IPv6 Tutorial

- Florent Parent
  Florent.Parent@viagenie.qc.ca

- Régis Desmeules
  Regis.Desmeules@viagenie.qc.ca

http://www.viagenie.qc.ca

13 march 2000

Plan

- *Overview of IPv6*
- DNS configuration
- Routing protocols
- Transition strategies
- Router configurations
- Host installation and configuration
- How to connect to the IPv6
- IPv6 deployment on the Internet
- IPv6 industry support and trends

Why IPv6 ?

Problems with IPv4

- IPv4 has been designed early in the 70s
- Many « add-ons» to the protocol :
  - Mobileip
  - QoS
  - Security (IPsec)
  - Others
- Using one « add-ons » -> easy
- Using two at the same time -> difficult
- Using three or more -> acrobatic !!!!

Why IPv6 ?

Problems with IPv4

- During the 80s, addresses delegation without optimisation and without aggregation

Possible solution : IP renumbering and unused address space redistribution

Consequences :

- Large routing table on the backbone
- Unthinkable for some sites

Why IPv6 ?

IPv4 address shortage (current situation)

Fact #1 : Few consequence in North America « Internet heaven »!

Fact #2 : Major problem for every other countries around the world

- China requested addresses to connect 60 000 schools and got one class B
- Several countries in Europe, Africa and Asia are using one class C for a whole country

Why IPv6 ?

IPv4 address shortage (current situation)

- Some ISP in these countries are providing private addresses to their clients (Suedish ISP using NAT)
- Internet users move from PPP connectivity to xDSL/cable modem (ratio users by IP address is changing from 10:1 to 1:1)
- ISP are delegating only few address space to their corporate client s
- Temporary solution --> NAT (but unfortunatly permanent)

Why IPv6 ?

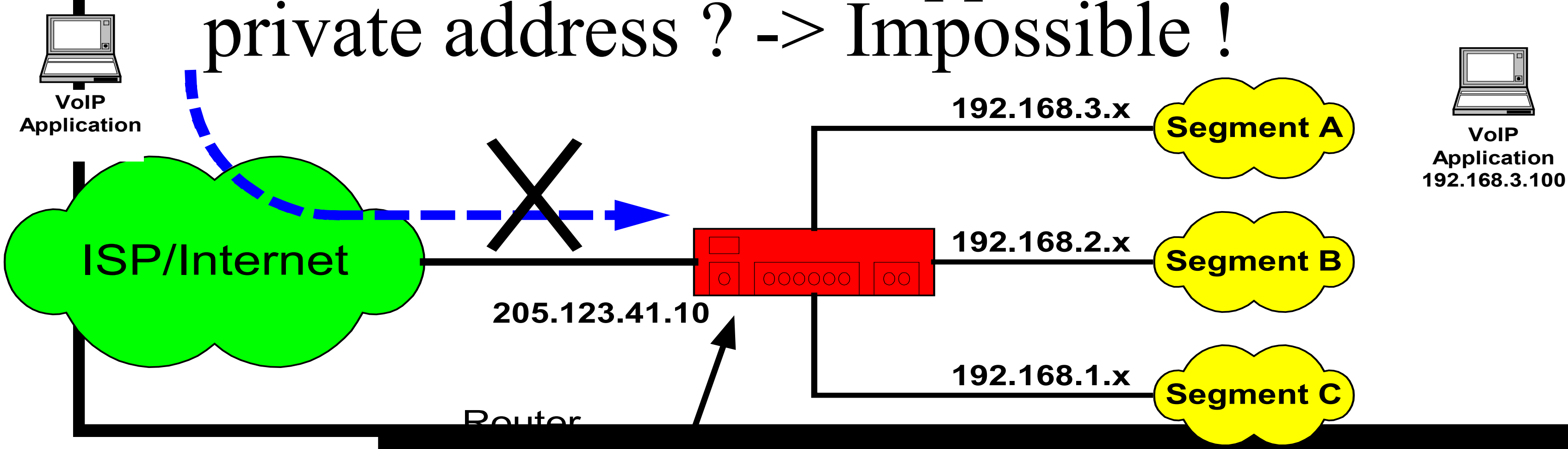
IPv4 address shortage in the future

- Internet growth in some regions :
  - Asia (2.5 billions people)
  - Eastern Europe (250 millions)
  - Africa (800 millions)
  - South and Central America (500 millions)
- Growth of the applications that need IP addresses globally scoped, unique and routable (VoIP, videoconferencing, games)

Why IPv6 ?

NAT « hinders » Internet applications deployment

- Unidirectionnal concept (from Intranets to Internet)
- How to reach a VoIP application with a private address ? -> Impossible !

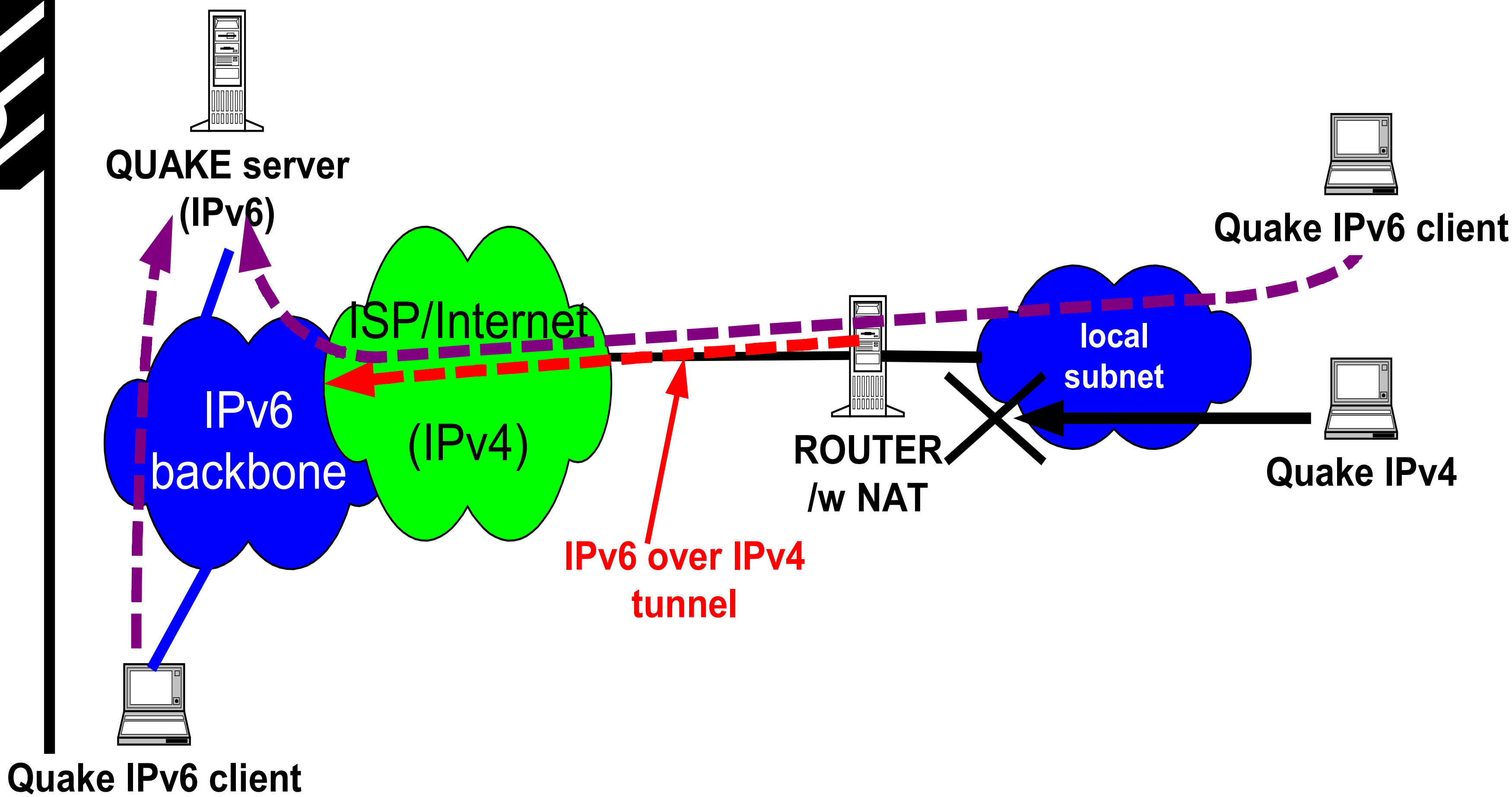


Why IPv6 ?

NAT « hinders » Internet applications deployment

- Comunication, security and game applications need bidirectionnel support
  - VoIP (RTP/RTCP)
  - Videoconferencing (RTP/RTCP)
  - IPsec
  - Network game (Quake multiplayer)
- RFC 2775 about *Internet Transparency* by Brian Carpenter

Home gaming IPv6 setup



Why IPv6 ?

NAT « hinders » Internet applications deployment

- Several protocols don 't pass throught NAT
  - IPsec -> NAT changes address in the packet header -> lost of integrity
  - Kerboros -> NAT changes address in the packet header -> K needs the source address
  - RTP/RTCP -> use UDP with dynamic ports assignation -> NAT is not able to support this translation during a session (except proxy)
  - Multicast is not easy to set-up !!!

Why IPv6 ?

Communications technologies need permanent addresses to get connected to the Internet

- Cellulars (500 millions)
- Standard phones (900 millions)
- Radio/TV (++ hundred millions)
- Industrials devices (billions of IP addresses)
- Any electronics device (walkman to download MP3 files, bulgar alarm to send e-mail to the police station …)

Why IPv6 ?

CONCLUSION :

The true question is not :

« *Do we need and do we believe in IPv6 ?* »

Not, the right one is :

« *Are we interested in a network that allows any IP electronic devices to communicate transparently to each other regarless its location on THE global net ?* »

- Viagénie

IPv6 Features

- Larger Address Space
- Aggregation-based address hierarchy
  - Efficient backbone routing
- Efficient and Extensible IP datagram
  - No fragmentation by routers
  - 64 bits field alignement
  - Simpler basic header
- Autoconfiguration
- Security
- IP Renumbering part of the protocol

History

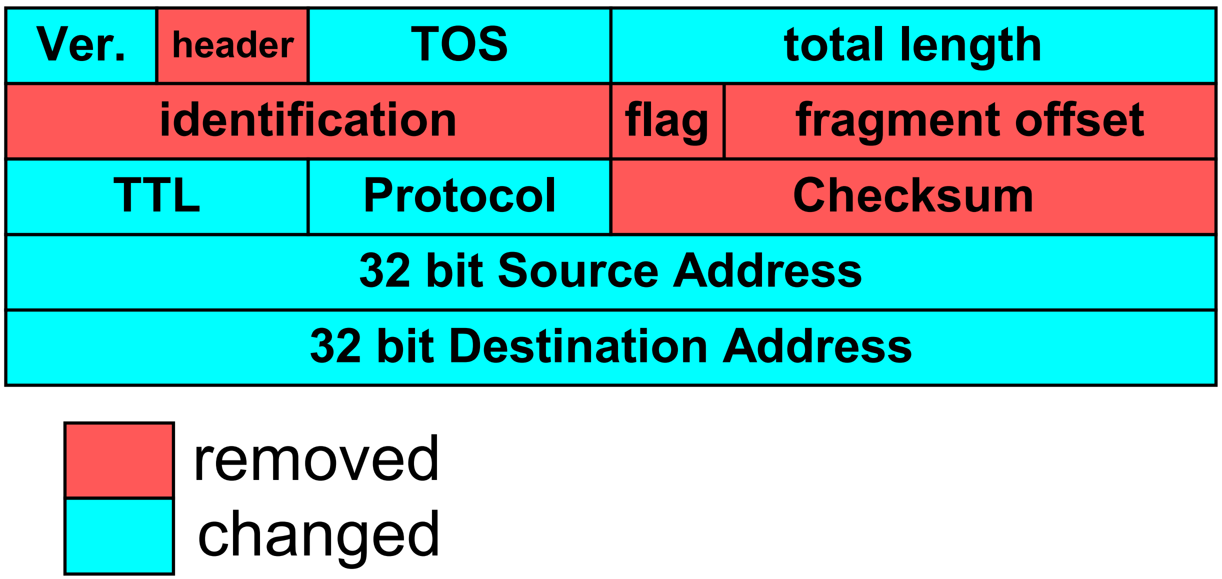
- TUBA (1992)
  - TCP and UDP over Bigger Addresses
  - Uses ISO CLNP (Connection-Less Network Protocol)
  - Dropped
- SIPP (1993)
  - Simple IP Plus
  - Merge of Sip and Pip
  - 64 bits addresses
- IPng adopted SIPP in 1994
  - Changed address size to 128 bits
  - Changed to IPv6

Design criterias for IPv6

- Number of addresses
- Efficiency in routers low and very high bandwidth (100G/bytes++)
- Security
- Mobility
- Autoconfig
- Seamless transition
  - Don’t require a day X for switching to IPv6
  - No need to change hardware

Basic specifications

- IPv4 packet description (20 bytes + options)



Basic specifications

- RFC2460
- IPv6 packet description (40 bytes)

<table>
<tr><td>Ver.</td><td colspan="2">TrafficClass</td><td colspan="3">Flow Label</td></tr>
<tr><td colspan="3">Payload Length</td><td colspan="2">Next Header</td><td>Hop Limit</td></tr>
<tr><td colspan="6">128 bit
Source Address</td></tr>
<tr><td colspan="6">128 bit
Destination Address</td></tr>
</table>

Basic specifications

- 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)
  - Experimental
  - Used by a source node to label sequences of packets
- Payload Length
  - ~= Total length in IPv4

Basic specifications

- Next Header (8 bits)
  - Used for extension headers
  - ~= Protocol field in IPv4
  - Most not processed by routers in the path
  - 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
  - Fragment (44)
    - used by source node (routers don’t fragment anymore !)

