like 0.0.0.0 in IPv4
- 0:0:0:0:0:0:0:0 or ::

Loopback

- Identifies self
- Localhost
- Like 127.0.0.1 in IPv4
- 0:0:0:0:0:0:0:1 or ::1
- To find if your IPv6 stack works:
  - Ping6 ::1

IPv4 Compatible

- Embedded mechanism in IPv6 stack
- To do automatic tunnels
- Format:
  - 0:0:0:0:0:0:<ipv4 address>
  - ::<ipv4 address>
  - Example:
    - ::192.168.1.1 (same as ::C0A8:0101)
- Usage:
  - ping6 ::192.168.1.1
  - Will build an IPv6 over IPv4 tunnel with IPv4 destination = 192.168.1.1
- Deprecated mechanism

Link-local

- Scoped address (new in IPv6)
- Scope = local link (i.e. VLAN, subnet)
  - Can only be used between nodes of the same link
  - Cannot be routed
- Automatically configured on each interface
  - Uses the interface identifier (based on MAC address)
- Format:
  - FE80:0:0:0:<interface identifier>
- Gives every node an IPv6 address to start communications

Site-local

- Scoped address
- Scope = site (a network of links)
  - Can only be used between nodes of the same site
  - Cannot be routed outside the site (i.e. the Internet)
  - Very similar to IPv4 private addresses
- Not configured by default

Site-local (cont.)

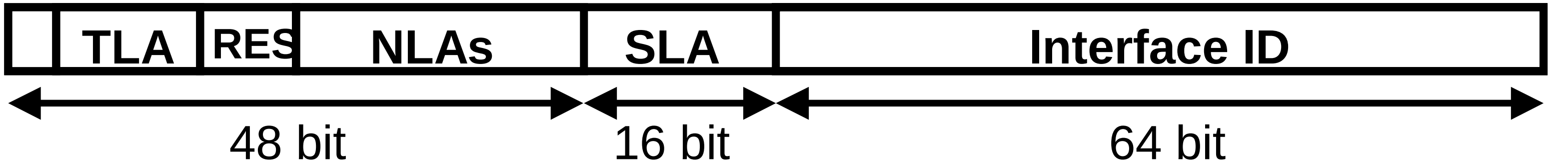
- Format:
  - FEC0:0:0:<subnet id>:<interface id>
  - Subnet id = 16 bits = 65K subnets.
    - Enables an addressing plan for a full site
- Usage example:
  - Number a site before connecting to the Internet:
    - Do your address plan using site locals and use the renumbering functions when connecting to the IPv6 Internet.
  - Private addresses (eg. local printers)

Aggregatable Global

- Generic use. Globally reachable.
- Allocated by IANA
  – To Regional Registries
  – Then to Tier-1 Providers
    - Called Top-level Aggregator (TLA)
  – Then to Intermediate Providers
    - Called Next-level Aggregator (NLA)
  – Then to sites
  – Then to subnets

Aggregatable Global

- Structure:



- 128 bits as the total
- 48 bits prefix to the site
- 16 bits for the subnets in the site
- 64 bits for host part

Aggregatable Global (cont.)

- Consists of the following (left to right):
  - 3 bits: 001 (10% of the total address space reserved)
  - 13 bits for the TLA
    - 2^{13} TLAs ~ 8K TLAs
  - 8 bits reserved
  - 24 bits for the NLAs
    - 2^{24} NLAs per TLA ~ 16M NLAs per TLA
  - 16 bits for the site subnets
    - 2^{16} subnets per site = 65536 subnets
  - 64 bits for the interface identifier
  - Total = 128 bits.

Multicast

- Multicast = one-to-many.
- No broadcast in IPv6. Multicast is used instead, mostly on local links.
- Scoped addresses:
  - Node, link, site, organisation, global
  - No TTL as in IPv4
- Format:
  - FF<flags><scope>::<multicast group>

Multicast Assigned Addresses

- Some reserved multicast addresses:

Address	Use	Scope
FF01::1	All Nodes	Node-local
FF02::1	All Nodes	Link-local
FF01::2	All Routers	Node-local
FF02::2	All Routers	Link-local
FF05::2	All Routers	Site-local
FF02::1:FFXX:XXXX	Solicited-Node	Link-local

Anycast

- One-to-nearest: great for discovery functions.
- Anycast address are indistguishable from unicast address
  - Allocated from the unicast address space
  - Some anycast addresses are reserved for specific uses
- Few uses:
  - router-subnet,
  - mobileIPv6 home-agent discovery,
  - discussions for DNS discovery.

Required Node Addresses

- Any IPv6 node should have the following addresses enabled:
  - Link-local address for each interface
  - Loopback address
  - Assigned unicast addresses
  - All-nodes multicast address
  - Solicited-node multicast address for each of its assigned unicast and anycast address
  - Multicast address of all other groups to which the host belongs
  - Site-local address if used

Required Router Addresses

- Any IPv6 router should have the following addresses enabled:
  - All the required node addresses
  - All-routers multicast addresses
  - Specific multicast addresses for routing protocols
  - Subnet-router anycast addresses for the interfaces configured to act as forwarding interfaces
  - Other anycast configured addresses

Address Selection

- A node have many IPv6 addresses.
- Which one to use as source and destination address for a given communication?
- Some issues to be addressed:
  - Scoped addresses are unreachable depending on the destination.
  - Preferred vs deprecated addresses.
  - IPv4 or IPv6 when DNS returns both.
  - IPv4 local scope (169.254/16) and IPv6 global scope. IPv6 local scope and IPv4 global scope.
  - MobileIP addresses, temporary addresses, scope addresses, etc.

Address Selection

- Algorithm:
  - Prefer same address.
  - Prefer appropriate scope.
  - Avoid deprecated addresses.
  - Prefer home addresses.
  - Prefer outgoing interface.
  - Prefer matching label.
  - Prefer temporary addresses.
  - Use longest matching prefix.

Address Selection

- Algorithm is basically:
  - Use the right scope based on the destination.
  - Use the most similar address.
  - Use home address instead of care-of-address, if in the mobility context.
- Default policy can be overridden by stack or application.

