Introduction to ICMPv6, Neighbor Discovery, Autoconfiguration and IPv6 DNS

Jean-Francois Tremblay
Sales Engineer
Jean-Francois.Tremblay@hexago.com

Overview

- ICMPv6
- Path MTU Discovery
- Multicast Listener Discovery
- Neighbor Discovery
  - Neighbor cache
  - DAD
- Autoconfiguration
  - DHCPv6
  - Router Advertisements
  - Privacy Extensions
  - Renumbering
  - SEND
- IPv6 DNS
  - IPv6 records and IPv6 transport
  - Current state of deployment

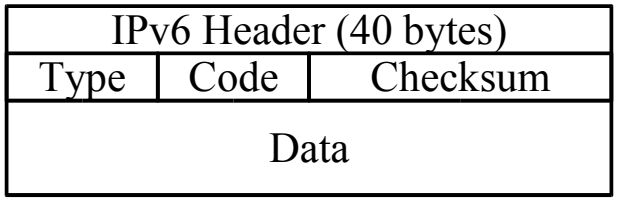
ICMPv6

- ICMPv6 is more than ping!
- In IPv4, ICMP is limited to error detection, test and router redirection.
- IPv6 usage:
  - Error handling, test, Path MTU Discovery
  - Neighbor Discovery
    - Discovery of hosts and routers on the link
    - Neighbor Unreachability Detection (NUD)
    - Physical layer binding (similar to ARP)
    - Duplicate Address Detection (DAD)
  - Management of multicast groups
  - Stateless automatic configuration
  - Router redirection

- Blocking ICMPv6 can lead to serious network malfunction.

ICMPv6 Messages

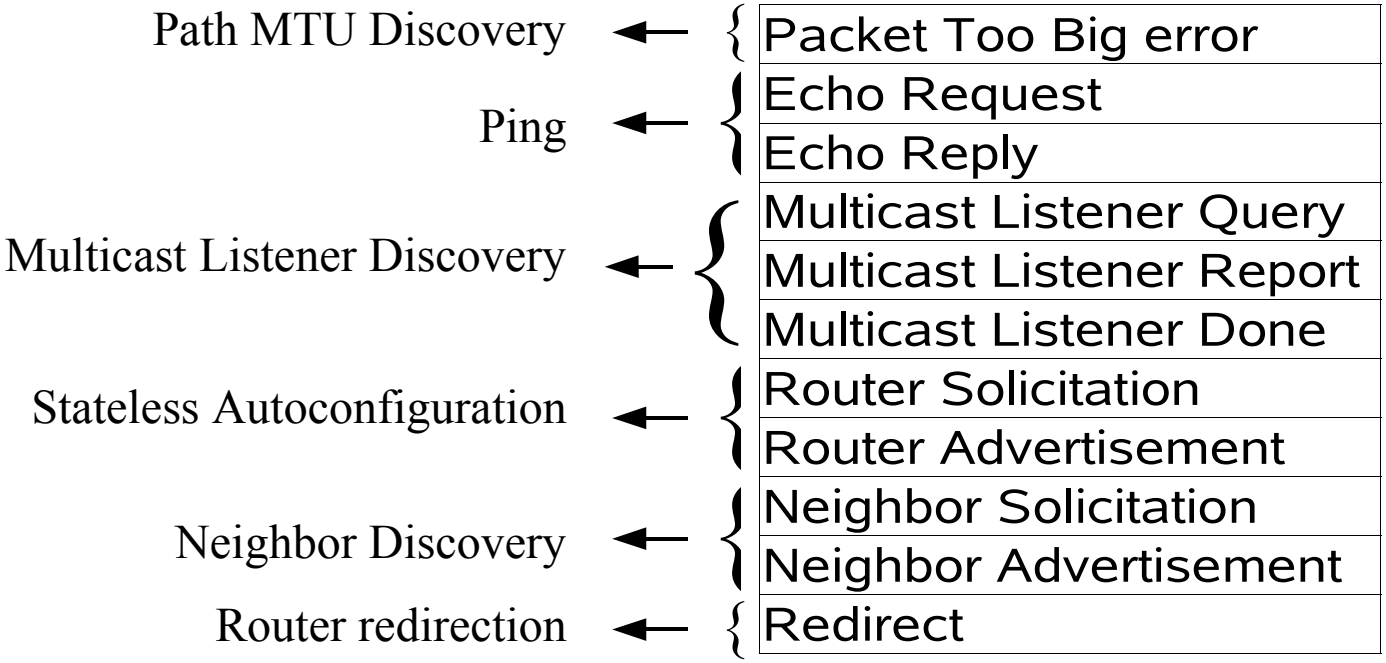
Generic ICMPv6 packet format:



ICMPv6 principal types and codes :

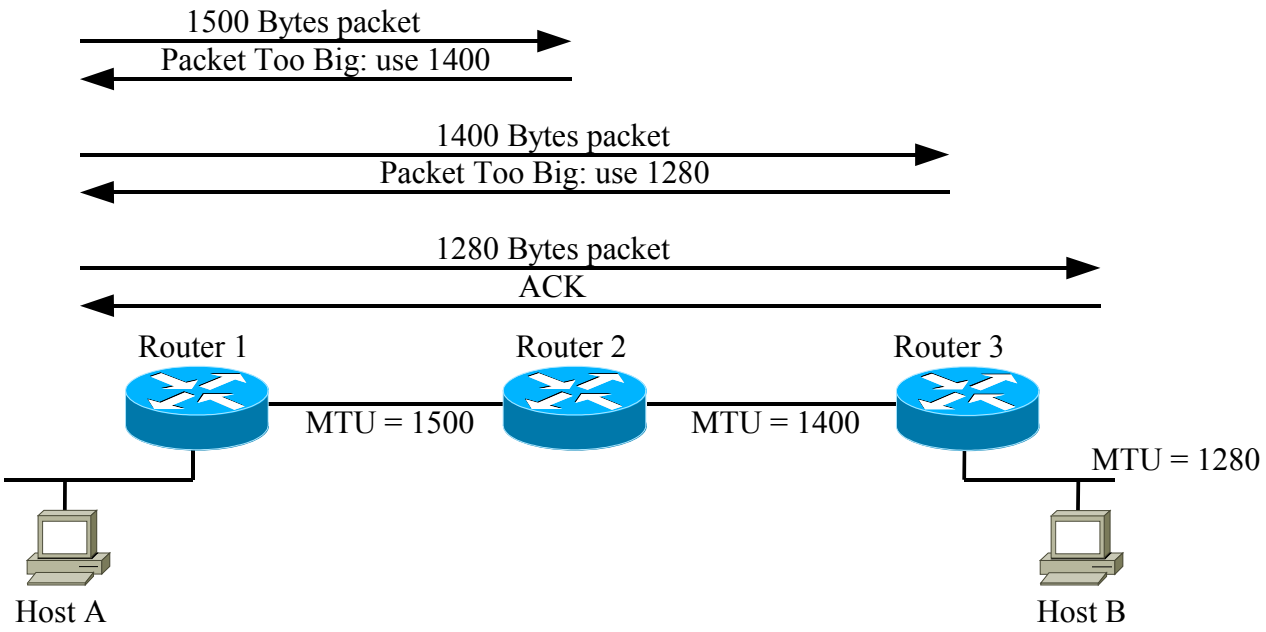
	Type	Meaning	Code	Code explanation
Errors	1	Destination Unreachable	0	No route to destination
			1	Administratively prohibited
			2	Out of scope
			3	Address unreachable
			4	Port unreachable
	2	Packet Too Big		
	3	Time Exceeded	0	Hop limit exceeded
			1	Fragment reassembly time exceeded
	4	Parameter Problem	0	Erroneous header field
			1	Unrecognized next header type
			2	Unrecognized IPv6 option
Informational	128	Echo Request		
	129	Echo Reply		
	130	Multicast Listener Query		
	131	Multicast Listener Report		
	132	Multicast Listener Done		
Neighbor Discovery	133	Router Solicitation		
	134	Router Advertisement		
	135	Neighbor Solicitation		
	136	Neighbor Advertisement		
	137	Redirect		

Packets Usage



Path MTU Discovery

- Fragmentation is not supported in IPv6.
- The maximum MTU of a link is discovered by processing Packet Too Big errors.



Host A will not talk to host B if ICMPv6 is blocked!