Basic specifications

- Next Header (8 bits) cont.
  - Destination options (60)
    - used to carry optional information that need to be examined only by a packet's destination node(s)
  - Authentication (IPsec)
  - ESP (IPsec)
- Hop Limit ~= TTL in IPv4
- MTU must be at least 1280 bytes (1500+ recommended). Nodes should use Path MTU discovery.
- UDP checksum required

IPv6 addresses

- 128 bits = 3,40 E 38 addresses
- Imagine Bill Gates' fortune is 85 billions $ (8.5 E 10)
  - Take 1 trillion Bill Gateses
  - Convert their fortune to pennies
  - Assign 1 E 12 addresses to each pennies
    - takes 8.5 E 36 addresses
  - You've just assigned 2.5% of the entire IPv6 address space
- http://www.cnn.com/TECH/computing/9909/21/ip.crunch.idg/index.html

IPv6 addressing

RFC2373 IP Version 6 Addressing Architecture

Reserved	0000 0000	1/256
Reserved for NSAP Allocation	0000 001	1/128
Reserved for IPX Allocation	0000 010	1/128
Aggregatable Global Unicast Addresses	001	1/8
Link-Local Unicast Addresses	1111 1110 10	1/1024
Site-Local Unicast Addresses	1111 1110 11	1/1024
Multicast Addresses	1111 1111	1/256

Total of about 15 % of address space reserved, but not necessarily assigned or allocated

IPv6 address representation

- Format is `x:x:x:x:x:x:x:x`
  - x is a 16 bit hexadecimal field
  - `FEDC:BA98:7654:3210:FEDC:BA98:7654:3210`
- Leading zeros in a field are optional
- `::` can be used to represent multiple groups of 16 bits of zero
  - `::` can only be used once in an address
  - `FF01:0:0:0:0:0:0:101 = FF01::101`
  - `0:0:0:0:0:0:0:1 = ::1`
  - `0:0:0:0:0:0:0:0 = ::`

IPv6 address representation

- RFC2732: Preferred Format for Literal IPv6 Addresses in URL

http://[1080::8:800:200C:417A]:80/index.html

IPv6 addressing

- Unicast address
  - FE80::/10 Link-Local Unicast Address
    - scope limited to local network
    - automatically configured on all nodes using interface identifiers
    - FE80::<interface id>
    - used for neighbor discovery and router discovery.
    - can also be used as a non-globally-routed IPv6 local network

IPv6 addressing

- Unicast address
  - FEC0::/10 Site-Local Unicast Address
    - confined to local site or organization
    - configured using interface identifier and a pre-defined 16 bits subnet ID
    - FEC0::<subnet id>:<interface id>
    - what is a site??? (few drafts: draft-haberman-ipv6-site-route-00.txt, draft-ietf-ipngwg-site-prefixes-02.txt)

Aggregatable Global Unicast Addresses

- RFC2374 Aggregatable Global Unicast Addresses
- 2000::/3

- TLA: Top Level Aggregator
  - Primary providers (default free)
- NLA: Next Level Aggregator
  - Can have multiple NLA as sub-NLA
- SLA: Site Level Aggregator
  - Your site (16 bits)

Aggregatable Global Unicast Addresses

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

TLA	NLA(s)	SLA	Interface ID
16 bits	32 bits	16 bits	64 bits

- Addresses are allocated from your provider
  - If you change provider, your prefix changes
  - But renumbering (of hosts, routers and sites) has been included in the IPv6 protocol

IPv6 Addressing

- **`::1`**
  - Loopback address (like 127.0.0.1 in IPv4)
- **`::`**
  - Unspecified address
- **`::<IPv4 address>`**
  - IPv4 compatible address
  - Auto-tunnels (IPv6 over IPv4)
- **`::FFFF:<IPv4 address>`**
  - IPv4 mapped address (used by resolver library)
  - IPv6 representation of an IPv4 node
  - 206.123.31.101 is mapped as ::FFFF:206.123.31.101

Multicast address

- RFC2375 IPv6 Multicast Address Assignments
- FF00::/8
  - FF02::1 all nodes on the local network
  - FF02::2 all routers on the local network

FF	0	0

multicast address

flags
0 = permanent
1 = not permanent

scope
1 = node
2 = link
5 = site
8 = organization
E = global

Solicited-Node multicast address

- 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

Anycast address

- Address assigned to more than one interface and/or node
- Packet sent to anycast address is routed to "closest" interface

subnet prefix	11111111111	ID
n bits	121-n bits	7 bits anycast ID

Example: **3FFE:B00:C18:1:FDFF:FFFF:FFFF:FFFE**

RFC2526: Reserved IPv6 Subnet Anycast Addresses

Required Node Addresses

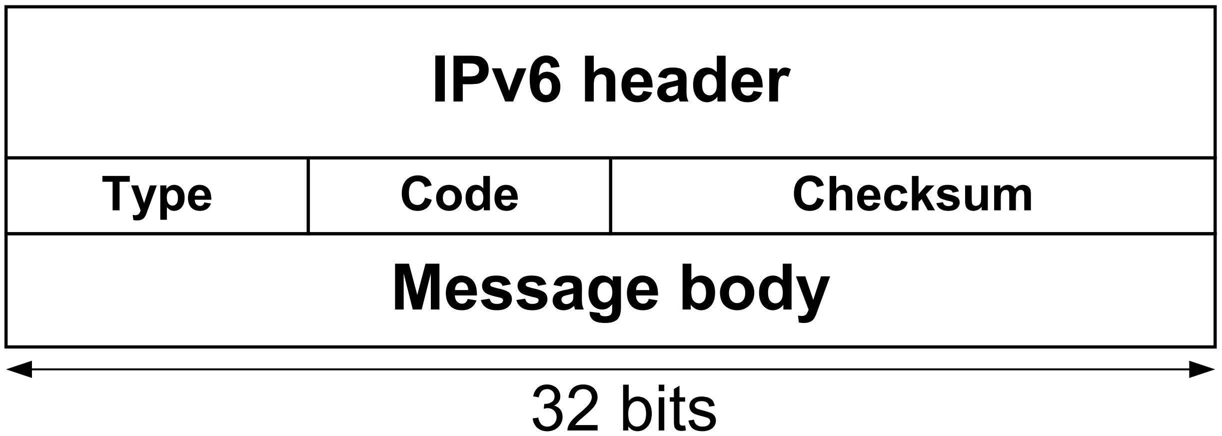
- Link-Local Address for each interface
- Assigned Unicast Addresses
- Loopback Address
- All-Nodes Multicast Addresses
- Solicited-Node Multicast Address for each of its assigned unicast and anycast addresses
- Multicast Addresses of all other groups to which the host belongs

Required Router Addresses

- All the required node addresses
- The Subnet-Router anycast addresses for the interfaces it is configured to act as a router on
- All other Anycast addresses which the router has been configured with
- All-Routers Multicast Addresses

ICMPv6

- RFC2463
- Protocol ICMPv6 (IPv6 Next Header 58)



ICMPv6 error messages

- Type 1: Destination Unreachable
  - Code 0: no route to destination
  - Code 1: communication administratively prohibited
  - Code 3: address unreachable
  - Code 4: port unreachable
- Type 2: Packet Too Big
  - Message contains MTU

ICMPv6 error messages

- Type 3: Time Exceeded
  - Code 0: hop limit exceeded
  - Code 1: fragment reassembly time exceeded
- Type 4: Parameter Problem
  - Code 0: erroneous header field
  - Code 1: unrecognized Next Header type
  - Code 2: unrecognized IPv6 option

ICMPv6 informational messages

- Type 128: Echo request
  - Message contains Identifier and Sequence number
- Type 129: Echo reply
  - Message contains Identifier and Sequence number
- ICMP “who are you”
  - draft-ietf-ipngwg-icmp-name-lookups-05.txt
  - Gets FQDN of remote node
  - Defines new ICMPv6 types for query and reply

Neighbor Discovery

- RFC2461
- ~= ARP in IPv4
- Uses ICMPv6 messages
- Used to:
  - Find link-layer address of neighbor
  - Find neighboring routers
  - Actively keep track of neighbor reachability
- Protocol used for host autoconfiguration
- All ND messages must have Hop Limit=255
  - Must originate from same link

Neighbor Discovery messages

- Router Solicitation
  - ICMP type 133
  - Host request routers to send Router Advertisement immediately

Neighbor Discovery messages

- Router Advertisement
  - ICMP type 134
  - Routers advertise periodically
    - max. time between advertisements can be in the range from 4 and 1800 seconds
  - Contains one or more prefixes
  - Prefixes have a lifetime
  - Specifies if stateful or stateless autoconfiguration is to be used
- Plays a key role in site renumbering

Neighbor Discovery messages

- Neighbor Advertisement
  - ICMP type 136
  - Response to a Neighbor Solicitation
- Neighbor Solicitation
  - ICMP type 135
  - Sent by node to determine link-layer address of a neighbor
- Route change, Redirect
  - Router send better hop for a destination
  - ~= ICMP redirect

IPv6 autoconfiguration

- Stateful autoconfiguration
  - Manual IP configuration
  - DHCP configuration (draft-ietf-dhc-dhcpv6-14.txt)

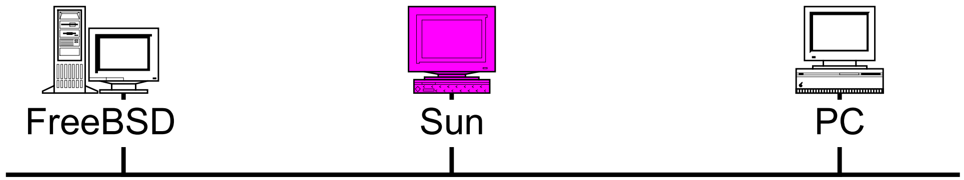
- Stateless Address Autoconfiguration (RFC2462)
  - Applies to hosts only (not to routers)
  - No manual configuration required, but does not specify the DNS servers, the prefix, lifetime and a default route
  - Assumes interface has unique identifier
  - Assumes multicast capable link
  - Uses Duplicate Address Detection

IPv6 autoconfiguration

- Duplicate Address Detection
  - Join all-nodes multicast address (FF02::1)
  - Join solicited-node multicast address of the tentative address
    - FF02:0:0:0:0:1:FF00: ...
  - Send Neighbor Solicitation on solicited-node multicast address
  - If no Neighbor Advertisement is received, address is ok

IPv6 autoconfiguration

Unicast	3FFE:B00:C18:1:280:C8FF:FE68:CF44	3FFE:B00:C18:1:290:27FF:FE17:FC1D
Solicited-Node	FF02::1:FF68:CF44	FF02::1:FF17:FC1D



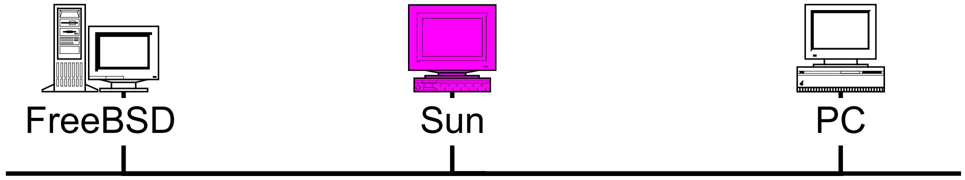
tentative address :

3FFE:B00:C18:1:290:27FF:FE17:FC0F

IPv6 autoconfiguration

Unicast 3FFE:B00:C18:1:280:C8FF:FE68:CF44 3FFE:B00:C18:1:290:27FF:FE17:FC1D

Solicited-Node FF02::1:FF68:CF44 FF02::1:FF17:FC1D



tentative address :

3FFE:B00:C18:1:290:27FF:FE17:FC0F

Join FF02::1 (All Nodes)

Join FF02::1:FF17FC0F

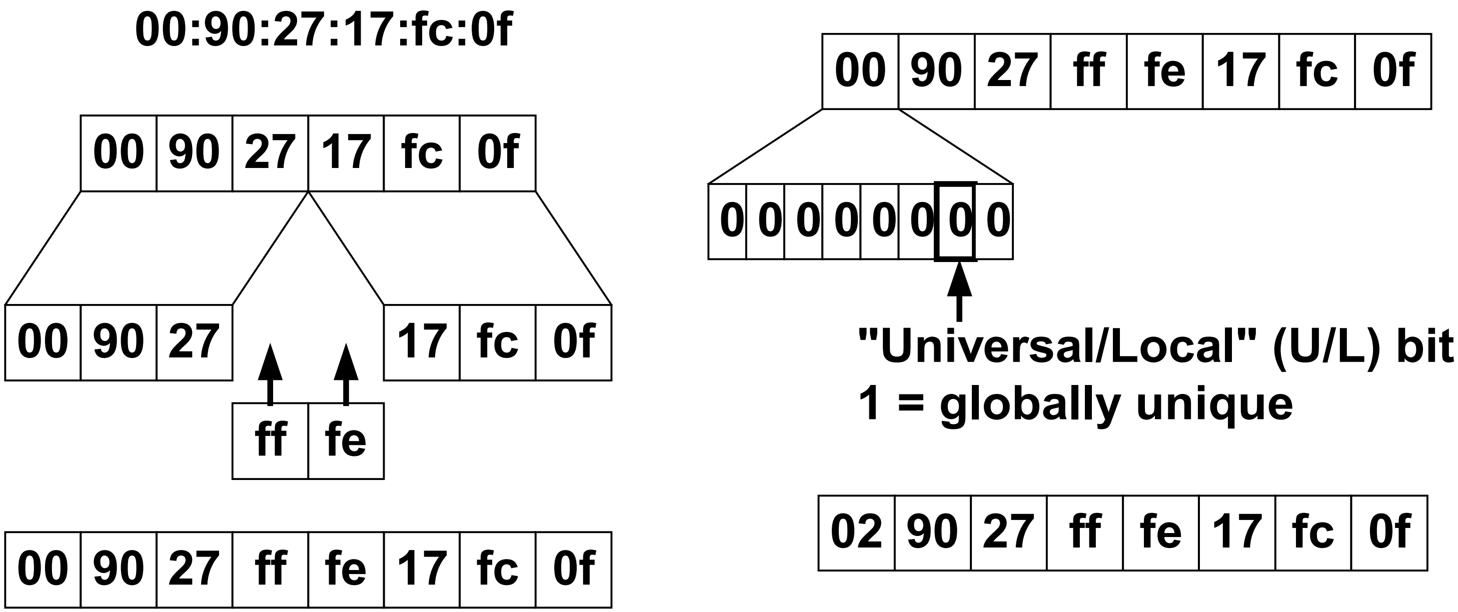
Send Neighbor Solicitation to FF02::1:FF17:FC0F
Listen for response on FF02::1

IPv6 over Ethernet

- RFC2464 (IPv6 over Ethernet)
- Interface Identifier for stateless autoconfiguration
  - EUI-64 interface identifier

IPv6 over Ethernet

- Interface Identifier for stateless autoconfiguration



So lower 64 bits in address are `02:90:27:ff:fe:17:fc:0f`

IPv6 over Ethernet

Frame Format

Destination Ethernet
Source Ethernet
86DD
IPv6 header and payload

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

IPv6 over Ethernet

- IPv6 multicast address mapping over Ethernet

FF02:0000:0000:0000:0000:0001:<u>FF17:FC0F</u>

Solicited multicast address

33-33-<u>FF-17-FC-0F</u>

Ethernet address

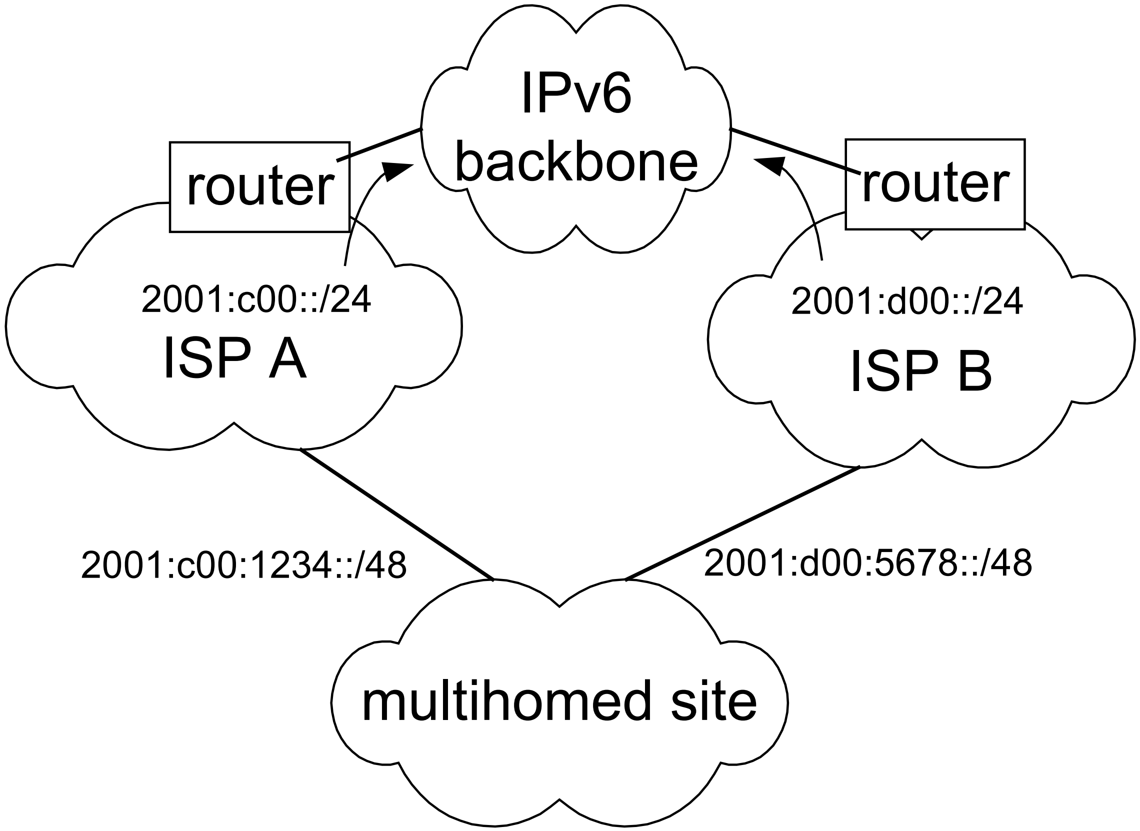
Renumbering

- Site Renumbering: hosts
  - Decrease the lifetime of the prefix in the router advertisement
- Router Renumbering
  - Protocol to renumber routers within a site
  - Defines new ICMPv6 messages
  - draft-ietf-ipngwg-router-renum-09.txt (work in progress)

Multihoming

- The IPv6 address assignment and allocation mechanism is fully hierarchical
  - A site uses its ISP prefix
- A multihomed site will have more than one prefix
- How does the hosts know which source address to use ?