Addressing Architecture

Prefix	Hex	Size	Allocation
0000 0000	0000-00FF	1/256	Reserved
0000 0001	0100-01FF	1/256	Unassigned
0000 001	0200-03FF	1/128	NSAP
0000 010	0400-05FF	1/128	Unassigned
0000 011	0600-07FF	1/128	Unassigned
0000 1	0800-0FFF	1/32	Unassigned
0001	1000-1FFF	1/16	Unassigned
001	2000-3FFF	1/8	Aggregatable: IANA to registries

Addressing Architecture (cont.)

Prefix	Hex	Size	Allocation
010, 011, 100, 101, 110	4000-CFFF	5 * 1/8 = 5/8	Unassigned
1110	D000-EFFF	1/16	Unassigned
1111 0	F000-F7FF	1/32	Unassigned
1111 10	F800-FBFF	1/64	Unassigned
1111 110	FC00-FDFF	1/128	Unassigned
1111 1110 0	FE00-FE7F	1/512	Unassigned
1111 1110 10	FE80-FEBF	1/1024	Link-local
1111 1110 11	FEC0-FEFF	1/1024	Site-local
1111 1111	FF00-FFFF	1/256	Multicast

Summary

- IPv6
  - has a much larger address space.
  - Has specific formatting for addresses.
  - Introduces new kind of addresses (scoped).
- IPv6 nodes have many addresses and need to select which one to use.
- Addressing architecture has a lot of space available for future use.

References

- *IP version 6 Addressing Architecture*, Hinden and Deering, RFC2373, July 1998, http://www.normos.org/ietf/rfc/rfc2373.txt
- *An IPv6 Aggregatable Global Unicast Address Format* , Hinden, O'Dell and Deering, RFC 2374, July 1998, http://www.normos.org/ietf/rfc/rfc2374.txt
- *Default Address Selection for IPv6*, Richard Draves, IETF internet-draft, May 2001, http://www.normos.org/ietf/draft/draft-ietf-ipngwg-default-addr-select-04.txt

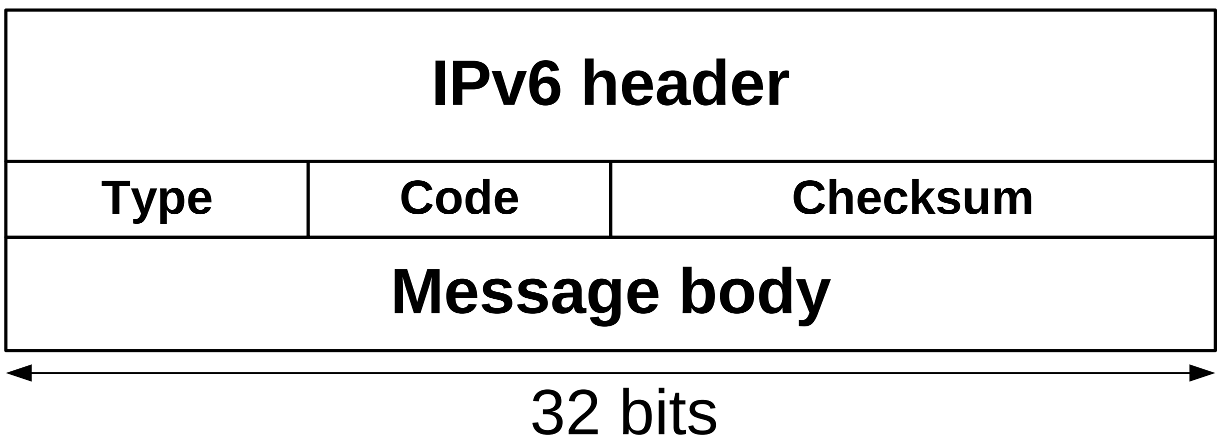
ICMP, Neighbor Discovery and Autoconfiguration

Plan

- ICMP
- Path MTU Discovery
- Neighbor Discovery
- Autoconfiguration
- Renumbering
- Duplicate Address Detection
- Temporary Addresses

ICMPv6

- Internet Control Message Protocol
- Same behavior as in IPv4, but few enhancements.
- IPv6 Next Header= 58



ICMPv6 messages

- Many messages are the same as the IPv4 counterpart:
  - Type 1: Destination Unreachable
  - Type 2: Packet Too Big (MTU)
  - Type 3: Time Exceeded
  - Type 4: Parameter Problem
  - Type 128/129: Echo request/Echo reply

Path MTU Discovery

- No fragmentation done by routers in IPv6.
- Fragmentation, if needed, is done by the source.
- Source should do Path MTU Discovery to find the right MTU.
- Minimum MTU is 1280.

PMTU Discovery Process

1. Send a message to the destination with MTU of your link.
2. If receive a ICMP error message, then resend the message with the new MTU.
3. Do 1 and 2 until response from destination.
4. Last MTU is the Path MTU.

- Solicited-Node multicast address
  - **FF02:0:0:0:0:1:FF00::/104**
    - address formed by appending the lower 24 bits of the IPv6 address
    - a node is required to join for every unicast and anycast address it is assigned

3FFE:0B00:0C18:0001:0290:27FF:FE17:FC0F

Global unicast address

FF02:0000:0000:0000:0000:0001:FF17:FC0F

Solicited multicast address

Neighbor Discovery

- Replaces IPv4 ARP, plus new features.
- Uses ICMPv6 messages.
- Used to:
  - Find link-layer address of neighbor.
  - Find neighboring routers.
  - Actively keep track of neighbor reachability.
  - Send network information from routers to hosts.
- Protocol used for host autoconfiguration
- All ND messages must have Hop Limit=255
  - Must originate and terminate from the same link

Neighbor Solicitation

- Sent by a node to determine link-layer address of a neighbor.
- =~ ARP request.
- Packet description:
  - Source Address = link-local address
  - Destination = solicited-node multicast address.
  - Data contains link-layer address of source. (for efficiency purposes)
  - Query is: give me your link-layer address?
  - ICMP type 135.

Neighbor Advertisement

- Response to a Neighbor Solicitation.
- =~ ARP response.
- Includes my MAC address so you can send me information.
- Packet description:
  - Source Address = link-local address of source
  - Destination = destination address
  - Data contains link-layer address of me.
  - ICMP type 136.