Multicast Listener Discovery

- Defines groups on a link, similar to IGMPv2 for IPv4.
- Three messages
  - Multicast Listener Query:
    - Sent periodically from a router to know which hosts are part of the group.
    - Sent from a host to know if there's a listener at a specific address.
  - Multicast Listener Report:
    - Sent to join the group or answer a previous query
  - Multicast Listener Done:
    - Sent to leave the group
- Most common multicast groups (see RFC 2375):
  - FF02::1 – All nodes on link
  - FF02::2 – All routers on link
  - FF02::5 – All OSPF routers
  - FF02::9 – All RIP routers

FF02::1:2 – All DHCP agents
FF05::1:3 – All DHCP routers
FF05::1:4 – All DHCP relays

MLD Example

- Finding all nodes on a link

```
[tremblay@localhost tremblay]$ ping6 -Ieth0 ff02::1
PING ff02::1(ff02::1) from fe80::200:86ff:fe3c:9c12 eth0: 56 data bytes
64 bytes from ::1: icmp_seq=0 ttl=64 time=0.078 ms
64 bytes from fe80::290:27ff:fe54:f2b0: icmp_seq=0 ttl=64 time=0.820 ms (DUP!)
64 bytes from fe80::2e0:18ff:fefa:b5c: icmp_seq=0 ttl=64 time=1.85 ms (DUP!)
64 bytes from fe80::2e0:18ff:fe06:965d: icmp_seq=0 ttl=64 time=1.97 ms (DUP!)
64 bytes from fe80::2e0:18ff:fefa:c51: icmp_seq=0 ttl=64 time=2.28 ms (DUP!)
64 bytes from fe80::202:b3ff:fea8:12df: icmp_seq=0 ttl=64 time=2.48 ms (DUP!)
64 bytes from fe80::200:86ff:fe4c:463b: icmp_seq=0 ttl=64 time=3.09 ms (DUP!)
64 bytes from ::1: icmp_seq=1 ttl=64 time=0.074 ms
```

- Finding routers on the link

```
[tremblay@localhost tremblay]$ ping6 -Ieth0 ff02::2
PING ff02::2(ff02::2) from fe80::200:86ff:fe3c:9c12 eth0: 56 data bytes
64 bytes from fe80::290:27ff:fe54:f2b0: icmp_seq=0 ttl=64 time=0.539 ms
64 bytes from fe80::290:27ff:fe54:f2b0: icmp_seq=1 ttl=64 time=0.597 ms
64 bytes from fe80::290:27ff:fe54:f2b0: icmp_seq=2 ttl=64 time=0.495 ms
--- ff02::2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 0.495/0.543/0.597/0.049 ms, pipe 2
```

Neighbor Discovery

- Purposes:
  - Address resolution: Similar to ARP, matches the IPv6 addresses to physical addresses.
  - Neighbor Unavailability Detection (NUD), prevents packets to be sent to unavailable hosts or routers
  - Duplicate Address Detection (DAD) is used to verify the uniqueness of an address on the link.
- Neighbor Solicitation
  - Sent to a node to request a link-layer address.
- Neighbor Advertisement
  - Sent periodically or as a response to a Neighbor Solicitation. Contains the link-layer address of the node.
- Each host maintains a neighbor cache table built from neighbor discovery information

Neighbor Discovery (cont.)