Multihoming



Multihoming

- Work underway at the IETF
- IPng Interim Meeting last sept. in Tokyo
  - focused on multihoming, multi-addressing issues
- 3 drafts are out

Multihoming

- Default Address Selection for IPv6
  - draft-ietf-ipngwg-default-addr-select-00.txt
- IPv6 Multihoming with Route Aggregation
  - draft-ietf-ipngwg-ipv6multihome-with-aggr-00.txt
- Multihomed routing domain issues for IPv6 aggregatable scheme
  - draft-ietf-ipngwg-multi-isp-00.txt

IPsec

- IPsec
  - Provides authentication (AH) and confidentiality (ESP) at the IP level
  - Mandatory in IPv6
  - IPv6 Next Header defines IPsec AH and ESP

Mobility

- Mobility
  – Allows a mobile node to keep the same IP address
  – Integrated in IPv6
  – draft-ietf-mobileip-ipv6-10.txt

Status update of the IETF standards

- IPng working group
  - http://playground.sun.com/pub/ipng/html/
  - Core specs are at draft standard
  - Identify "base set" to move to full standard
  - Start an IPv6 host and router requirements document (similar to RFC1122, 1123)

Status update of the IETF standards

- NGtrans working group
  - http://www.6bone.net/ngtrans/minutes/default.htm
  - Specifying the tools and mechanisms that might be used for transition to IPv6
  - Document transition tools and mechanisms that might apply to various scenarios for a transition to IPv6
  - Development, testing, and deployment of IPv6 on the IPv6 6Bone testbed

Plan

- Overview of IPv6
- ***DNS configuration***
- Routing protocols
- Transition strategies
- Router configurations
- Host installation and configuration
- How to connect to the IPv6
- IPv6 deployment on the Internet
- IPv6 industry support and trends

DNS

- IPv6 AAAA records supported starting in Bind 4.9.5 and 8.1.x
- Newer records such as A6, DNAME and Binary labels are supported starting Bind9

DNS

- Changes to the records to support IPv6 addresses:
  - AAAA (new record)
    - defines the mapping from the domain name to the IPv6 address
    - equivalent to the IPv4 A record
  - A6 (Bind9)
    - same function as the AAAA record
    - helps renumbering
    - maps a domain name to IPv6 address (uses indirection)
    - will eventually replace AAAA records

DNS

- PTR
  - defines the mapping from the IPv6 address to the domain name
  - same record as for IPv4
  - new top level for the IPv6 space is used: IP6.INT
  - uses binary labels and DNAME record

DNS configuration with BIND

- Bind version starting at 4.9.5 and Bind 8.1.x
  - supports AAAA records

- AAAA records

$ORIGIN ipv6.viagenie.qc.ca.

www in aaaa 3ffe:b00:c18:1:290:27ff:fe17:fc1d

- PTR records (ip6.int)

$ORIGIN 1.0.0.0.8.1.c.0.0.0.b.0.e.f.f.3.ip6.int.

d.1.c.f.7.1.e.f.f.f.7.2.0.9.2.0 in ptr www.ipv6.viagenie.qc.ca.

DNS

- RFC1886
- draft-ietf-ipngwg-dns-lookups-06.txt
  - DNS Extensions to Support IP Version 6
  - A6, DNAME, binary labels
  - Supports address aggregation and renumbering
  - New prefix delegation method (DNAME)
- Will eventually replace AAAA records
- Supported in Bind 9

DNS

- A6 records

a.b.c A6 64 ::0290:27FF:FE17:FC1D SLA.b.c

<prefix length>
0 - 128

<address suffix>
IPv6 address

<prefix name>

  - <prefix length> = 128 - length of <address suffix>
  - <prefix name> absent IF <prefix length> = 0

DNS

- Binary labels (Bind9)
  - Address boundary can be specified at binary level
  - Without binary labels, delegation is possible only at nibble boundaries
  - 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.int. can be written as: **`\[x3FFE0b000c18000127fffe17fc1d/128].IP6.INT.`**

DNS configuration

- DNAME records
  - Analogue to the CNAME record
  - d.e.f DNAME x.yz
  - Lookup of a.b.c.d.e.f gives a.b.c.x.yz

DNS configuration

www.ipv6.viagenie.qc.ca

Provider's space		Client's space	
TLA	NLA(s)	SLA	Interface ID

3FFE:0B00:0C18:0001:0290:27FF:FE17:FC1D

A6 record

```
www.ipv6.viagenie.qc.ca  A6 0 3ffe:b00:c18:0290:27FF:FE17:FC1D
```

PTR record

```
\[x3FFE0B000C18029027FFFE17FC1D/128].IP6.INT  PTR  www.ipv6.viagenie.qc.ca
```

Client's DNS

All address information in one record and in client's DNS only: does not support renumbering very well

DNS configuration

www.ipv6.viagenie.qc.ca

Provider's space		Client's space	
TLA	NLA(s)	SLA	Interface ID

3FFE:0B00:0C18:0001:0290:27FF:FE17:FC1D

A6 record

```
www.ipv6.viagenie.qc.ca  A6 64 ::0290:27FF:FE17:FC1D SLA.viagenie.qc.ca

SLA.viagenie.qc.ca  A6 48 0:0:0:1:: viagenie.provider.net
```

Client's DNS

```
viagenie.provider.net  A6 0 3ffe:b00:c18::
```

Provider's DNS

DNS configuration

PTR record

```
\[x3FFE0B000C18/48].IP6.INT           DNAME   IP6.viagenie.qc.ca
```

Provider's DNS

```
\[x0001/16].IP6.viagenie.qc.ca    DNAME   IP6.SLA.viagenie.qc.ca.

\[x029027FFFE17FC1D].SLA.viagenie.qc.ca   PTR   www.ipv6.viagenie.qc.ca
```

Client's DNS

DNS

- IPv6 data queries over IPv4 and IPv6
  - Current Bind distribution answers to IPv4 queries only
  - Extensions to Bind 8.1.2 are available to allow IPv6 DNS queries
  - Bind 9 can answer to IPv6 queries
- Root servers
  - Not configured for IPv6 native queries now
  - But AAAA records can be used on the current root servers

Plan

- Overview of IPv6
- DNS configuration
- *Routing protocols*
- Transition strategies
- Router configurations
- Host installation and configuration
- How to connect to the IPv6
- IPv6 deployment on the Internet
- IPv6 industry support and trends

Routing protocols: RIPng

- RIPng (RFC2080)
  - RIP (Routing information protocol)
  - Interior Gateway Protocol
  - Used in local networks
  - Has the same limitations as RIP-2 (15 hops diameter, fixed metric)
  - Implementations: GateD, Mrtd, Kame route6d, Zebra, Cisco, etc.

Routing protocols: OSPFv6

- RFC2740: OSPF for IPv6
  - OSPF (Open Shortest Path First)
  - Interior Gateway Protocol
  - Used in local networks
  - Changes required from IPv4 (remove IPv4 dependencies)
  - Implementations: Telebit, IBM*, Zebra*, Gated*, MRTd*, Cisco*
    - (* under development)

Routing protocols: BGP4+

- BGP4+ (RFC2283, RFC2545)
  - BGP: Border Gateway Protocol
  - Inter-domain Routing protocol
  - Used between ISPs and large corporations
  - Uses the concept of “autonomous systems”
  - BGP4+ Adds multiprotocol extensions
  - Used to exchange routes between networks on the 6Bone
  - Implementations: GateD, Mrtd, Kame BGPd, Zebra, Cisco, etc.

Plan

- Overview of IPv6
- DNS configuration
- Routing protocols
- ***Transition strategies***
- Router configurations
- Host installation and configuration
- How to connect to the IPv6
- IPv6 deployment on the Internet
- IPv6 industry support and trends

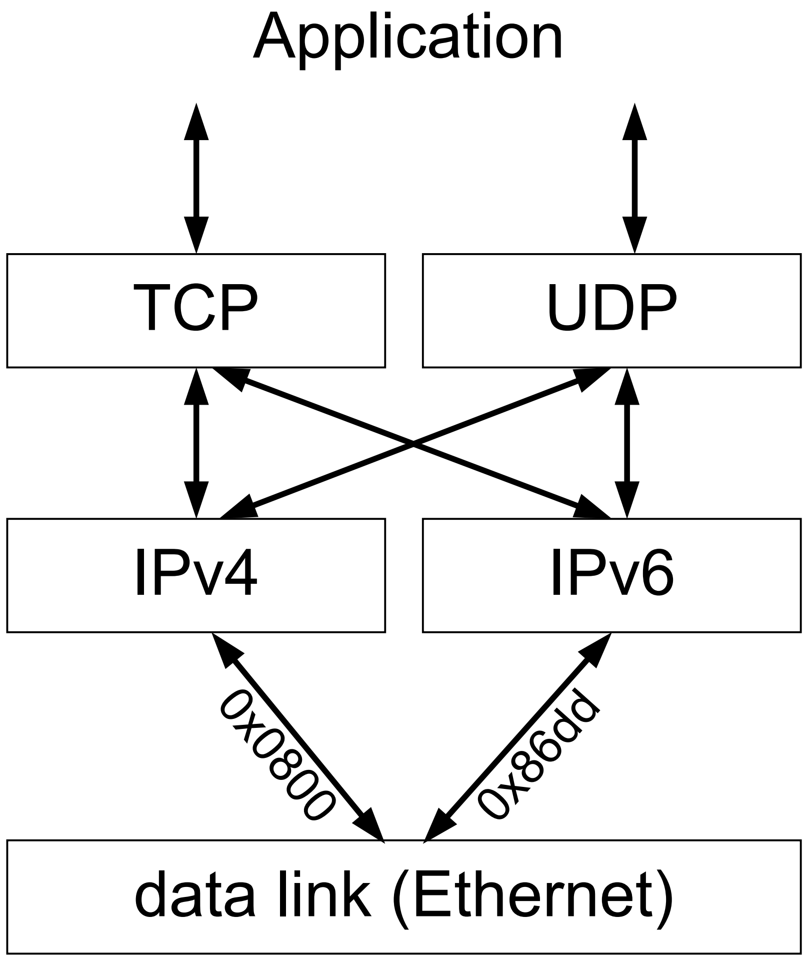
IPv4 and IPv6 transition strategies

- Basic mechanisms:
  - RFC 1933, draft-ietf-ngtrans-mech-04.txt
  - Dual stack host
    - can communicate IPv6 if peer is reachable
  - Configured Tunneling
  - Automatic Tunneling
    - IPv4-compatible IPv6 addresses
  - IPv4 multicast tunneling

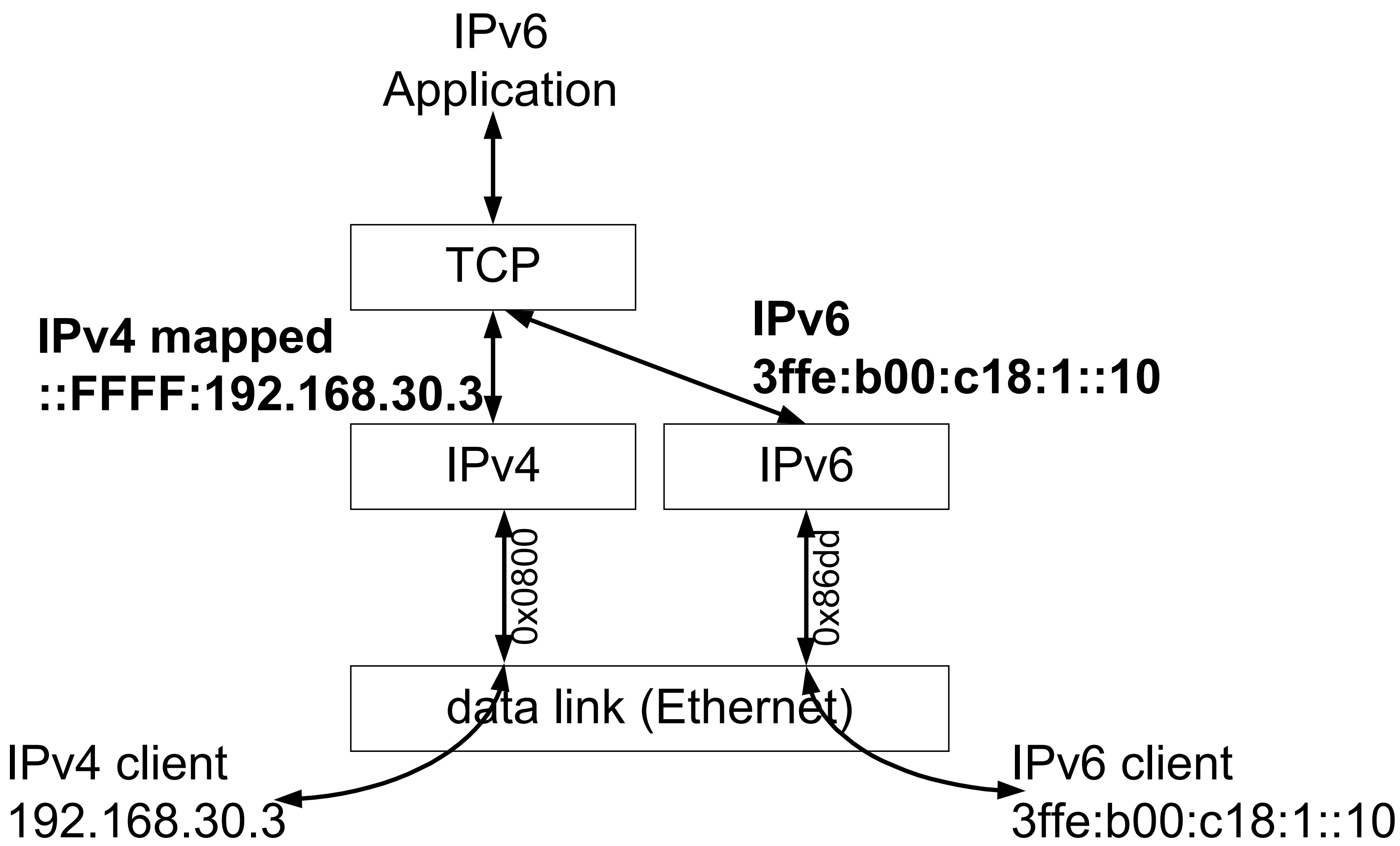
Dual Stack

- Node has both IPv4 and IPv6 stacks and addresses
- DNS resolver
  – returns IPv6, IPv4 or both to application
- IPv6 application can use IPv4 mapped addresses to communicate with IPv4 nodes