Router Advertisement

- Routers advertise periodically
  - Max. time between advertisements can be in the range from 4 and 1800 seconds
  - The advertisement has a lifetime
- Advertisement contains one or more prefixes
  - Prefixes have a lifetime
    - Preferred lifetime
    - Valid lifetime
- Specifies if stateful or stateless autoconfiguration is to be used
- Plays a key role in site renumbering

Router Advertisement (cont.)

- Packet description:
  - Source: router address on the link.
  - Destination: multicast address of all nodes on the link (FF02::1)
  - Data: prefix, lifetimes, default router, options.
  - ICMP type 134

Router Solicitation

- When booting, nodes don't want to wait until the next router advertisement to configure themselves.
- Host then request routers to send Router Advertisement immediately.
- Packet description:
  - Source: link-local address
  - Destination: multicast address all-routers on this link (FF02::2)
  - ICMP type 133

Redirect

- ~= ICMP redirect
- Route change
- Router send better hop for a destination
- Packet description:
  - Source: router address
  - Destination: source node of the packet that needs rerouting.
  - Data: better router address.

Autoconfiguration

- Stateful configuration
  - Manual IP configuration
  - DHCP configuration
- Stateless Address Autoconfiguration
  - Applies to hosts only (not to routers)
  - No manual configuration required
    - specifies the prefix, default route and lifetime
    - but does not specify the DNS servers
  - Assumes interface has unique identifier
  - Assumes multicast capable link
  - Uses Duplicate Address Detection

Autoconfiguration Process

- Host configured for autoconfiguration.
- Host boots. Sends a Router Solicitation.
- Host receives the Router Advertisement, specifying subnet prefix, lifetimes, default router, ...
- Host generates its IP address by appending:
  - Received subnet prefix (64 bits)
  - Interface address modified for EUI-64 format
- Host verify usability of the address by doing the Duplicate Address Detection process.

Duplicate Address Detection

- Similar to IPv4 ARP self.
- Join all-nodes multicast address (FF02::1)
- Join solicited-node multicast address of the tentative address.
  - FF02:0:0:0:0:1:FFxx: <last 24 bits of my new address>
- Send Neighbor Solicitation on solicited-node multicast address.
- If no Neighbor Advertisement is received, address is ok.

Renumbering

- Hosts renumbering
  - On the router, decrease the lifetime of the prefix in the router advertisement.
    - Preferred lifetime = 0. (this "old" address cannot be used for new connections.)
    - Valid lifetime decreasing.
  - Start advertising the new prefix.
  - Hosts configure the new address and start using it.
- Connections will continue without interruption.
- Hosts must always listen to router advertisements, even after autoconfiguration.

Router Renumbering

- Router Renumbering
  - Protocol to renumber routers within a site
  - Defines new ICMPv6 messages
  - Uses IPsec for authentication purposes.
- At this time, no implementation known.

Temporary Addresses

- Tracability of the MAC address in IPv6 address, if using autoconfig all time.
- Privacy concerns
- Algorithm (RFC3041) defined to:
  - Generate a random address for the rightmost 64 bits.
  - Define it as temporary
  - Recycle it as needed

Summary

- ICMPv6 is similar to IPv4, but is enhanced for Neighbor Discovery functions.
- Path MTU discovery is used.
- Neighbor discovery enables ARP like functions and autoconfiguration.
- Duplicate address detection is used to ensure uniqueness of addresses on link.
- Renumbering is acheived by modifying the advertisements of prefixes.
- Temporary addresses are used in case of privacy concerns.

References

- Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) , A. Conta, S. Deering, RFC 2463, December 1998.
- Path MTU Discovery for IP version 6, J. McCann, S. Deering, J. Mogul, RFC1981, August 1996.
- Neighbor Discovery for IP Version 6 (IPv6) , T. Narten, E. Nordmark, W. Simpson, RFC2461 , December 1998.

References

- IPv6 Stateless Address Autoconfiguration, S. Thompson, T. Narten, RFC2462 , December 1998.
- Privacy Extensions for Stateless Address Autoconfiguration in IPv6 , T. Narten, R. Draves, RFC3041 January 2001.
- Router Renumbering for IPv6 , M. Crawford, RFC2894 , August 2000.

IPv6 Over Data Link

Plan

- IPv6 over Ethernet
- Ethernet Frame ID
- EUI-64 format
- Multicast over Ethernet
- IPv6 over other Data Link Layers

IPv6 over Ethernet

- Interface identifier for stateless autoconfiguration
  - EUI-64 interface identifier
- Specific frame id for IPv6 (different from IPv4)
  - IPv4-only stacks won't process those packets
- Multicast IPv6 is mapped to Multicast Ethernet

Ethernet Frames

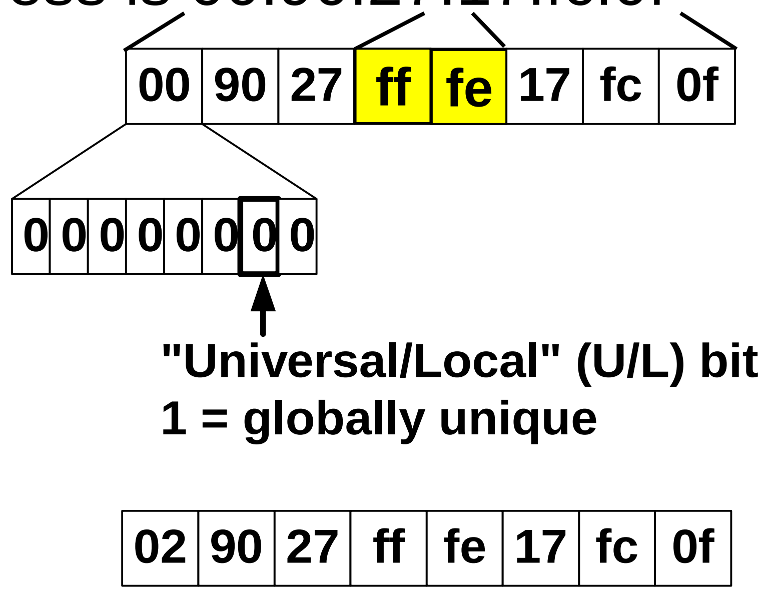
Frame Format

Destination Ethernet
Source Ethernet
86DD
IPv6 header and payload

- IPv6 unicast mapping over Ethernet
  - Uses Neighbor Solicitation to get link-layer address

EUI-64

- Interface Identifier for stateless autoconfiguration
  - Ethernet MAC address is 00:90:27:17:fc:0f



The EUI-64 **0290:27ff:fe17:fc0f** is used as the interface identifier (last 64 bits) in the IPv6 address

Ethernet with Multicast

- IPv6 multicast address mapping over Ethernet

FF02:0000:0000:0000:0000:0001:FF17:FC0F

Solicited multicast address

33-33-FF-17-FC-0F

Ethernet address

IPv6 over Data Link layers

- IPv6 is defined for most data link layers:
  - Ethernet
  - FDDI
  - Token Ring
  - Arcnet
  - PPP
  - Non-Broadcast Multiple Access (NBMA)
  - ATM
  - Frame Relay
  - ...