- Example of neighbor cache on Windows

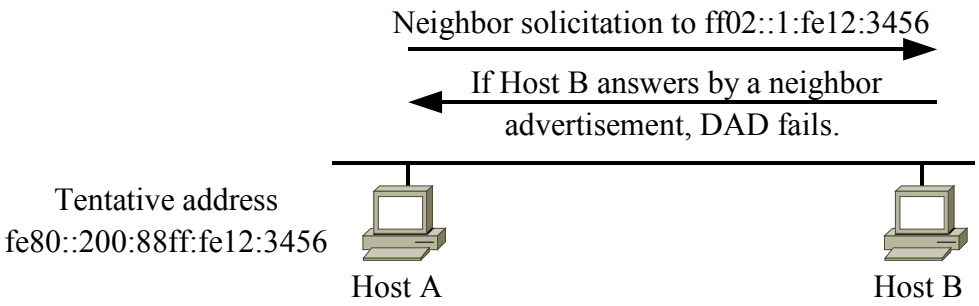
```
C:\>netsh show interface ipv6 show neighbors
Interface 2: Local Area Connection
Internet Address                                Physical Address   Type
----------------------------------------------  -----------------  -----------
2001:5c0::186                                   00-09-6b-49-85-82  Stale
fe80::290:27ff:fe54:f2b0                        00-09-6b-49-65-a7  Stale (router)
fe80::200:86ff:fe3c:9c12                        00-00-86-3c-9c-12  Permanent
2001:5c0::200:86ff:fe3c:9c12                    00-00-86-3c-9c-12  Permanent
2001:5c0::6c90:a713:d180:3d96                   00-00-86-3c-9c-12  Permanent
fe80::209:6bff:fe49:8582                        00-09-6b-49-85-82  Stale
2001:5c0::1                                     00-09-6b-49-65-a7  Stale
2001:5c0::b8b8:5276:d43a:822e                   00-00-86-3c-9c-12  Permanent
```

- Example of neighbor cache on Linux

```
[root@localhost]$ ip -6 neigh
[tremblay@localhost tremblay]$ ip -6 neigh
fe80::20c:29ff:fec3:4a1b dev eth0 lladdr 00:0c:29:c3:4a:1b nud delay
fe80::260:97ff:fec3:3a24 dev eth0 lladdr 00:60:97:c3:3a:24 nud reachable
fe80::20d:60ff:fe8d:498b dev eth0 lladdr 00:0d:60:8d:49:8b nud reachable
fe80::209:6bff:fe49:8582 dev eth0 lladdr 00:09:6b:49:85:82 nud delay
fe80::203:47ff:fee3:5035 dev eth0 lladdr 00:03:47:e3:50:35 router nud reachable
fe80::200:86ff:fe4c:463b dev eth0 lladdr 00:00:86:4c:46:3b nud reachable
```

Duplicate Address Detection

- To make sure an address is not used twice on the link.
- When creating a new unicast IPv6 address, a node must verify if this address is unique using DAD.
- In this example, Host A must verify if the tentative address fe80::200:88ff:fe12:3456 is unique.
  - Host A sends neighbor solicitation packet at the solicited-node multicast address FF02::1:fe12:3456, with the unspecified address (all 0's) as a source.
  - If Host B answers to the all-node multicast address FF02::1, then DAD fails.
  - The address is unique if no response is received.

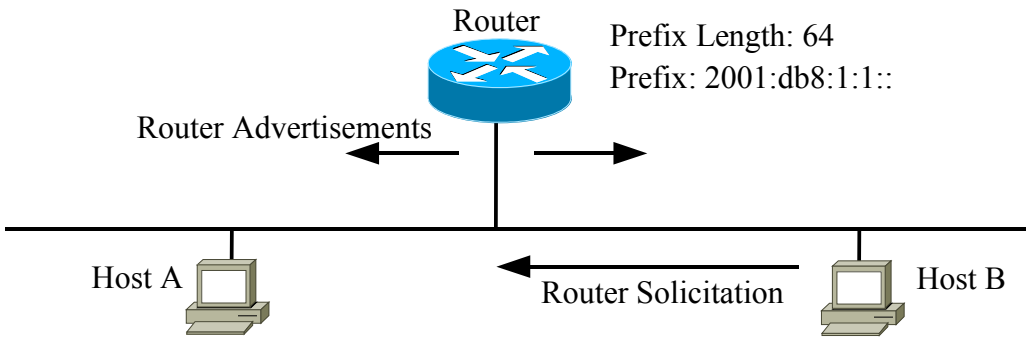


Autoconfiguration

Stateless	Stateful
Uses Neighbhor Discovery	Uses DHCPv6
Lightweight, included in IPv6	Requires DHCPv6 server and clients
Minimal setup and administration time	Requires some administration of the DHCP server
Gradual renumbering	Triggered renumbering
Addresses automatically generated	More controled addresses
Fixed addresses manually configured	Fixed addresses may be assigned
DNS parameters not (yet) provided	Provides DNS parameters

Stateless Autoconfiguration

- The goal: to have nodes on the link configured with a global IPv6 address and a default router.
- Router Solicitation
  - Sent by a node joining a new link to the all-router multicast address FF02::2.
- Router Advertisements :
  - Sent periodically on the link to all-node multicast group or in response to Router Solicitation.
  - Contains the prefix to use for configuration and its length (64).