Dual stack


Application
TCP
UDP
IPv4
IPv6
0x0800
0x86dd
data link (Ethernet)

Dual stack

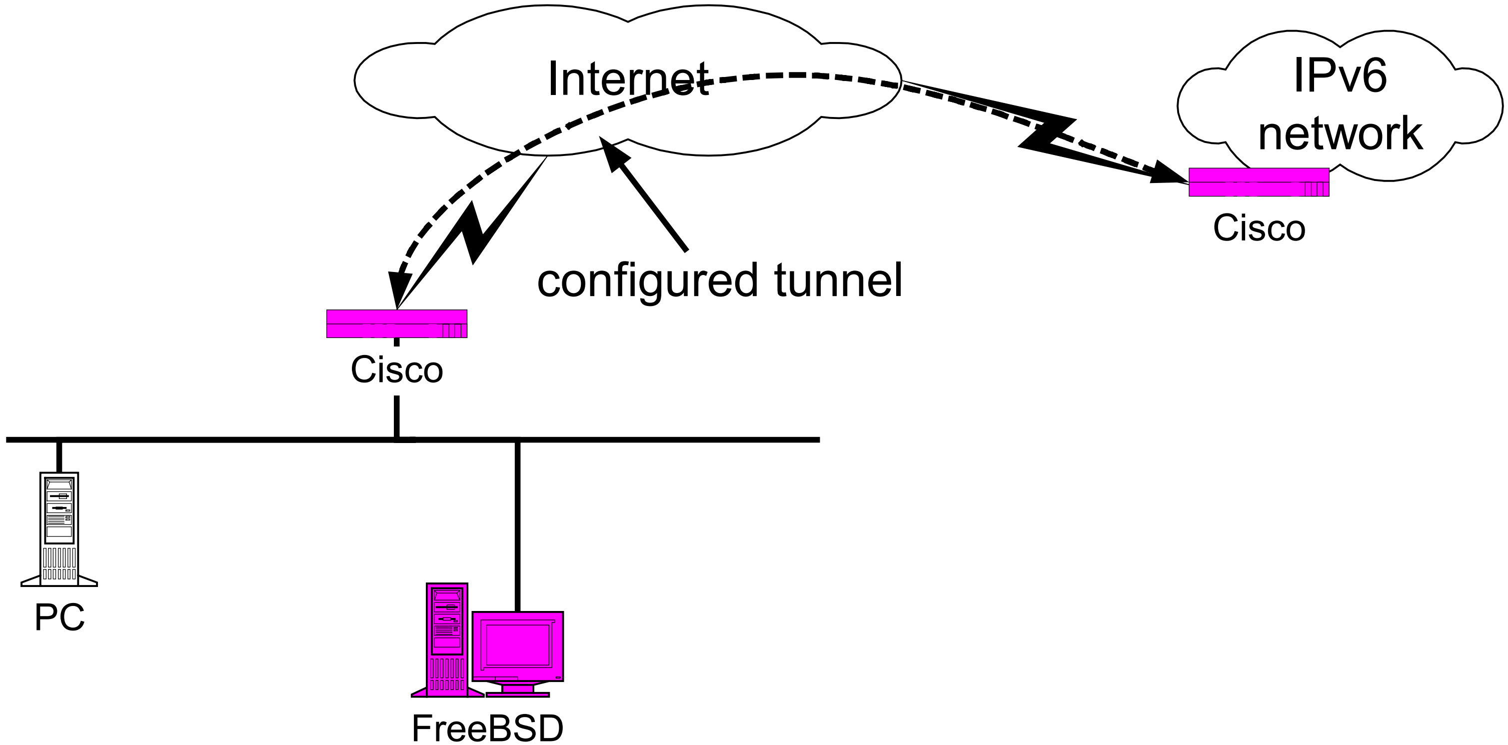

IPv6
Application
TCP
IPv4 mapped
::FFFF:192.168.30.3
IPv6
3ffe:b00:c18:1::10
IPv4
IPv6
0x0800
0x86dd
data link (Ethernet)
IPv4 client
192.168.30.3
IPv6 client
3ffe:b00:c18:1::10

Tunneling

- IPv6 encapsulated in IPv4
- Configured tunneling
  - Routing table chooses which tunnel to take
- Automatic tunneling
- IPv4 multicast tunneling

Configured tunneling

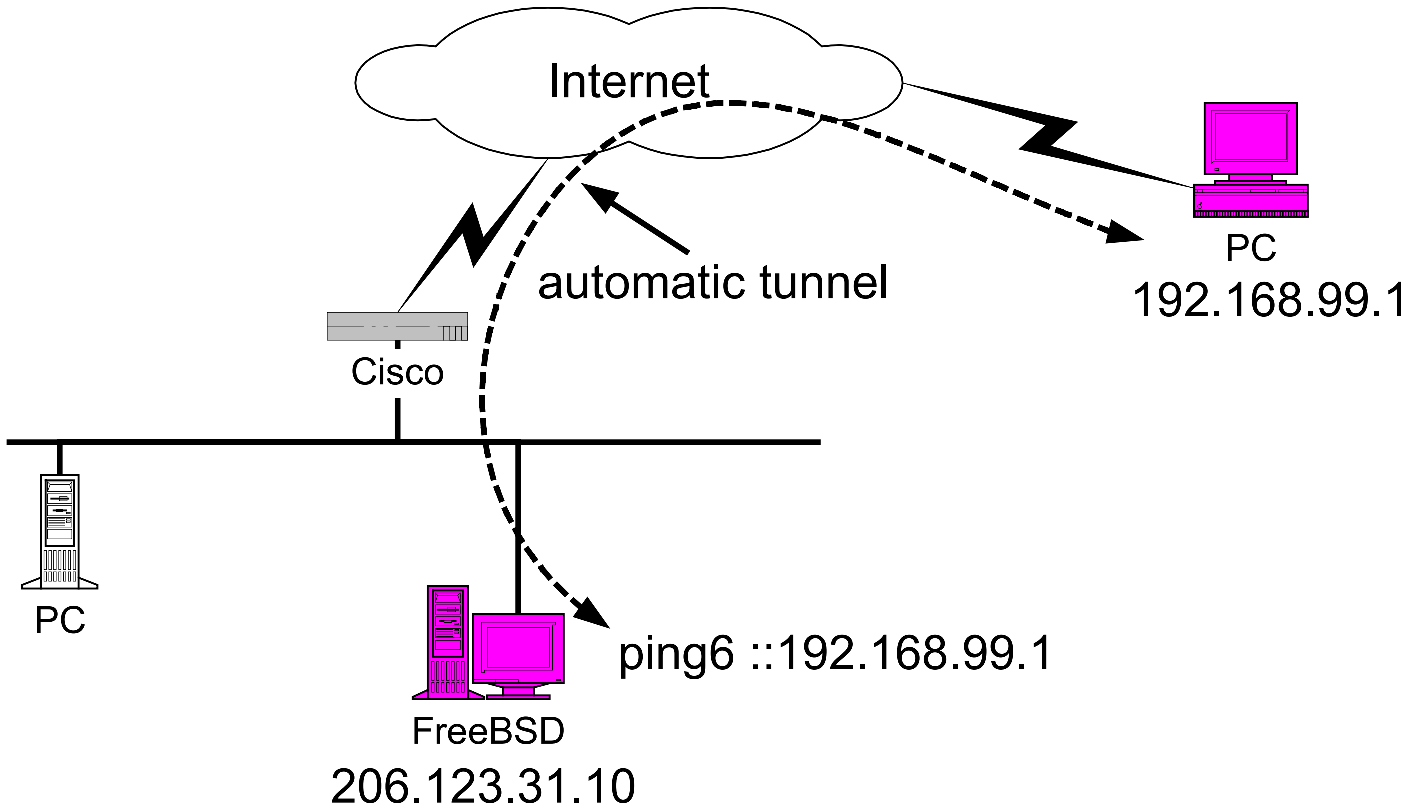
- Tunnel endpoints are explicitly configured



Automatic tunneling

- Node is assigned an IPv4 compatible address
  - ::206.123.31.101
- If destination is an IPv4 compatible address, automatic tunneling is used
  - Routing table redirects ::/96 to automatic tunnel interface

Automatic tunneling



IPv4 multicast tunneling

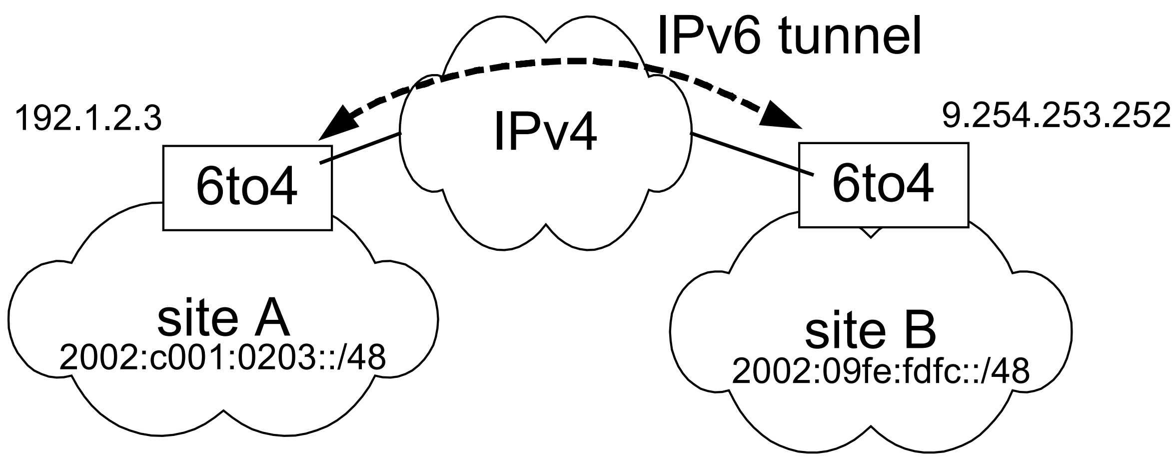
- RFC2529: known as “6 over 4”
- IPv4 tunnel endpoints determined by Neighbor Discovery
- Does not require IPv4 compatible addresses
- IPv4 multicast infrastructure required

6to4

- Interconnection of isolated IPv6 domains in an IPv4 world
- Lets IPv6 sites communicate transparently over the IPv4 Internet backbone
- The egress router of the IPv6 domain creates a tunnel to the other domain
- The IPv4 endpoints of the tunnel are identified in the prefix of the IPv6 domain
- draft-ietf-ngtrans-6to4-03.txt

6to4

- This prefix is made up of a unique 6TO4 TLA plus an NLA that identifies the site by the IPv4 address of the translating egress 6to4 router
- 6to4 TLA is 2002::/16
- If 6to4 router has IPv4 address 192.1.2.3, then 6to4 site prefix is 2002:c001:0203::/48



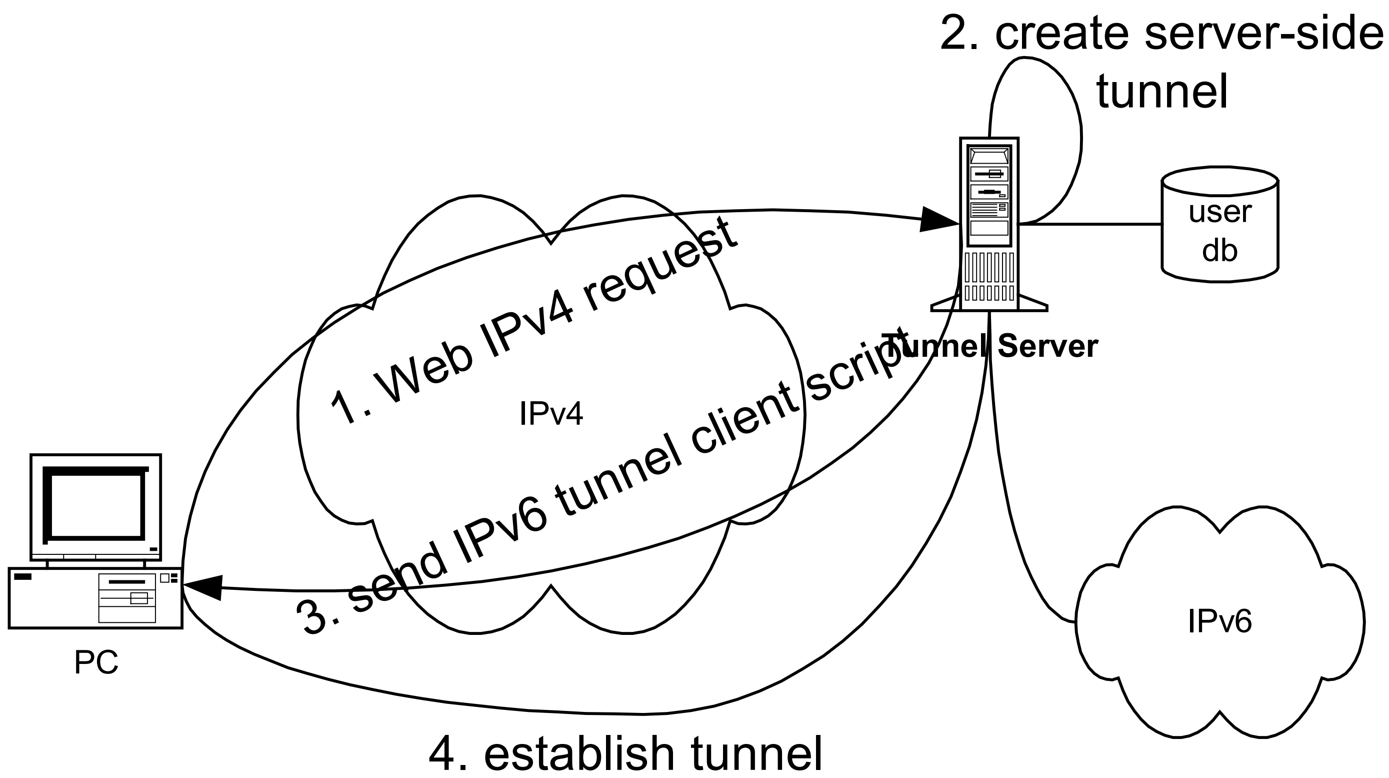
Tunnel Server

- http://www.freenet6.net
- A freenet concept for IPv6
- Plug-and-play IPv6 using the current IPv4 Internet as the transport
- Looks like an IPv6 NAS:
  - Provides IPv6 connectivity on demand
  - Assigns an IPv6 address to the host
  - Connects the host to the IPv6 Internet
- Ideas:
  - Tunnel broker (Alain Durand)
  - Web site for info/implementations/… (Orlando BOF, 1998)