Summary

- IPv6 is available on most link-layer technologies.
- Specificity of each link-layer is taken care.
- Ethernet MAC address is used as EUI-64 interface identifier.
- Ethernet Frame ID is different for IPv6 than IPv4.
- Multicast IPv6 is handled in Ethernet Multicast.

References

- *Tranmission of IPv6 Packets over Ethernet Networks*, Crawford, IETF RFC2464, December 1998, http://www.normos.org/ietf/rfc/rfc2464.txt
- *Tranmission of IPv6 Packets over FDDI Networks*, Crawford, IETF RFC2467, December 1998, http://www.normos.org/ietf/rfc/rfc2467.txt
- *Tranmission of IPv6 Packets over Token Ring Networks*, Crawford, Narten and Thomas, IETF RFC2470, December 1998, http://www.normos.org/ietf/rfc/rfc2470.txt
- *Transmission of IPv6 Packets over ARCnet Networks*, Souvatzis, IETF RFC2497, January 1999, http://www.normos.org/ietf/rfc/rfc2497.txt

References

- *IP Version 6 over PPP*, Hasken and Allen, IETF RFC2472, December 1998, http://www.normos.org/ietf/rfc/rfc2472.txt
- *IPv6 over Non-Broadcast Multiple Access (NBMA) networks*, Armitage et al, IETF RFC2491, January 1999, http://www.normos.org/ietf/rfc/rfc2491.txt
- *IPv6 over ATM Networks*, Armitage, Schulter and Jork, IETF RFC2492, January 1999, http://www.normos.org/ietf/rfc/rfc2492.txt
- *Transmission of IPv6 Packets over Frame Relay Networks Specification*, Conta, Malis and Mueller, IETF RFC2590, May 1999, http://www.normos.org/ietf/rfc/rfc2490.txt

IPv6 Transition Tools and Mechanisms

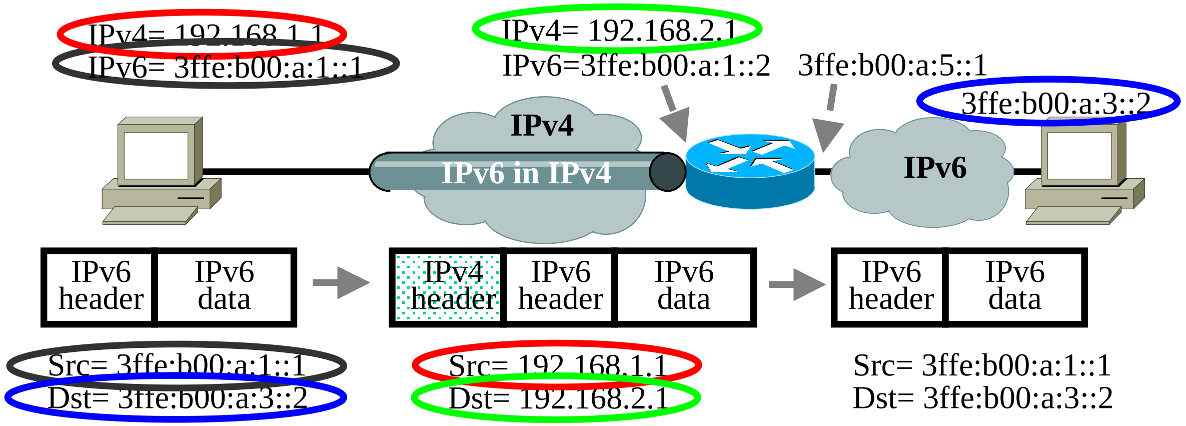
Plan

- Dual Stack
- Tunnelling over IPv4
  - Configured tunnels
  - Automatic tunnelling
    - 6to4
    - Tunnel Broker/server

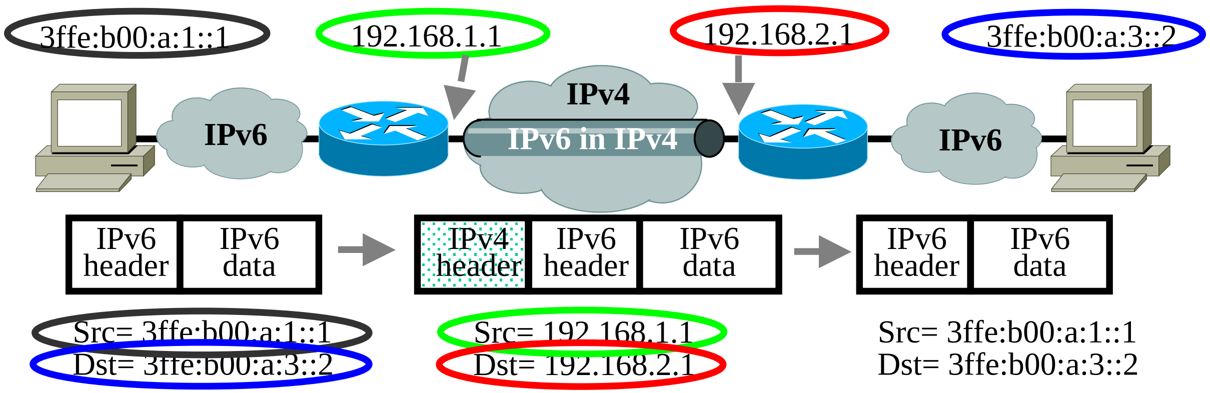
Configured Tunnels

- Manually configured:
  - IPv4 source and destination addresses
  - IPv6 source and destination addresses
- Between:
  - Two hosts
  - One host and one router
  - Two routers