Prefix Information

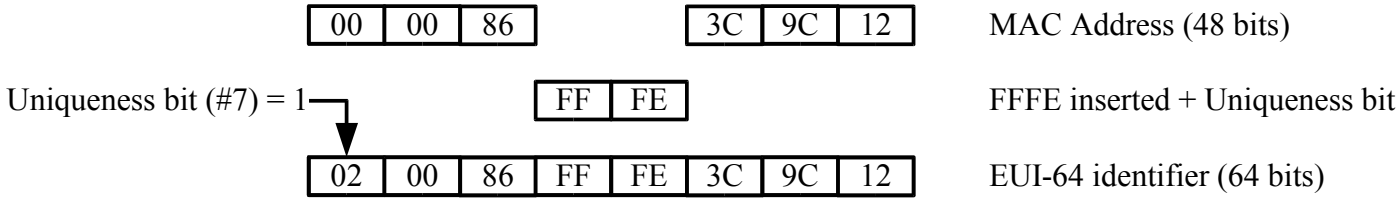
Router Advertisment

IPv6 Header		
134	0	Checksum
Hops	MOH	Lifetime
Reachable Time		
Retransmit Interval		
Options (MTU, Prefix, Link-layer address)		

Prefix information (option)

Type	Length	Pr. Len	LAR
Valid Lifetime			
Preferred Lifetime			
Reserved			
Prefix			

- Prefix information Option:
  - A flag: Prefix can be used for autoconfiguration.
  - Valid Lifetime: how long the prefix is valid
  - Preferred Lifetime: how long the address can be used before being deprecated.
- If the autoconfiguration flag is on, hosts use the prefix information to configure an address.
- By default, nodes configure a global unicast address by concatenating an advertised prefix (first 64 bits) with an interface identifier in the EUI-64 format:



Full Address: 2001:0db8:0001:0001:0200:86FF:FE3c:9C12

Privacy Extensions

- The interface identifier of an IPv6 address being unique, it can be used to track users.
  (for example on web sites, in a way similar to cookies)
- Privacy extensions aims at preventing tracking by randomizing the interface identifier.
- A MD5 hash of the EUI concatenated with a random number is used as the interface identifier.
- This is why some operating systems have by default two global IPv6 addresses configured on each interface:

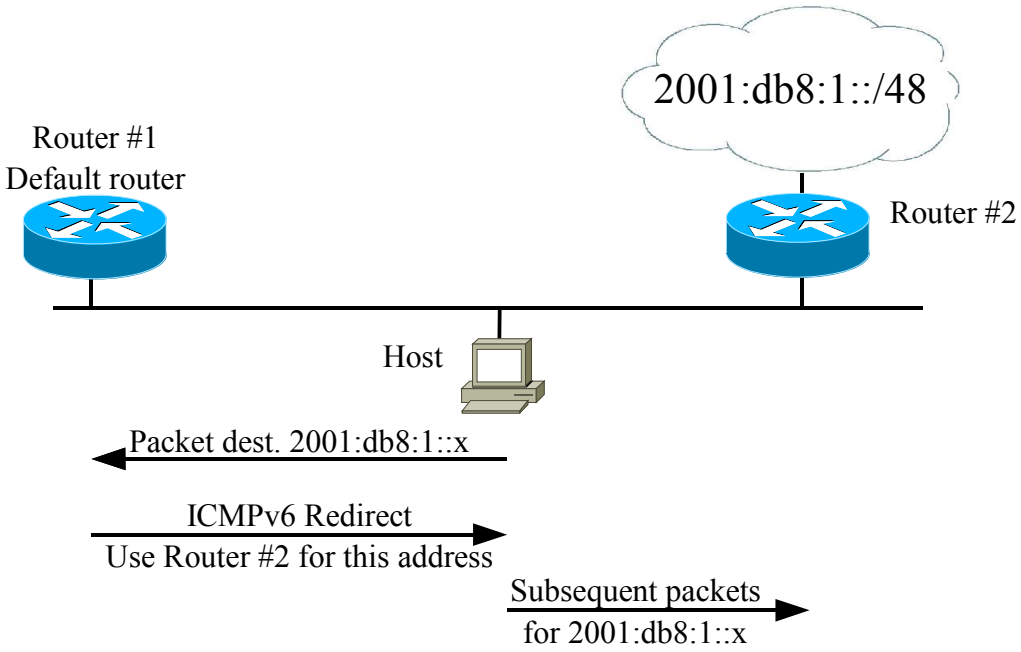
```
c:\> ipconfig
Ethernet adapter Local Area Connection:
...
        IP Address. . . . . . . . . . . . : 2001:5c0::d9de:360d:bdf6:a081
        IP Address. . . . . . . . . . . . : 2001:5c0::200:86ff:fe3c:9c12
        IP Address. . . . . . . . . . . . : fe80::200:86ff:fe3c:9c12%2
```