User interface

- Fill-out a Web form
  - Choose your OS
  - Verify your IPv4 address
  - Enter a nickname and your country (for DNS)
- Server creates its tunnel end point
- Client receives a script that should be executed: this script creates the tunnel on the client side
- You are connected

Tunnel server model



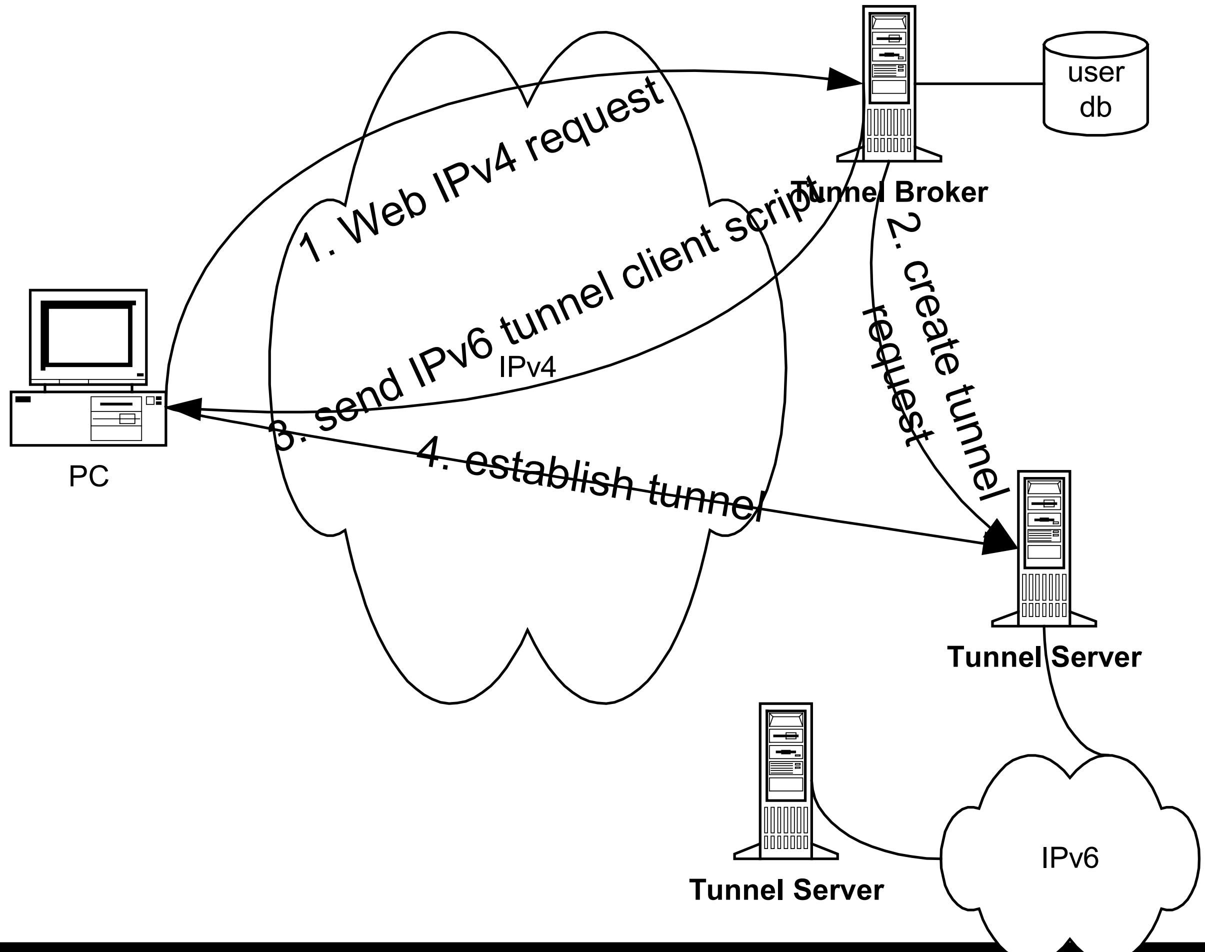
Implementation and future work

- FreeBSD with KAME stack
- Currently supported clients: NT, FreeBSD/Kame, FreeBSD/Inria, Cisco (as a host), Linux, Solaris8
- Very easy to add new clients (if the tunnel creation can be scripted)
- Add support for more host implementations
- Add support for IPv6 routers (net behind tunnel endpoint)
- Make the code available

Tunnel Broker

- draft-ietf-ngtrans-broker-02.txt
- Alain Durand idea
- Implementation by CSELT
  – http//carmen.cselt.it/ipv6/download.html
- Broker
  – User has a username/password
  – Receives the users request by the Web
  – Sends a “create-tunnel” command to one of the tunnel servers
  – Tunnel server creates the tunnel end point
  – Client receives the script to create its tunnel end point
  – User can come back to delete his tunnel by using his username/password to authentify

Tunnel broker model



Other transition mechanisms

- Many other mechanisms being studied in the ngtrans wg
- Other mechanisms take care of IPv4-only clouds speaking with IPv6-only clouds
- Work in progress
- No decision on which to use, forward to standard track
- New drafts to compare the mechanisms, roadmap
  – draft-ietf-ngtrans-introduction-to-ipv6-transition-02.txt

Plan

- Overview of IPv6
- DNS configuration
- Routing protocols
- Transition strategies
- *Router configurations*
- Host installation and configuration
- How to connect to the IPv6
- IPv6 deployment on the Internet
- IPv6 industry support and trends

Router configurations

- Cisco
- 3COM
- FreeBSD/MRTd

Cisco

- IPv6 code based on IOS 11.3(5)T
- http://www.cisco.com/warp/public/732/ipv6/index.html
- IPv6 code based on IOS 12.0T
- http://www.cisco.com/go/ipv6

- Available on many platforms:
  - c1000 c1005 c1600 c2500 c2600 c3620 c3660 c4000 c4500 c5200 c7200 c5rsm* gsr* ...
    (* 12.0T)

Cisco

Global commands

```
ipv6 unicast-routing [table-count <num>]
```

Enables forwarding of IPv6 unicast datagrams

```
ipv6 route <prefix> {<next-hop> | <interface>}
  [<distance>] [table <num>]
```

Configures a static IPv6 prefix route

```
ipv6 route 3FFE:B00:C18:2::0/64 Tunnel1
```

Cisco

Interface commands

`ipv6 enable`

Enables IPv6 in interface

Auto-configures the IPv6 link-local unicast address

`tunnel mode ipv6ip`

Encapsulate IPv6 packets in IPv4. Used for tunnels.

Cisco

Interface commands

```
ipv6 address <ipv6addr>[/<prefix-length>] [link-
  local]
ipv6 address <ipv6prefix>/<prefix-length> eui-64
```

configures IPv6 address on interface

```
ipv6 address 3FFE:B00:C18:3::0/64 eui-64
```

Cisco

Interface commands

nd: Neighbor Discovery configuration

```
ipv6 nd prefix-advertisement
   <routing-prefix>/<length>
   <valid-lifetime>
   <preferred-lifetime>
   [onlink | autoconfig]
```

```
ipv6 nd prefix-advertisement
   3FFE:B00:C18:3::0/64 86400 86400 onlink
   autoconfig
```

Cisco

- ATM interface commands

```
interface atm[physical int #].[sub-interface #]
         point-to-point
```

Creates an ATM sub interface

```
atm pvc [pvc_name] [VPI] [VCI] aal5snap
```

PVC with multiprotocols over ATM support

```
ipv6 address <ipv6addr>[/<prefix-length>]
```

Configures IPv6 address on sub-interface

Cisco

- ATM interface commands

Example

```
interface ATM0.3 point-to-point
 description Native IPv6 over ATM
 atm pvc 111 7 100 aal5snap
 ipv6 enable
 ipv6 address 3FFE:B00:C18:7000::1/64
```

Cisco

- Debug commands

```
debug ipv6 packet
debug ipv6 icmp
debug ipv6 nd
debug ipv6 bgp
```

- traceroute ipv6 <destination>
- ping ipv6 <destination>

Cisco

```
show ipv6 route [connected | local |
  static] | [<prefix> | addr-or-name>]
```

```
#show ipv6 route 3ffe:1c00::3

IPv6 Routing Table - 110 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
Timers: Uptime/Expires

B 3FFE:1C00::0/24 [20/5]
    via FE80::60:2F03:5C08:E, ATM0.2, 18:06:46/never
```

Cisco RIPng

- RIPng

```
ipv6 rip <tag> enable

ipv6 rip <tag> summary-address <prefix>/<length>
```

summarize routing information

```
ipv6 rip <tag> input-filter|output-filter <name>

ipv6 rip <tag> redistribute static
```

Cisco BGP4+

- BGP4+ global commands in 11.3(5)T

```
ipv6 bgp neighbor <address> remote-as <as-num>
```

  - defines a neighbor

```
ipv6 bgp neighbor 3FFE:B00:800:1::1 remote-as
  6509
```

```
ipv6 bgp neighbor <address> route-map <name> in
```

  - filter updates received from this neighbor

```
ipv6 bgp neighbor 3FFE:B00:800:1::1 route-map
  PrefNative in
```

Cisco BGP4+

```
ipv6 bgp neighbor <address> route-map <name> out
```

– filter updates sent to this neighbor

```
ipv6 bgp network <prefix>
```

– advertises prefix as “internal”

```
ipv6 bgp network <prefix> summary
ipv6 bgp redistribute connected
ipv6 bgp redistribute static
ipv6 bgp redistribute rip <tag>
```

Cisco BGP4+

`show ipv6 bgp`

- Displays IPv6 BGP table, can also specify specific route on last argument

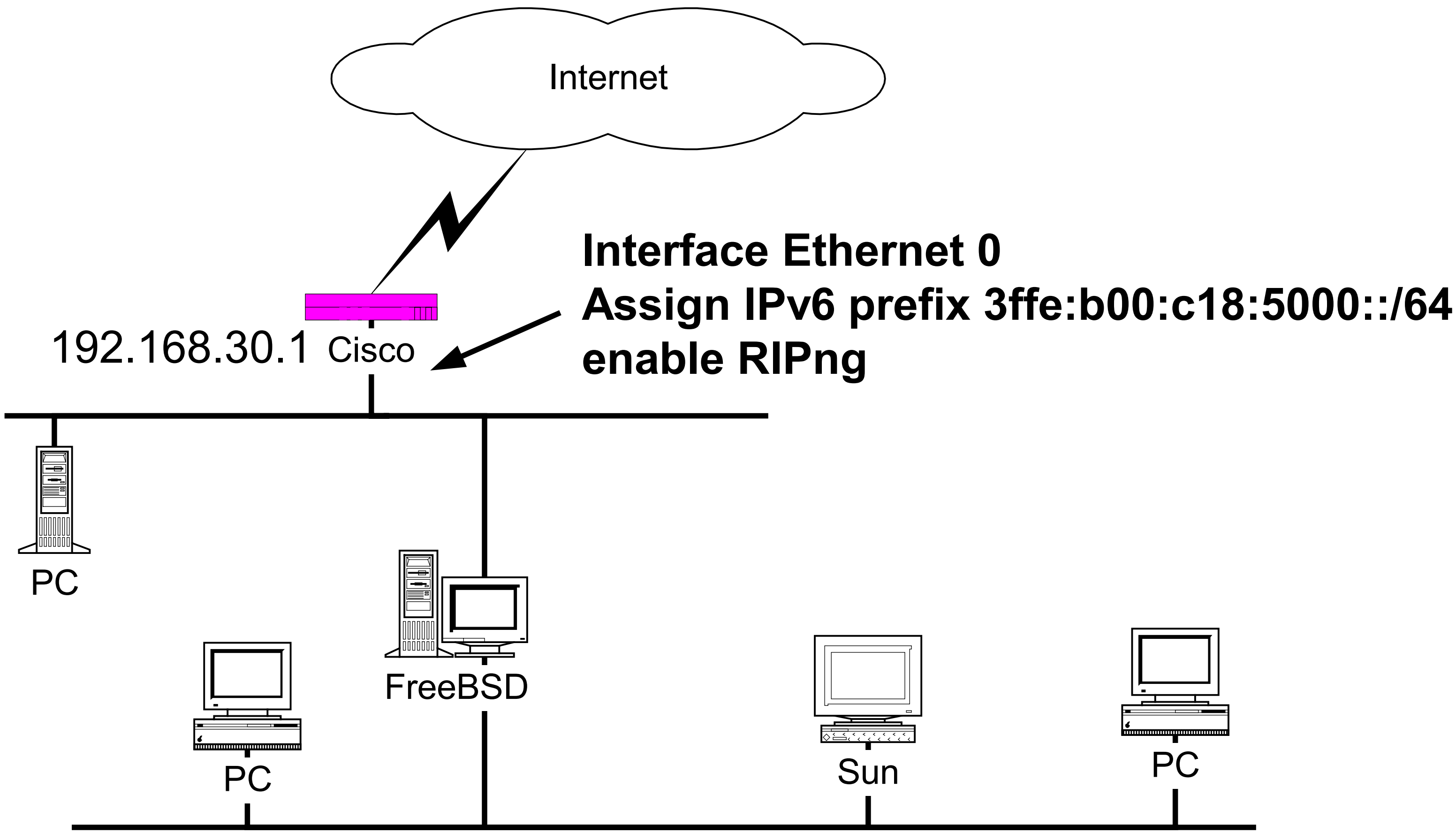
`show ipv6 bgp summary`

- State of IPv6 BGP neighbors

`clear ipv6 bgp neighbor`

- Resets peering with neighbor (* = all neighbors)