Configured Tunnels

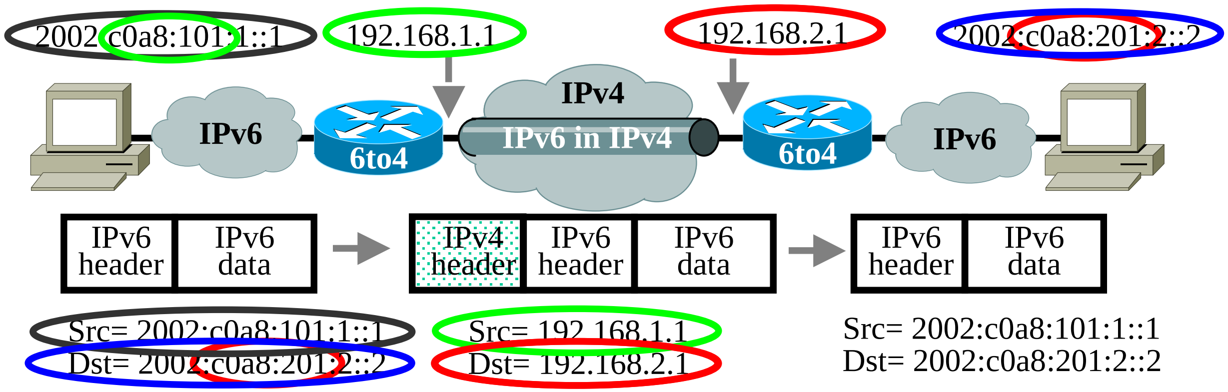


Configured Tunnels

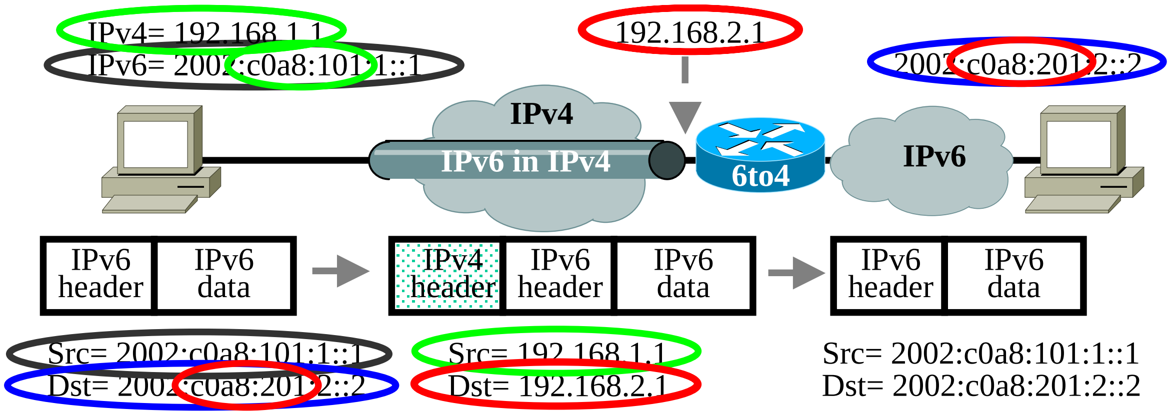


6to4

- Automatic establishment of the tunnel
- Under the 2002:/16 prefix.
- Format: 2002:<ipv4add>:<subnet>::/64
- IPv4 external address embedded: 2002:<ipv4 ext address>::/48
- Gives a full /48 to a site based on its external IPv4 address.


2002:c0a8:101:1::1
192.168.1.1
192.168.2.1
2002:c0a8:201:2::2
IPv6
6to4
IPv4
IPv6 in IPv4
6to4
IPv6
IPv6 header
IPv6 data
IPv4 header
IPv6 header
IPv6 data
IPv6 header
IPv6 data
Src= 2002:c0a8:101:1::1
Dst= 2002:c0a8:201:2::2
Src= 192.168.1.1
Dst= 192.168.2.1
Src= 2002:c0a8:101:1::1
Dst= 2002:c0a8:201:2::2

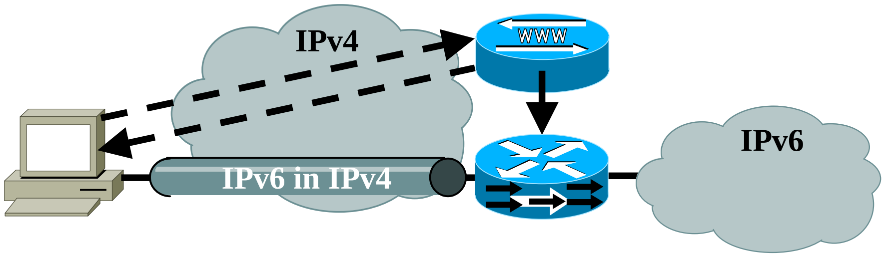
6to4 Host



IPv4= 192.168.1.1
IPv6= 2002:c0a8:101:1::1
192.168.2.1
2002:c0a8:201:2::2
IPv4
IPv6 in IPv4
6to4
IPv6
IPv6 header
IPv6 data
IPv4 header
IPv6 header
IPv6 data
IPv6 header
IPv6 data
Src= 2002:c0a8:101:1::1
Dst= 2002:c0a8:201:2::2
Src= 192.168.1.1
Dst= 192.168.2.1
Src= 2002:c0a8:101:1::1
Dst= 2002:c0a8:201:2::2