DHCPv6

- Similar to DHCPv4, but hosts check RAs first.
- Can be used concurrently with stateless configuration (M bit in Router Advertisements).
- DHCP solicit messages are sent on the all-DHCP-agents multicast address: FF02::1:2.
- Can be used for renumbering using Reconfigure-Init message.
- DHCPv6 provides DNS information, stateless autoconfiguration does not, for now.

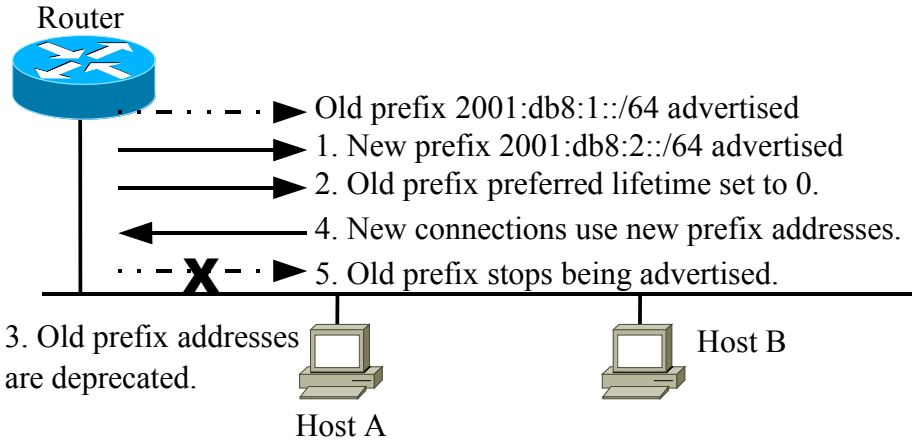
Redirection

- ICMPv6 Redirect packet is sent by the default router when a better first hop is available
  (i.e. when a packet is returned by the same interface it arrived on).
- Similar to ICMPv4 Redirect.



Renumbering

- Stateless configuration was designed to facilitate renumbering.
- General steps:
  1) A new prefix is advertised on the link.
  2) Current prefix preferred lifetime is set to 0, valid lifetime reduced (for example to 1 hour).
  3) Addresses with old prefix become deprecated.
  4) New connections use addresses with the new prefix.
  5) Old prefix is no longer advertised.
- Some details in draft-ietf-v6ops-renumbering-procedure-03.txt.



Secure Neighbor Discovery (SEND)

- Neighbor Discovery is vital for a network to work properly. However, it is not secure.
- Neighbor or router spoofing are possible attacks, along with rogue advertisers, redirect and unreachability attacks.
- IPSec unpractical for securing ND.
- Improvements on standard neighbor discovery:
  - CGAs makes it possible to prove the ownership of a specific address.
  - Signed ND messages protect message integrity and authenticate the sender.
  - Nonce prevent replay attacks.
  - Trust anchors may certify the authority of routers.

DNS

IPv6 Records

- IPv6 DNS uses the same base principles and servers than IPv4.
- Some records were added to support IPv6 addresses:
  - AAAA records:
    - Entire address in a single record.
    - Supported in resolvers since 1996.
    - Similar to A records, but with IPv6 address.
    - Supported in BIND since 4.9.5.
  - A6 and DNAME records were designed to facilitate renumbering but are no longer used (mostly for performance reasons).