Cisco example

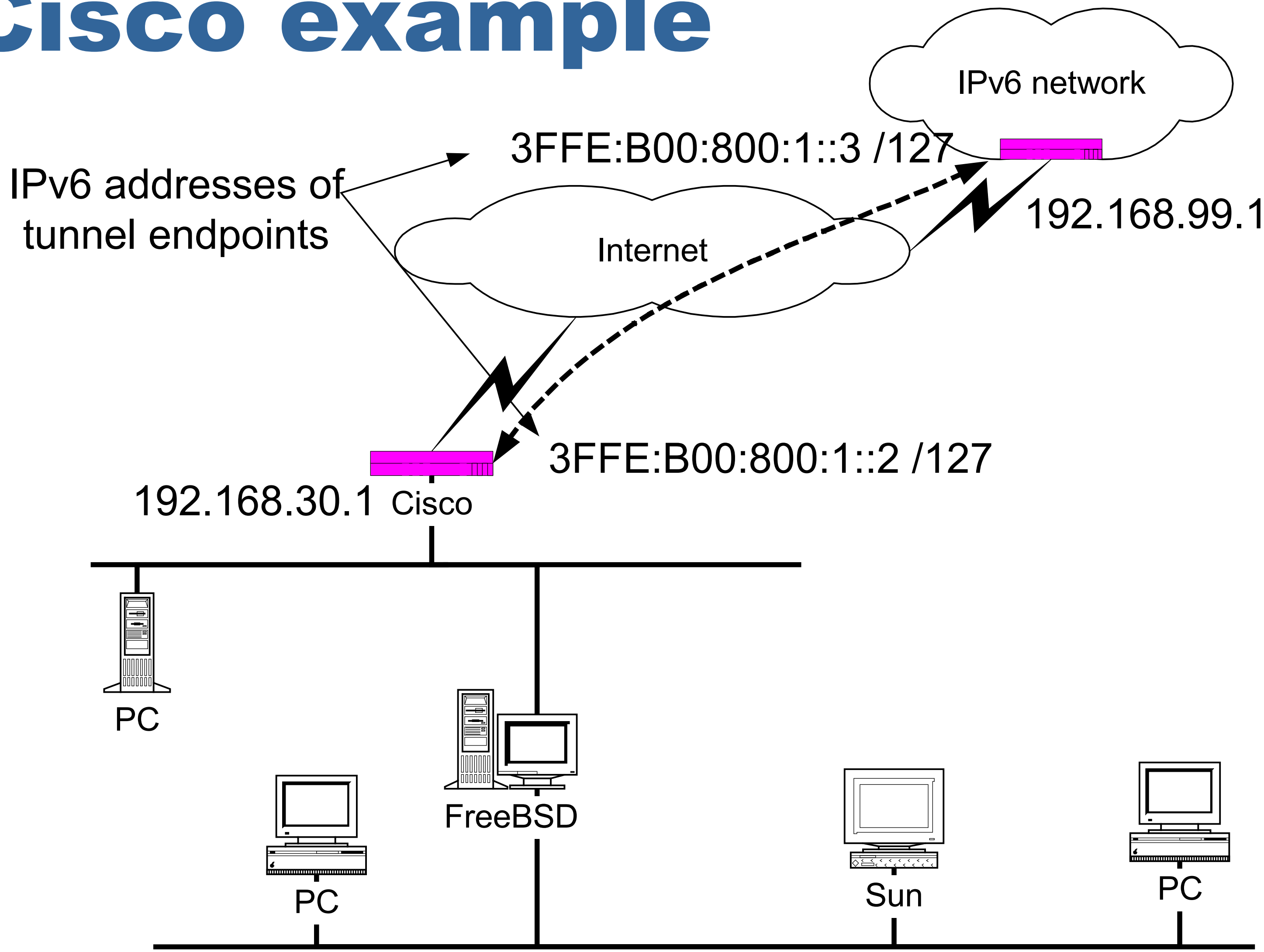


Cisco example

```
interface Ethernet0

 ip address 192.168.30.1 255.255.255.0
 ipv6 enable
 ipv6 address 3FFE:B00:C18:5000::0/64 eui-64
 ipv6 nd prefix-advertisement
  3FFE:B00:C18:5000::0/64 86400 86400 onlink
  autoconfig
 ipv6 rip TO enable
```

Cisco example

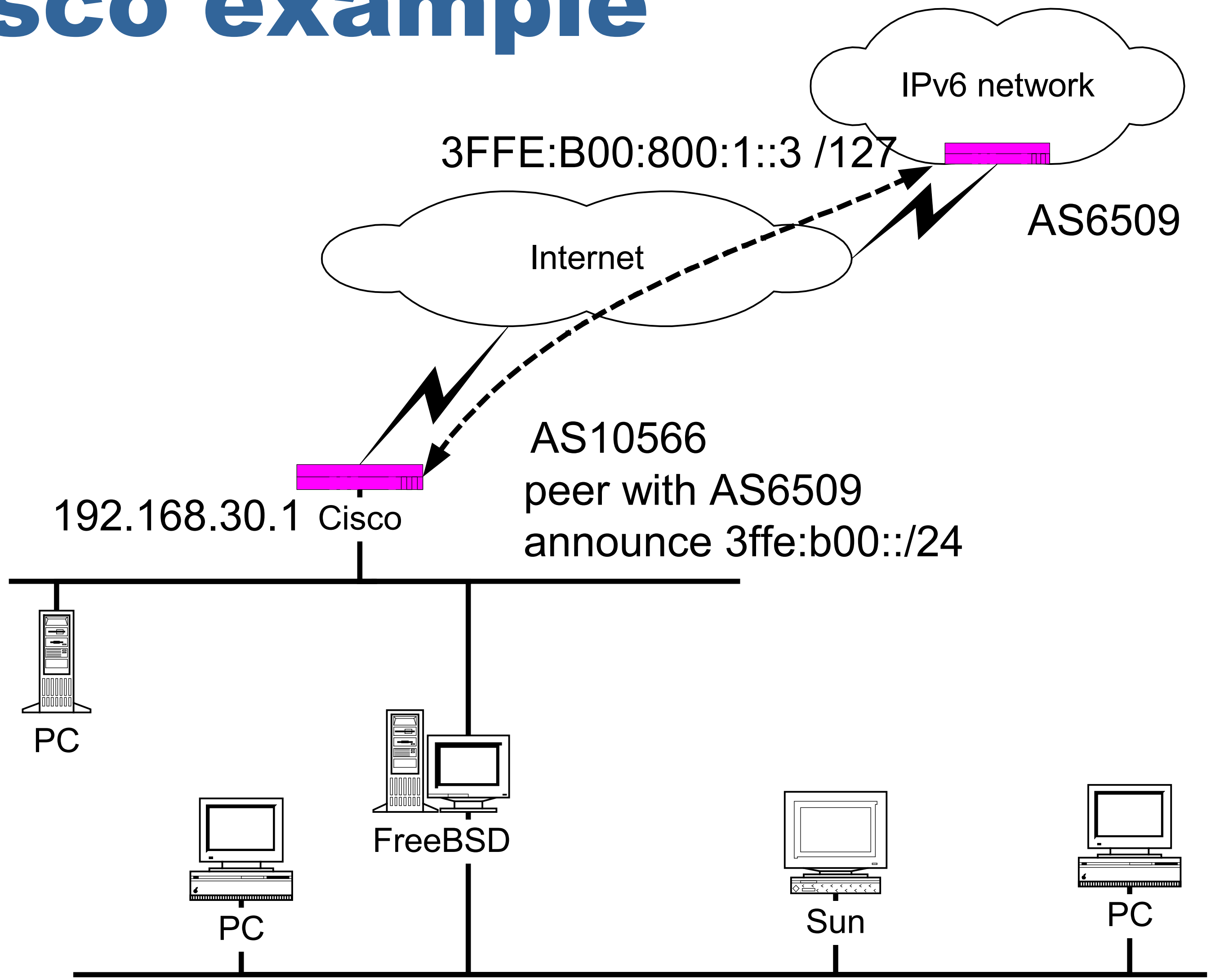


Cisco example

```
interface Tunnel0

 no ip address
 ipv6 enable
 ipv6 address 3FFE:B00:800:1::2/127
 tunnel source 192.168.30.1
 tunnel destination 192.168.99.1
 tunnel mode ipv6ip
```

Cisco example



Cisco example

```
router bgp 10566
ipv6 unicast-routing
ipv6 bgp redistribute connected
ipv6 bgp neighbor 3FFE:B00:800:1::3 remote-as
  6509
ipv6 bgp neighbor 3FFE:B00:800:1::3 route-map
  PrefNative in
ipv6 bgp network 3FFE:B00::/24 summary

route-map PrefNative permit 10
 set local-preference 200
```

3COM

- NetBuilder router
- supports IPv6 over Ethernet, FDDI, PPP, ATM, FrameRelay, X.25
- BGP, RIPng
- ping6, traceroute6, telnet6

3COM - enabling IPv6

- Enable IPv6 on Internal port:
  - SETDefault !<port> -IPV6 CONTrol = ROute
  - setdefault !1 -ipv6 control=route
- Configure static IPv6 address:
  - ADD !<port> -IPV6 NETaddr <IPV6 address>
  - add !1 -ipv6 netaddr 3ffe:b00:c18:2::/64

3COM - static route

- Adding/deleting a static route

ADD !<port> -IPV6 ROUte <IPV6 address> [<gateway>] <metric> [Override]
DELete -IPV6 ROUte <IPV6 address> [<gateway>]

  - add !1 -ipv6 route 3ffe::/16 fe80::250:3eff:fee4:4c00 1

3COM - tunnels

- SETD !<tunnel id> -IPV6 tunnel=<local IPV4 addr> <remote IPV4 addr>
- SETD !<tunnel id> -IPV6 CONTrol=ROUte

  – tunnel id t0 is reserved for automatic tunnels

  – setd !t1 -ipv6 tunnel=206.123.31.163 206.123.31.101
  – setd !t1 -ipv6 control=route
  – add !t1 -ipv6 netaddr 3ffe:b00:c18:500::2 noaddrconf

3COM - useful commands

- show -ipv6 address
- show -ipv6 conf
- show -ipv6 allroute
- show -ipv6 tunnel
- show -sys statistics -ipv6 | -bgp | -ripng

Merit MRTd routing daemon

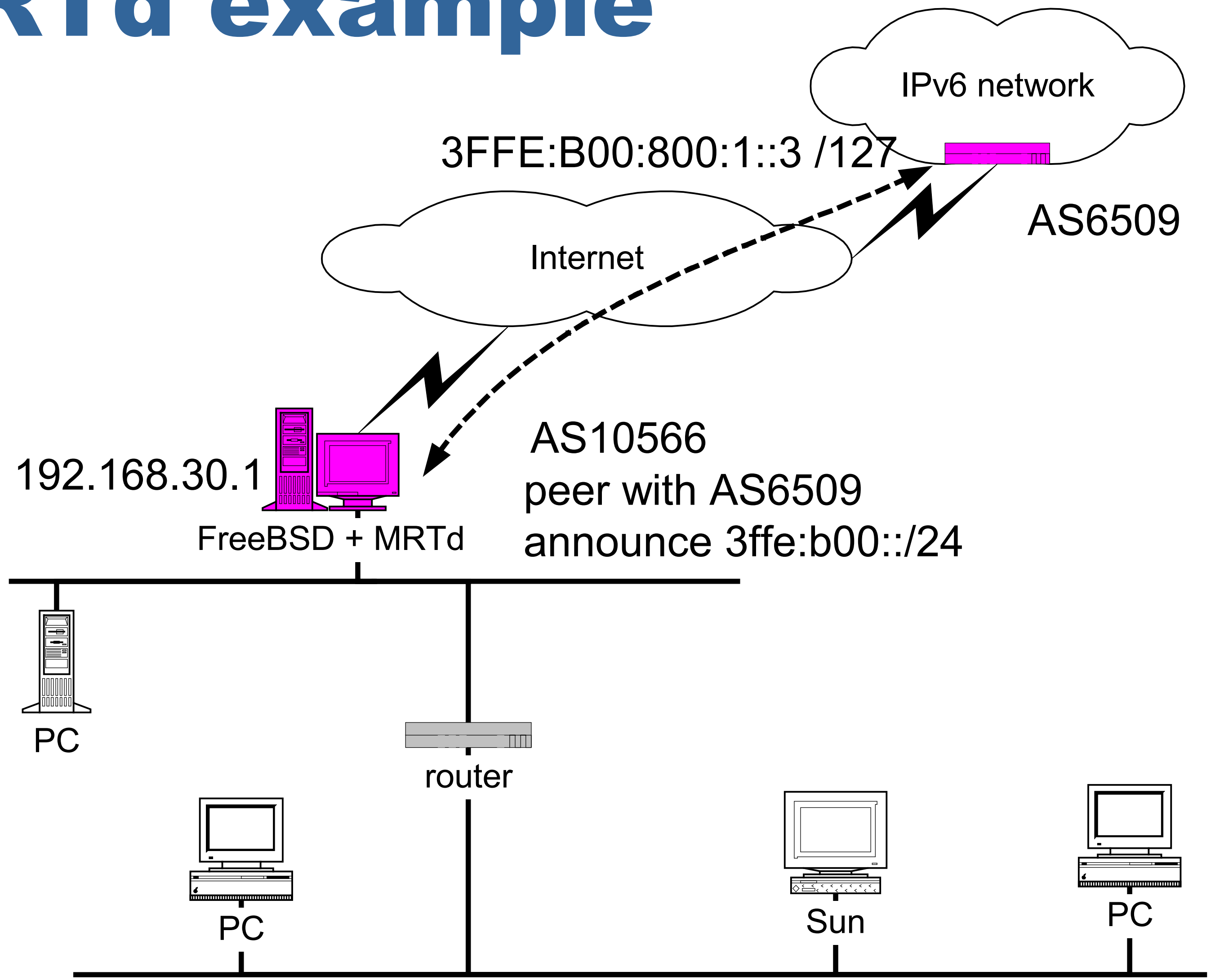
- Merit MRTd routing daemon (current version 2.2.1a)
- Runs on Solaris, FreeBSD, NetBSD, Linux and NT.
- IPv6 and multicast support
- BGP4+, RIPng
- Cisco-like configuration language
- Extensive tracing and logging

Merit MRTd routing daemon

- telnet interface

```
MRTd#
  clear
  config                         Configure MRTd
  dump
  enable                         Enable
  exit                           Quit from the current level
  kill                           Kill program
  load
  no
  quit                           Quit from the current level
  reboot                         Reboot MRTd
  show
  start
  trace
  write                          Save configuration file to disk
MRTd#
```

MRTd example



MRTd example

```
router bgp 10566
  redistribute static
  network 3ffe:b00::/24  ! announce our network
  aggregate-address 3ffe:b00::/24 summary-only
  neighbor 3FFE:B00:800:1::3 remote-as 6509
  neighbor 3FFE:B00:800:1::3 bgp4+ 1
```

RIPng configuration

```
router ripng
  network 3ffe:b00:c18:1::/64
  network 3ffe:b00:c18::b/127
  redistribute static
  redistribute bgp
```

Telnet interface

- provides an interactive user interface for management
- `telnet <IP address of router> 5674`

show bgp	*shows BGP peers and their status*
show bgp routes	*shows BGP routing table*
show bgp neighbors a:b:c::d errors	*shows errors with peer a:b:c::d*
show config	*shows the current configuration*
show ipv6	*shows IPv6 routing table*

Debugging from Telnet interface

MRT> show bgp

```
Routing Protocol is "BGP4+", Local Router ID is
   206.123.31.101, Local AS is 10566
Trace flags 0xf

peer 3ffe:1cff:0:fb::1 AS237 on gif0 [Established] 03:12:56
  Router ID 198.108.0.3 (index #1) eBGP4+ draft 1
  Local Address 3ffe:1cff:0:fb::2 (socket 14)
  KeepAlive 24  Starttimer Off  Holdtime 84  ConnectRetry Off
  Packets Recv 17992  Updates Recv 17990  Notifications Recv 0
  Packets Sent 13804  Updates Sent 13801  Notifications Sent 0
  Connections Established 1  Connections dropped 0
```