Tunnel Broker

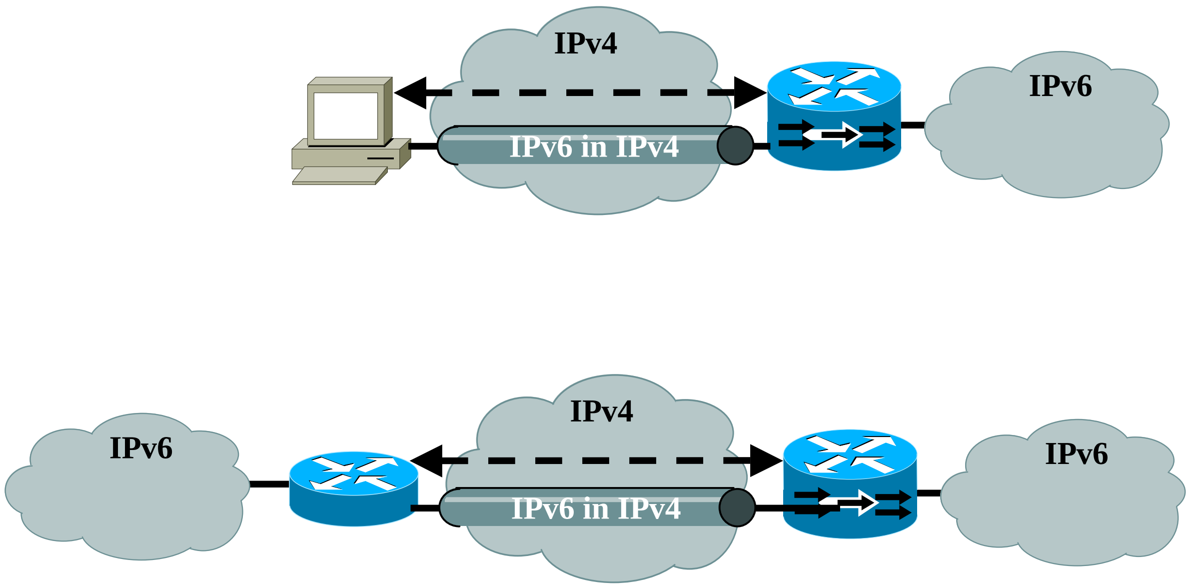
- Semi-automated
- Broker
  - Is a web server receiving requests from clients
  - Generates the tunnel and sends back info to client
  - Configures the server


IPv4
WWW
IPv6 in IPv4
IPv6

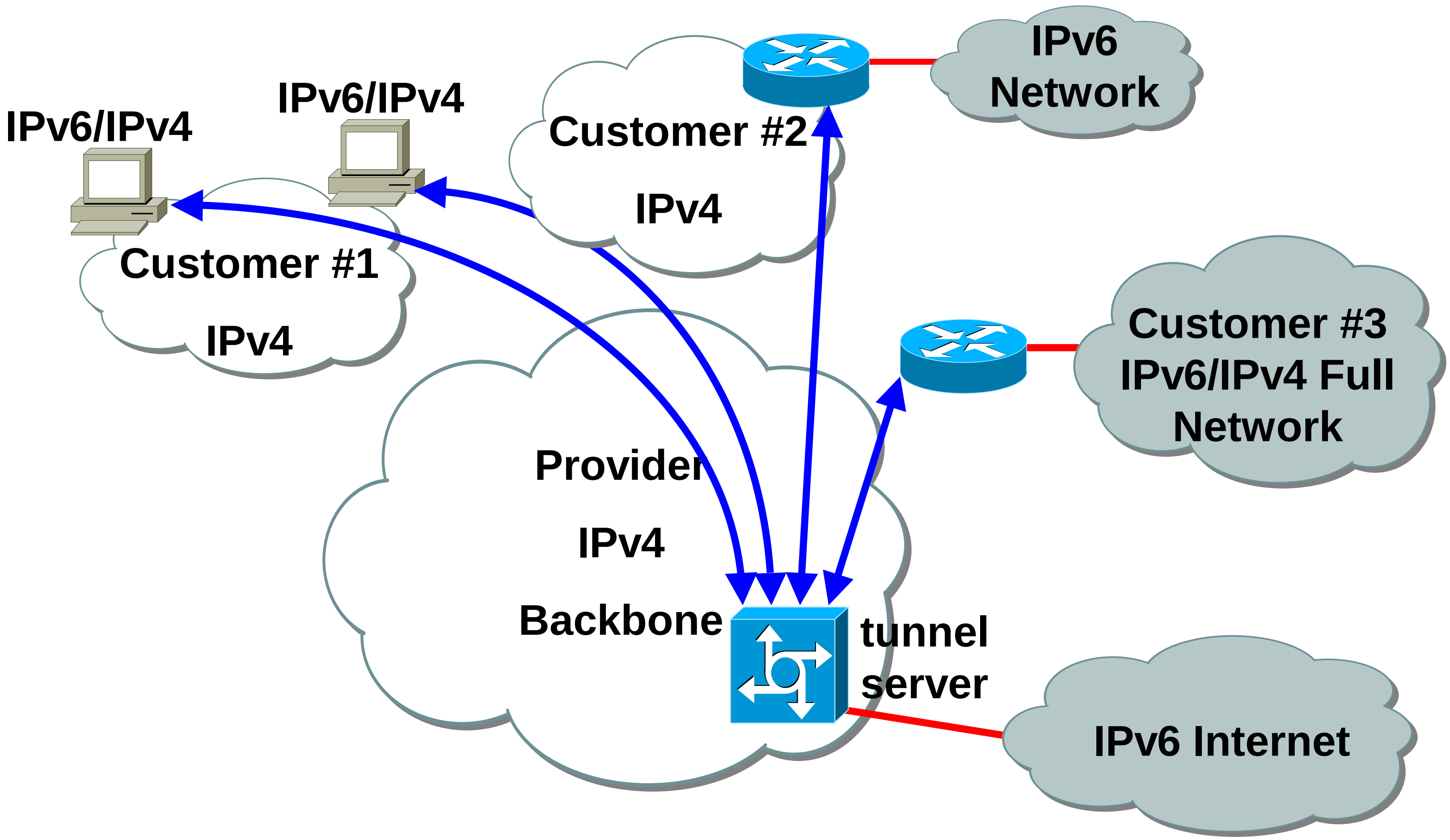
Tunnel Server

- Semi-automated
- Server:
  - Receives requests from clients (web, xml)
  - Generates a tunnel
  - Sends back the tunnel info to the client
  - Configures itself as end point of the tunnel

Tunnel Server



Provider


IPv6/IPv4
IPv6/IPv4
Customer #1
IPv4
Customer #2
IPv4
IPv6
Network
Customer #3
IPv6/IPv4 Full
Network
Provider
IPv4
Backbone
tunnel
server
IPv6 Internet

Tunnel Server Implementation

- http://www.Freenet6.net
- Now implements:
  - Control channel
    - Automation of connection at boot phase
    - Client may change its IPv4 configuration
  - Permanent tunnels
  - Authentication
  - Prefix delegation
    - Everybody requesting it will receive a /48.
    - This is 64K networks for any individual!
  - Same client support: Windows, Linux, *BSD, Solaris, Cisco. Others to come.
- Available as a service.
  - Still free and managed on best effort.