- Example:

```
> dig www.hexago.com
...
sonata.hexago.com.      23h14m6s IN A    209.71.226.2
sonata.hexago.com.      2d6h52m55s IN AAAA  2001:5c0:0:1::2
```

IPv6 First

- IPv6-aware applications usually try IPv6 addresses first.
- Also, if no IPv6 router is present, the destination is assumed to be on-link (RFC2461).
- Issue described in draft-ietf-v6ops-v6onbydefault-03.txt.
- This might cause delays if a dual-stack node does not have IPv6 connectivity.
- Also, adding AAAA records for servers that do not implement IPv6 should be avoided.
- Example:
  - A web browser may take a large amount of time timing out on the IPv6 address before trying IPv4.

Reverse Lookups

- PTR records are the same than with IPv4
- The original inverse-mapping domain for IPv6 was ip6.int.
  - Now replaced by ip6.arpa.
  - However several resolver implementation still look for ip6.int, especially for 6Bone addresses.
- Example:

```
> dig 0.c.5.0.1.0.0.2.ip6.arpa
...
;; AUTHORITY SECTION:
0.c.5.0.1.0.0.2.ip6.arpa.  56m11s IN SOA  sonata.hexago.com.
...
> dig 0.c.5.0.1.0.0.2.ip6.arpa ns +norecurse
...
;; ANSWER SECTION:
0.c.5.0.1.0.0.2.ip6.arpa.  2d23h59m46s IN NS  sonata.hexago.com.

;; ADDITIONAL SECTION:
sonata.hexago.com.      20h39m59s IN A  209.71.226.2
sonata.hexago.com.      2d4h18m48s IN AAAA  2001:5c0:0:1::2
```

DNS IPv6 Transport

- Records can be served either over IPv4 or IPv6 transport (or both).
- Deployment issues:
  - Fragmentation: packets are limited to 512 bytes. Limits the number of AAAA records in a response.
  - IPv6 operational guidelines suggests that DNS servers should be either IPv4 or dual-stack.
- Current state of support:
  - Gnu LibC 2.2 and above have IPv6 capability (for resolvers).
  - BSD, Linux, OS X (10.2 and above) resolvers have IPv6 transport. Windows does not.
  - IPv6 transport supported in BIND 8.4.0 and 9. (Note: 9.3.0 does not listen on v6 by default).

State Of IPv6 DNS Deployment

- May 25, ICANN recommended to proceed with AAAA records of TLDs that request it.
- On 20 July 2004, the IPv6 AAAA records for the Japan (.jp) and Korea (.kr) country code Top Level Domain (ccTLD) were added.
- Record for France ccTLD to be added shortly.
- October 19th, AAAA records added to some .com and .net top level domain servers (a and b.gtld-servers.net).
- DNS servers with IPv6 transport are still rare.
- AAAA records are more and more common.

State DNS Deployment (cont.)

```
> dig com. ns
...
;; ADDITIONAL SECTION:
a.gtld-servers.net.     1d23h51m28s IN A  192.5.6.30
a.gtld-servers.net.     1d23h51m28s IN AAAA  2001:503:a83e::2:30
b.gtld-servers.net.     1d23h51m35s IN A  192.33.14.30
b.gtld-servers.net.     1d23h51m35s IN AAAA  2001:503:231d::2:30
...
> dig a.dns.jp AAAA
...
;; ANSWER SECTION:
a.dns.jp.               23h59m41s IN AAAA  2001:dc4::1
...
> dig fr. ns
...
;; ADDITIONAL SECTION:
b.nic.fr.               14h23m11s IN AAAA  2001:660:3005:1::1:2
```

RFCs

- RFC2461: Neighbor Discovery
- RFC2710: Multicast Listener Discovery
- RFC2462: stateless autoconfiguration
- RFC3041: privacy extensions.
- Secure Neighbor Discovery (SEND):
  - ND Thrust Models and Threats: RFC3756
  - SEND: draft-ietf-send-ndopt-06.txt (in RFC queue)
  - Cryptographically Generated Addresses: draft-ietf-send-cga-06 (also in RFC queue).
- RFCs related to NDS: 1886, 2672, 2673, 2874, 3363, 3364.
- RFC3901: IPv6 Transport Operational Guidelines.
- RFC3484: Default Address Selection.

The End