Debugging from Telnet interface

MRTd# show bgp summary

```
Routing Protocol is "BGP4+", Local Router ID is 206.123.31.101, Local AS is 10566
Neighbor                  V    AS Update(R/S)  Notify  Up/Dwn Hours State
3ffe:1cff:0:fb::1         +   237 18299/14048   0/0     1/0    3.27 Established
fe80::260:3eff:fe47:1530  + 10566     0/15060   0/0     1/0    3.27 Established
3ffe:2d00:1::9            +  5408  1944/14783   0/0     1/0    3.27 Established
3ffe:b00:c18::3           +  3748  8609/11443   0/0     1/0    3.26 Established
3ffe:b00:c18::d           +   561  5758/14677   0/0     1/0    3.27 Established
3ffe:b00:c18::f           +  6175  4779/14579   0/0     1/0    3.27 Established
3ffe:b00:c18::11          +  1930  3789/14527   0/0     2/1    2.98 Established
3ffe:b00:c18::13          +  4697     5/14820   0/0     1/0    3.23 Established
3ffe:3600::4              +  3462  2827/6743   73/0    72/72   0.01 Active
3ffe:b00:c18::21          + 10318   188/11195   1/0     2/1    1.64 Established
3ffe:1ce1:0:ff01::1       +     3     0/15070   0/0     1/0    3.27 Established
```

Plan

- Overview of IPv6
- DNS configuration
- Routing protocols
- Transition strategies
- Router configurations
- ***Host installation and configuration***
- How to connect to the IPv6
- IPv6 deployment on the Internet
- IPv6 industry support and trends

Host installation and configuration

- FreeBSD
- Solaris 8
- Windows NT

FreeBSD with Kame

- A merge of 3 IPv6 implementations
  - Kame, Inria and NRL
- Permanent development team
  - SNAP every Monday
- NetBSD1.4.1, FreeBSD 2.2.8 and 3.4, BSD/OS 3.1 and 4.1, OpenBSD 2.6
  - Will be included in the standard FreeBSD distribution

FreeBSD with Kame

- Need to recompile kernel and applications
- Important files
  - /usr/local/v6/etc/rc.net6

Native IPv6 config

- Edit the file /usr/local/v6/etc/rc.net6

ROUTER

ip6router = YES
iface="fxp0 fxp1"
prefix_fxp0 = "3ffe:b00:c18:5000"
prefix_fxp1 = "3ffe:b00:c18:6000"
start "rtadvd" (router advertisement daemon)
routing daemon started (default: route6d)

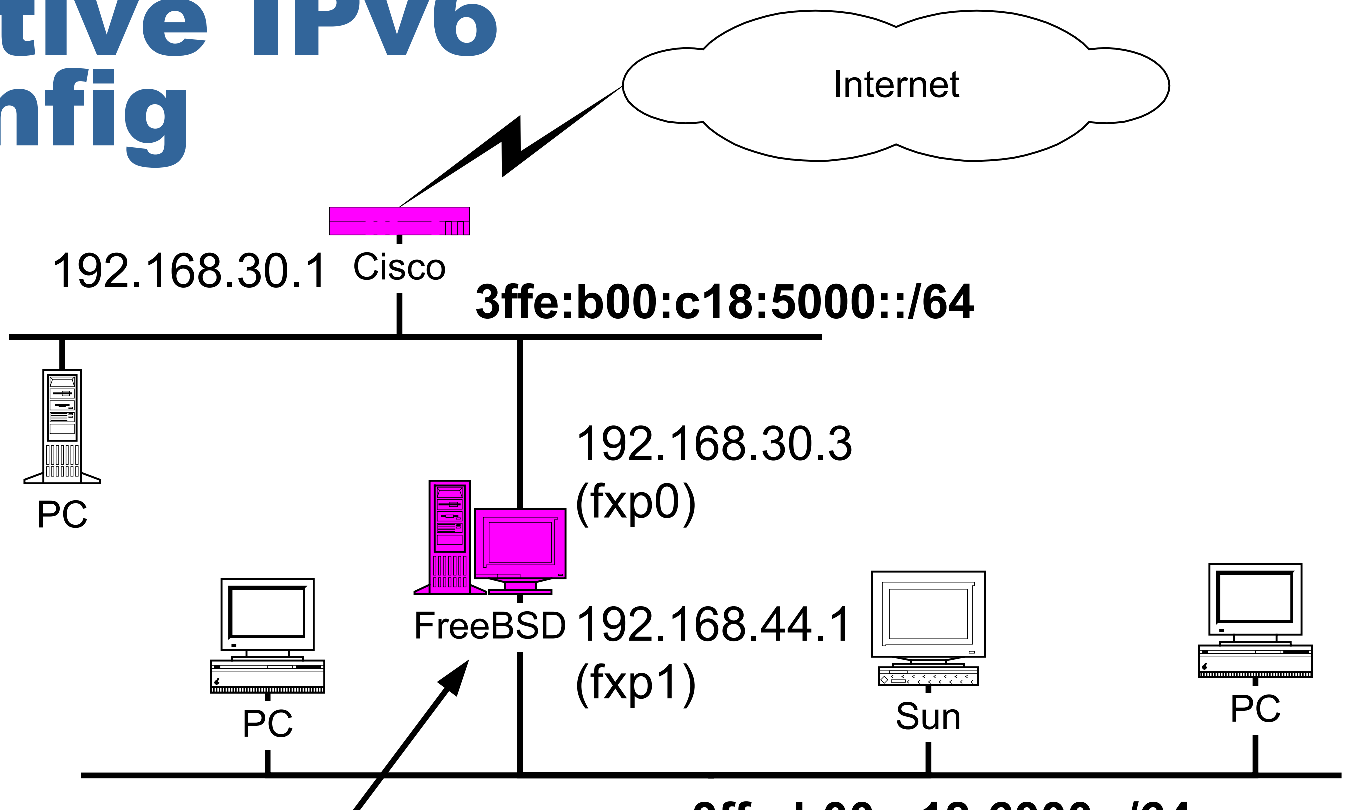
HOST

ip6router = NO
iface="fxp0"
ND (router solicitation) automatically used

Native IPv6 config

```
# ifconfig fxp0
fxp0:
flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu
   1500
    inet6 fe80:1::290:27ff:fe17:fc0f prefixlen 64
    inet 192.168.30.3 netmask 0xffffff00 broadcast
   192.168.30.255
    inet6 3ffe:b00:c18:5000:290:27ff:fe17:fc0f prefixlen 64
    inet6 3ffe:b00:c18:5000:: prefixlen 64 anycast
    ether 00:90:27:17:fc:0f
```

Native IPv6 config



ip6router = YES
prefix_fxp0 = "3ffe:b00:c18:5000"
prefix_fxp1 = "3ffe:b00:c18:6000"
start "rtadvd" (router advertisement daemon)
routing daemon started (default: route6d)

IPv6 tunnel

- 1. Build an IPv6 tunnel with source-destination values

```
# gifconfig gif2 inet 206.123.31.101 198.166.1.133
```

IPv4 source IPv4 destination

- 2. IPv6 address to the tunnel endpoints (numbered)

```
# ifconfig gif2 inet6 3ffe:b00:c18::a 3ffe:b00:c18::b
   prefixlen 127
```

IPv6 source IPv6 destination

- 3. Use default route (or use a routing protocol)

```
# route6 add  –inet6 3ffe::/16 3ffe:b00:c18::b
```

IPv6 tunnel

```
# gifconfig gif2
gif2: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 1280
    inet6 fe80:5::290:27ff:fe17:fc0f  prefixlen 64
    inet6 3ffe:b00:c18::a --> 3ffe:b00:c18::b  prefixlen 127
    physical address inet 206.123.31.101 --> 198.166.1.133


# ping6 3ffe:b00:c18::b
PING6(56=40+8+8 bytes) 3ffe:b00:c18::a --> 3ffe:b00:c18::b
16 bytes from 3ffe:b00:c18::b, icmp_seq=0 hlim=255 time=113.113 ms
16 bytes from 3ffe:b00:c18::b, icmp_seq=1 hlim=255 time=112.814 ms
16 bytes from 3ffe:b00:c18::b, icmp_seq=2 hlim=255 time=114.023 ms
```

Network debbuging with tcpdump (IPv6)

- Can decode IPv6 and BGP4+ packets

- To grab traffic from an IPv6 tunnel (protocol 41) to a file:

```
tcpdump -s 1500 -n -w /tmp/packets ip proto 41
```

- Visualize the packets:

```
tcpdump -r /tmp/packets -n -v | more
```

IPv6 on Solaris 8

- IPv6 is supported “out-of-the-box” !

- http://www.sun.com/software/solaris/ipv6/
- Manuals available on-line
  - http://docs.sun.com

IPv6 on Solaris 8

- Enabling IPv6 for a node
- For each network interface
  - Create empty file /etc/hostname6.<interface>
  - After reboot, autoconfiguration will assign address

IPv6 on Solaris 8

- Enabling IPv6 on a router
- /etc/inet/ndpd.conf
  - Router advertisement configuration
- RIPng (or install MRTd)

Configured tunnel on Solaris 8

- /etc/hostname6.ip.tun0
  - tsrc 206.123.31.101 tdst 198.166.1.133 up
  - addif 3ffe:b00:c18::a/127 3ffe:b00:c18::b up
- Run “/etc/init.d/inetinit start” to enable

Automatic tunnel on Solaris 8

- /etc/hostname6.ip.atun0
  - tsrc 206.123.31.101 ::206.123.31.101/96 up
- Run “/etc/init.d/inetinit start” to enable

IPv6 on Solaris 8

- /etc/inet/ipnodes
  - static list of IPv6 and IPv4 nodes
- /etc/nsswitch.conf
  - ipnodes: files dns
- NIS and NIS+ extensions for IPv6
- NFS and RPC IPv6 support

Microsoft Research Windows NT IPv6 stack

- http://www.research.microsoft.com/msripv6/

- Runs on NT 4 and Windows 2000
- Has host and router functionality
- Supports IPv6 tunneling
- Supports 6to4 transition mechanism
- Implemented as a separate protocol stack

MSR IPv6 applications and utilities

- ping6, tracert6, ttcp6, ftp6/ftpd6
- IPv6 version of wininet.dll
  - Can use Internet Explorer on IPv6
- Fnord! Web server
- SDR, RAT conferencing tool
- Network Monitor parser for IPv6

MSR IPv6 configuration

- Install
- If there is an IPv6 router in your network, you're configured (router solicitation)
- If not, configure a tunnel with an IPv6 peer...

MSR IPv6 tunnel configuration

- ipv6.exe rtu ::/0 2/::206.123.31.102 pub
  - Creates a tunnel with ::206.123.31.102
  - Creates a default IPv6 route to ::206.123.31.102
- ipv6.exe adu 2/3ffe:b00:c18:1fff:0:0:0:3
  - Assigns 3ffe:b00:c18:1fff:0:0:0:3 to tunnel endpoint

Plan

- Overview of IPv6
- DNS configuration
- Routing protocols
- Transition strategies
- Router configurations
- Host installation and configuration
- ***How to connect to the IPv6***
- IPv6 deployment on the Internet
- IPv6 industry support and trends

6Bone

- http://www.6bone.net
- IPv6 networks interconnected through tunnels and some native links (mostly IPv6 over ATM)
- RFC2471: IPv6 Testing Address Allocation
  - 3FFE::/16 6bone TLA
- Not a production network
- 6Bone routing policies (RFC2546, Informational)
- 6Bone registry provided (by ISI before), now Qwest (David Kessens)

6Bone

	March 1998	August 1998	April 1999
countries	32	35	41
ipv6-sites	240	302	385
inet6num	94	175	238

6Bone

- 6Bone hardening
  - Current 6bone is not necessarily highly stable, still carrying not-routable addresses, etc.
  - Review 6Bone Routing Policies document
- RFC 2772 : 6Bone Backbone Routing Guidelines
  - Route filtering
  - Tunnel peering strategies
  - BGP4+ monitoring, Registry, DNS

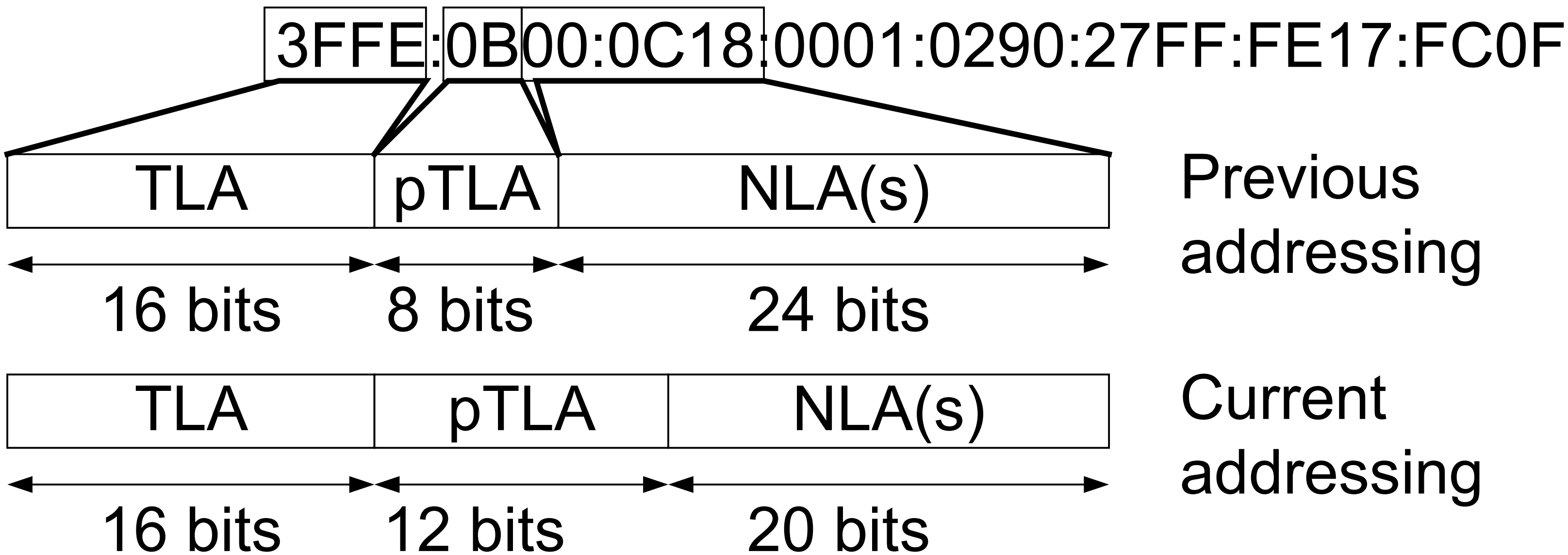
6Bone

- Address scheme changed to allow for more growth
  - draft-ietf-ngtrans-6bone-ptla-00.txt

- 3FFE:0000::/24 thru 3FFE:7F00::/24 old 8-bit pTLA space
- 3FFE:8000::/28 thru 3FFE:FFF0::/28 new 12-bit pTLA space

6Bone

New address scheme



How to become sub-Top Level Aggregator (sTLA)

- Methods (2) :
  - 6Bone Pre-Qualification for Address Prefix Allocation (6PAPA) by Bob Fink
  - Regional Internet Registries (RIRs) Guidelines for Requesting Initial IPv6 Address Space
    - Criterias until 100 requesting --> Bootstrap phase