Useful Reading

- A Guide to the Introduction of IPv6 in the IPv4 World
  - draft-ietf-ngtrans-introduction-to-ipv6-transition-06.txt
  - Lists "all" transition mechanisms with a small description. Try to classify them
  - Other information for IPv6 deployment

Summary

- Many transition tools exists:
  - To tunnel between IPv6 islands
  - To translate between IPv4 and IPv6
- Others are available, others will be defined.
- None is for all possible scenarios.
- Not all will succeed on the market.
- Choose the right one for your scenario.

References

- Transition Mechanisms for IPv6 Hosts and Routers, RFC 2893.
- Transmission of IPv6 over IPv4 Domains without Explicit Tunnels (6over4), RFC 2529.
- Connection of IPv6 Domains via IPv4 Clouds (6to4), RFC3056.
- Network Address Translation—Protocol Translation (NAT-PT), RFC 2766.
- Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS), RFC 2767.
- draft-ietf-ngtrans-introduction-to-ipv6-transition-04.txt, An overview of the introduction of IPv6 in the Internet
- IPv6 Tunnel Broker, RFC 3053.

IPv6 on CA*net3

Viagénie

CA*Net3

- First DWDM backbone.
- Cisco GSRs with Nortel DWDM gear are used.
- IPv6:
  - No support of IPv6 really available yet on GSRs (only process switched :-(( )
    - Have to tunnel over.
  - Peering at the 6tap and with many others.
  - Backbone is only one step. Gigapops and campus should also be deployed up to the end user.

DNS IPv6 Root Server available

- DNS IPv6 root and gtld (.com, .org, .net) server running on CA*Net3.
  - For the purpose of testing and help deployment.
  - No intent to run a root server in the long term.
  - Many parallel initiatives. Some warnings. Some coordination is being discussed.
  - If you want to try or participate, contact me.

Cisco Configuration

Plan

- Roadmap
- Enabling IPv6
- Interface
- Routing
- Tunnels

Roadmap

- Implementation since 1995.
- RIP and BGP since the beginning.
- Bundled and supported in 12.2(1)T.
  - RIP, BGP, ND,RA, Autoconfiguration, ICMP, 6to4, Standard access lists, IP over typical medias, utilities (ping,...)
  - All hardware except GSR and router blades in switches.
  - Unsupported in current release:
    - DAD, ICMP redirect.

Roadmap (cont.)

- Next major release 12.2(3)T:
  - ISIS, CEF, NAT-PT, Dial, Basic mibs.
- After:
  - OSPF, IPSec, MobileIP, Multicast, QOS, Netflow, Encapsulation on other medias, MIBs+, Hardware support.

Enabling IPv6

- **`ipv6 unicast-routing`**
  - Global context
  - Enables the router to act as an IPv6 router (ipv6 forwarding table, router advertisements, ...)
  - Without it, the router can only be an IPv6 host.

Interface Commands

- Under the interface context, configure the address:
  - **ipv6 address <ipv6addr>[/<prefix-length>] [link-local]**
  - **ipv6 address <ipv6prefix>/<prefix-length> eui-64**
  - **ipv6 unnumbered <interface>**
- Example:
  - **ipv6 address 3ffe:b00:c18:1::1/64**

Interface Commands (cont.)

- As soon as an address is configured on an interface:
  - The link-local address is configured.
  - Router advertisements are sent.

Router Advertisements

- By default, router advertisements are automatically sent, using default parameters:
  - Prefix based on interface prefix
  - Lifetimes set to infinite
- Router advertisements can be adjusted from their default values using the **ipv6 nd** commands on the interface
- **`ipv6 nd prefix-advertisement <routing-prefix>/<length> <valid-lifetime> <preferred-lifetime> [onlink | autoconfig]`**
  - Useful to specify finite lifetimes.
  - Don't forget the optional keywords "onlink autoconfig" for most cases.

Example

```
ipv6 unicast-routing
interface Ethernet0
   ipv6 address 3ffe:b00:c18:1::/64 eui-64
   ipv6 nd prefix-advertisement 3ffe:b00:c18:1::/64
     43200 43200 onlink autoconfig
```

BGP

- Use a new BGP address family for IPv6
  - **address-family ipv6**
- If configuring IPv6 only router, add:
  - **no bgp default ipv4-unicast**
  - **bgp router-id <router-id>**
- Neighbor commands with IPv6 addresses
- Prefix filtering
  - **ipv6 prefix-list**
- Route maps for attribute modification available.

BGP Example

```
ipv6 unicast-routing
interface Ethernet0
  ipv6 address 3FFE:B00:C18:2:1::F/64
router bgp 65001
  no bgp default ipv4-unicast
  bgp router-id 192.168.1.1
  neighbor 3FFE:B00:C18:2:1::1 remote-as 65002
  address-family ipv6
    neighbor 3FFE:B00:C18:2:1::1 activate
    neighbor 3FFE:B00:C18:2:1::1 prefix-list bgp65002in in
    neighbor 3FFE:B00:C18:2:1::1 prefix-list bgp65002out out
    exit-address-family
```

Tunnels

- Similar to IPv4 tunnels, but:
  - Use tunnel mode ipv6ip.
- Example:

```
interface Tunnel0
   no ip address
   ipv6 address 3ffe:b00:c18:1::3/64
   tunnel source 192.168.99.1
   tunnel destination 192.168.30.1
   tunnel mode ipv6ip
```

6to4

- Uses the tunnel interface
  - Don't have to specify the IPv4 address of the destination
  - Specify 6to4 in the tunnel mode
- Example:

```
interface Tunnel0
 no ip address
 ipv6 unnumbered Ethernet0
 tunnel source 192.168.99.1
 tunnel mode ipv6ip 6to4
```

Troubleshooting

- **Ping, traceroute**
- **telnet**
- **debug ipv6 packet**
- **show ipv6 neighbors**
  - Neighbors cache
- **debug ipv6 icmp**
- **debug ipv6 nd**
  - Neighbor discovery

Summary

- Cisco IOS supports main features of IPv6, and will continue to add.
- Interface, Routing, Tunnels and troubleshooting commands are similar to their IPv4 counterpart, but with new features.

References

- Cisco roadmap: http://www.cisco.com/ipv6
- Cisco documentation: http://www.cisco.com

Acknowledgements

- This work is partly funded by Canarie (http://www.canarie.ca).

- We are using the Kame Stack (http://www.kame.net).

Annexe (if time allows...)

IPv6 and DNS

Viagénie

Plan

- Name to IPv6 Address
- IPv6 Address to Name
- Renumbering with A6
- Binary labels and DNAME
- Transport
- Root Servers
- Transition

Name To IPv6 Address

- New records:
  - AAAA
    - Defines the mapping from the domain name to the IPv6 address.
    - Equivalent to the IPv4 A record.
    - Supported in Bind since 4.9.5
  - A6
    - Same function as the AAAA record
    - Helps renumbering by cutting the request in sub-requests.
    - Will eventually replace AAAA records
    - Supported in Bind9

IPv6 Address To Name

- PTR record
  - Defines the mapping from an IPv6 address to a name
  - Same record as for IPv4
  - New top level for the IPv6 space is used: was ip6.int, moving to ip6.arpa)
  - May use the new records: binary labels and DNAME.

Configuration with BIND

- AAAA records

```
$ORIGIN ipv6.viagenie.qc.ca.
www  in  aaaa  3ffe:b00:c18:1:290:27ff:fe17:fc1d
```

- PTR records (ip6.arpa)

```
$ORIGIN 1.0.0.0.8.1.c.0.0.0.b.0.e.f.f.3.ip6.arpa.
d.1.c.f.7.1.e.f.f.f.7.2.0.9.2.0   in   ptr   www.ipv6.viagenie.qc.ca.
```

Renumbering with A6

- When renumbering a site, the 64 rightmost bits do not change, but the leftmost do.
- DNS Name-to-address mapping must be updated to reflect the new leftmost bits.
- A6 divide the Name-to-address query into sub-queries:
  - Give me the rightmost 64 bits and give me the other entry that points to the leftmost 64 bits.
  - The entry for leftmost 64 bits will be the only place in the DNS tree to change when renumbering happens. No need to change all occurrences in all possible zone files.

Example

1. Client sends A6 request to server1 for www.abc.com
2. Server1 returns part of the answer:
   - N1.abc.com:0290:27FF:FE17:FC1D
   - It sends 3 informations:
     - Part of the address.
     - Length of this part.
     - Name for the next part.
3. Client sends A6 request to server2 for N1.abc.com.
4. Server2 returns the other part of the answer:
   - 3FFE:0B00:0C18:0001:...
5. Client assembles the answers together:
   - 3FFE:0B00:0C18:0001:0290:27FF:FE17:FC1D

Example With Renumbering

- Client sends A6 request to server1 for www.abc.com
- Server1 returns part of the answer:
  - N1.abc.com:0290:27FF:FE17:FC1D
- Client sends A6 request to server2 for N1.abc.com.
- Server2 returns the other part of the answer, which is the new prefix:
  - 2001:2:6301:0001:...
- Client assembles the answers together:
  - 2001:2:6301:0001:0290:27FF:FE17:FC1D

Consequences

- Example use only 2 sub-queries. Could be more (one for the provider prefix, one for the site prefix, one for the host part).
- For 1 typical request, then many sub-requests to be done=> more trafic, more latency before full answer.
- Handling caching:
  – Needs to decrease the TTL for right timing of the change.

A6 BIND syntax

- Syntax:
  - a.b.c A6 <NN> <address-suffix> <name>
  - Where:
    - a.b.c is the domain name
    - NN is the prefix length
      - = 128 - length of address-suffix
    - Address-suffix is the specified part of the address being handled by this sub-record
    - Name is the next sub-record to get additional information.
      - Absent if NN = 0.
  - A6 0 == AAAA.

Binary Labels

- Inverse delegation for PTR records:
  - Each character in the tree is a nibble (4 bits).
  - No easy way to delegate on a non-nibble boundary.
- Binary labels
  - Help the writing of PTR records.
  - Address boundaries can be at the bit level.
- Example:
  - d.1.c.f.7.1.e.f.f.f.7.2.0.9.2.0.1.0.0.0.8.1.c.0.0.0.b.0.e.f.f.3.ip6.arpa.
  - can be written as:
  - \[x3FFE0b000c18000127fffe17fc1d/128].IP6.ARPA.

DNAME

- DNAME records
  - Similar to the CNAME record
  - Example:
    - d.e.f DNAME x.yz
    - Lookup of a.b.c.d.e.f gives a.b.c.x.yz
  - Enables easier renumbering for PTR records.

Transport

- IPv6 data queries over IPv4 and IPv6
  - Bind4-8 answers to queries over IPv4 transport only.
  - Bind 9 can answer to queries over IPv6 transport.

Root servers

- Not configured for IPv6 native queries now.
- But AAAA records can be used on the current root servers.
- Issues concerning:
  - the maximum size of data in the hints send back to clients if all root servers are all IPv4 and IPv6: too much space in the return packet.
  - Cache pollution if IPv6 root servers change over time.

Transition

- Intent to not have two record types for IPv6 addresses: AAAA and A6.
- Have to get rid of one.
- Current discussions are:
  - Use A6 in zone files
  - Have DNS servers receiving AAAA queries from current deployed clients to synthesize the AAAA response and do the recursive A6 sub-queries.

Summary

- New records:
  - AAAA
  - A6
  - Binary Labels
  - DNAME
- PTR is same but with a different root ip6.arpa.
- New records help renumbering and delegation.
- Root servers on IPv6 have some issues.

References

- *DNS Extensions to support IP version 6*, Thompson and Huitema, IETF RFC1886, December 1995, http://www.normos.org/ietf/rfc/rfc1886.txt
- *DNS Extensions to Support IPv6 Address Aggregation and Renumbering*, Crawford and Huitema, IETF RFC2874, July 2000, http://www.normos.org/ietf/rfc/rfc2874.txt