Pre-Qualification steps (6PAPA)

- sub-TLA requestor (sTR) places sub-TLA request with its RIR (ARIN,APNIC,RIPE)
  - declaring intend to use pre-qualification process (6PAPA)
- follows the published process for becoming a pseudo-TLA (6Bone)
  - RFC 2546 (6Bone routing practice)
  - minimum time for joining the 6bone as end-site network to becoming a pTLA is 3 months

Pre-Qualification steps (6PAPA)

- sTR must operate a pTLA
  - at least minimum 3 months
  - with at least 3 delegations under its pTLA

- 6Bone steering group evaluates the sTR to be sure it has met the 6Bone routing practice

Pre-Qualification steps (6PAPA)

- After assignment of sTLA by RIRs to the sTR
  - optionnaly renumber from the 6bone pTLA prefix to the sTLA prefix
  - continue to use its pTLA
- References :

  draft-ietf-ngtrans-6bone-6papa-01.txt

Bootstrap Phase Criteria

- (1.) must have BGP peering relationship
  - at least 3 other public AS in the default-free

AND

- (2.) must demonstrate plan to provide production IPv6 service
  - within 12 months after receiving sTLA prefix
  - provide subsentiating document
    - engineering
    - deployment plan

Bootstrap Phase Criteria

AND EITHER

- (1.) must be an IPv4 transit provider
  - must show via RIRs Shared WHOIS it already has issue :
    - IPv4 space to at least 40 customer sites (/48)
  - must have an up-to-date routing policy registred in the RIRs database

OR

- (1.) must demonstrate it has actively participated in the 6Bone
  - for at least 6 months
  - operated a pTLA at least during 3 months

General criteria for initial sTLA allocation

- (1.) must have BGP peering with IPv6 networks
  - at least 3 others organisation that have already sTLA

AND EITHER

- (2a) have already deployed an IPv6 network and reassigned IPv6 addresses received from it upstream provider
  - at least 40 SLA customers
- (2b)must demonstrate plan to provide IPv6 service
  - with in 12 months after receiving sTLA space
  - subsential documentation (eng, deployment plans)

6Bone Registry

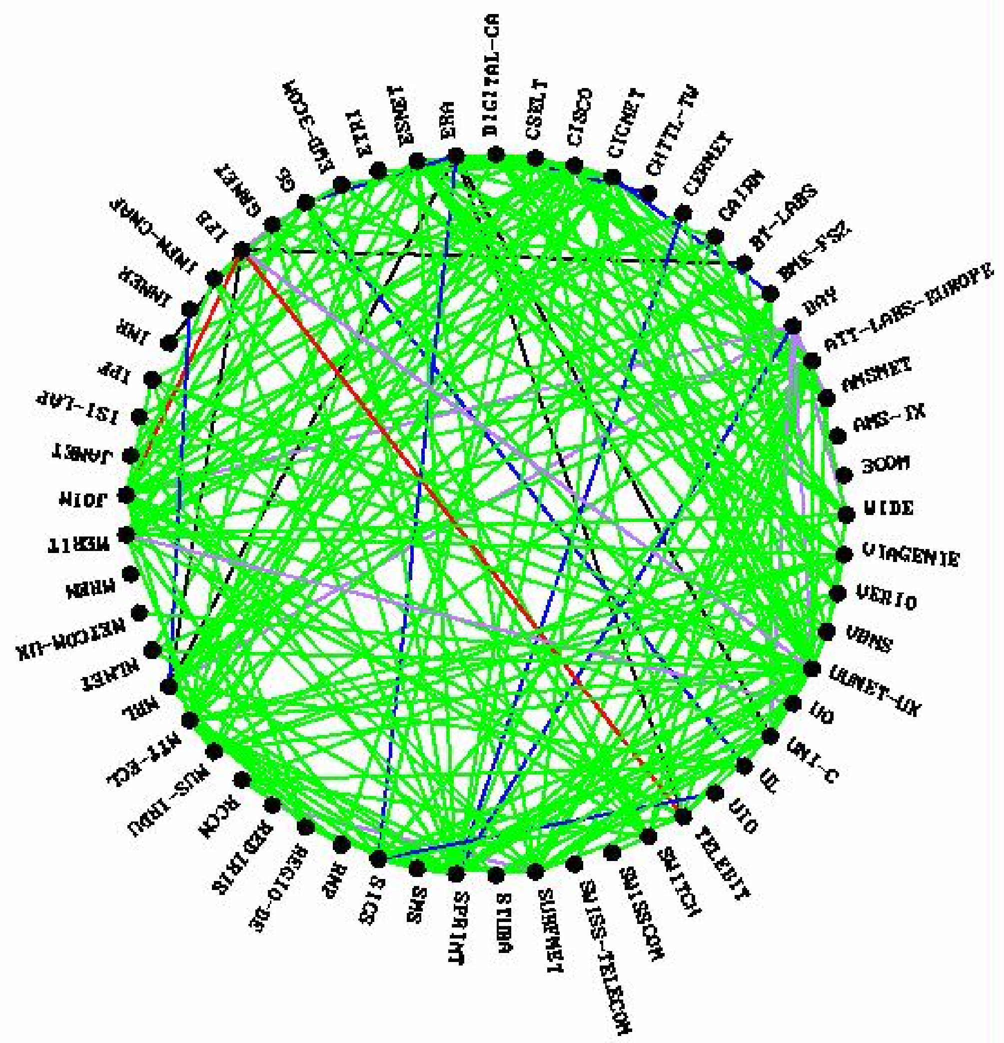
- RIPE style database with WHOIS interface
- Contains (should) every IPv6 sites connected to the 6Bone
- Primary database located at Qwest, mirror at Viagénie
- Can add, modify or delete any objects by e-mail
- 6Bone topology mapping uses registry data

Automatic drawings from registry

Service by

UK Lancaster Univ.

links between 6bone backbone nodes →



Whois Query

- Qwest 6Bone Whois Query
  - **whois -h whois.6bone.net OBJECT**
- Viagénie Whois Query
  - **http:://www.viagenie.qc.ca/en/ipv6/whois.html**
  - **whois -h whois.viagenie.qc.ca OBJECT**

6Bone registry Web interface

- Interface to create, update and maintain registry objects directly to the 6Bone registry
- Generic and extensible:
  - Parses the object descriptions in the database
  - Generates dynamic Web forms from the object descriptions
  - Shows required information for each objects
  - Supports multiple copies of attributes
- Password for mtners required
- Online help

Plan

- Overview of IPv6
- DNS configuration
- Routing protocols
- Transition strategies
- Router configurations
- Host installation and configuration
- How to connect to the IPv6
- ***IPv6 deployment on the Internet***
- IPv6 industry support and trends

6REN

- Not a network
- GOALS:
  - To provide production quality transit for IPv6
  - To develop operational procedures
  - To promote deployment
- Native IPv6 links over ATM mostly through the Startap
- Cairn (US) ,Canarie (CA), Chungwa (TW), ESnet (US), I2 (US), Ipfnet (DE), MCI/vBNS (US), NTT (JP), Renater (FR), Sprint (US), SURFnet(NL), WIDE (JP), SingAren (SG), Cernet(CN), Aarnet(AU)

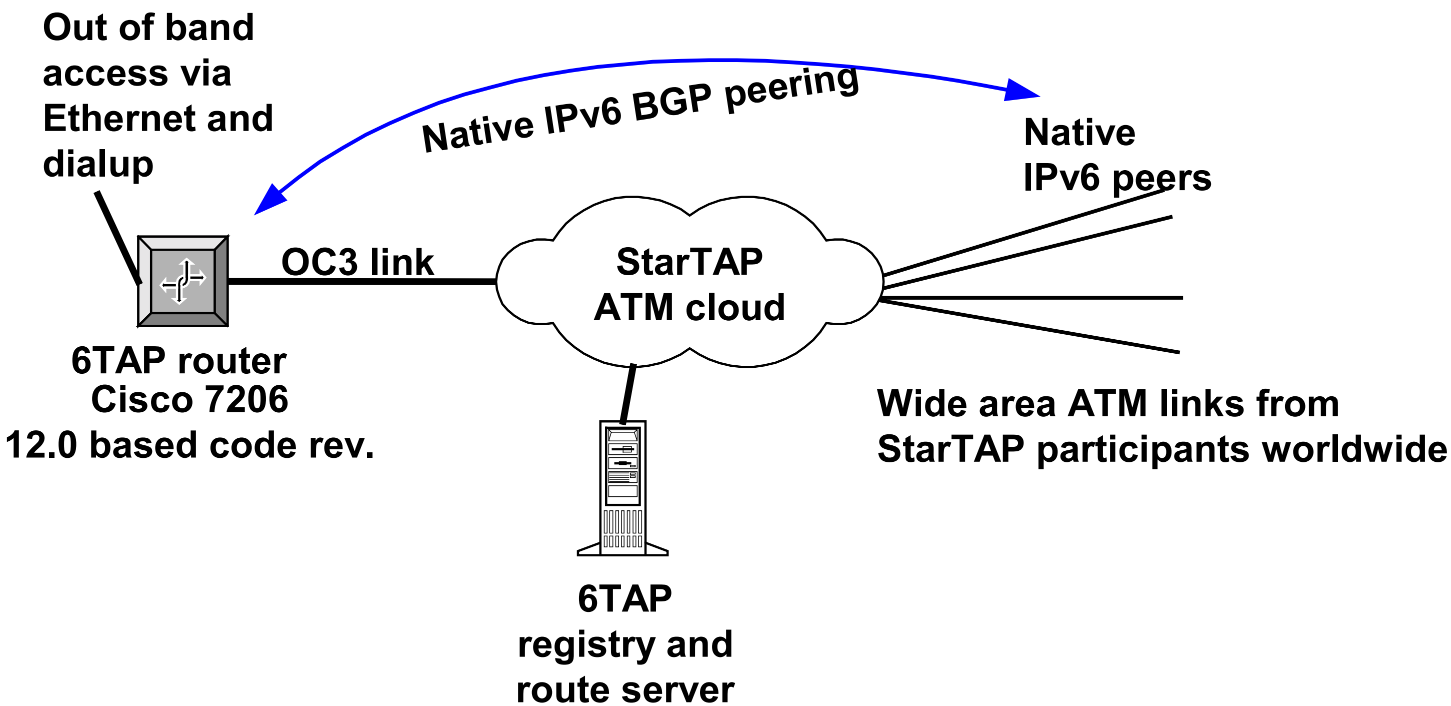
6TAP

- IPv6 exchange
- Facilitates interconnection of IPv6 production networks
- Qwest providing registry
- ESnet provides transit to 6Bone, IPv6 router and 24/7 operational support
- Canarie/Viagénie will develop and provide IPv6 route server (next phase)
- Co-located at StarTap (Chicago, US)

6TAP goals

- Exchange point for IPv6 networks
  - Located at the Chicago NAP (Ameritech) at the Startap (ATM exchange)
- First phase is a single router receiving all ATM IPv6 PVCs
- ESNet is providing transit trafic to the 6Bone

6TAP architecture



- R&E network already using v4 over ATM to reach StarTAP simply sets up a v6 ATM PVC path to the 6TAP router

6TAP operations

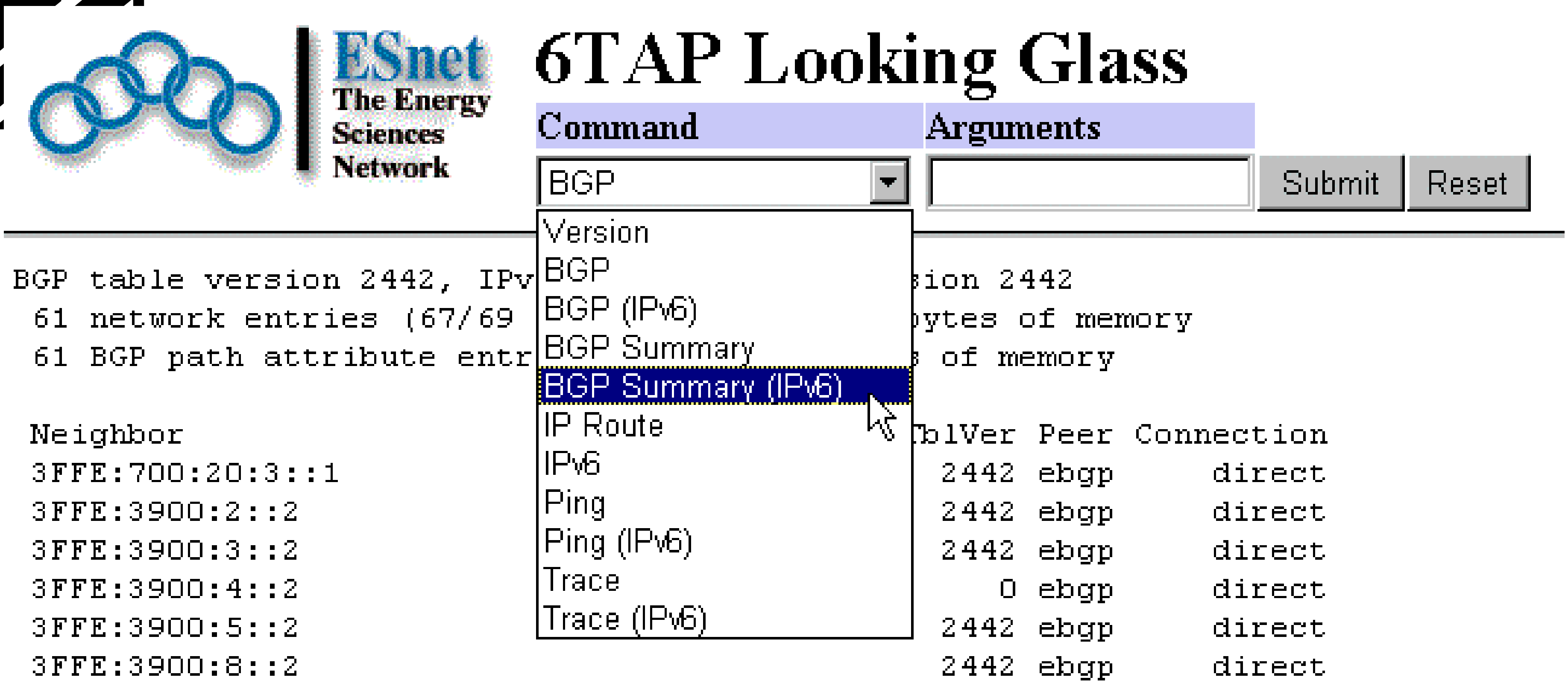
Equipment and operational support are being provided by:

- **Router**: ESnet
- **Route Server**:
  - Viagénie under a Canarie project will develop and provide an IPv6 route server, based on the MERIT MRT project.
  - Sun will provide servers for the route server
- **Registry:** QWEST (continuation of work started at ISI)
- **ATM switches/ports:** NSF (à la Star TAP)

6TAP services

- Web page at http://www.6tap.net
  - For establishing BGP peering sessions
    - route announcements
    - point-to-point address pref
    - AS number
    - contact info
    - etc.
  - Looking glass for routing info
    - BGP peering status (up/down)
    - MRTG Stats

Looking Glass



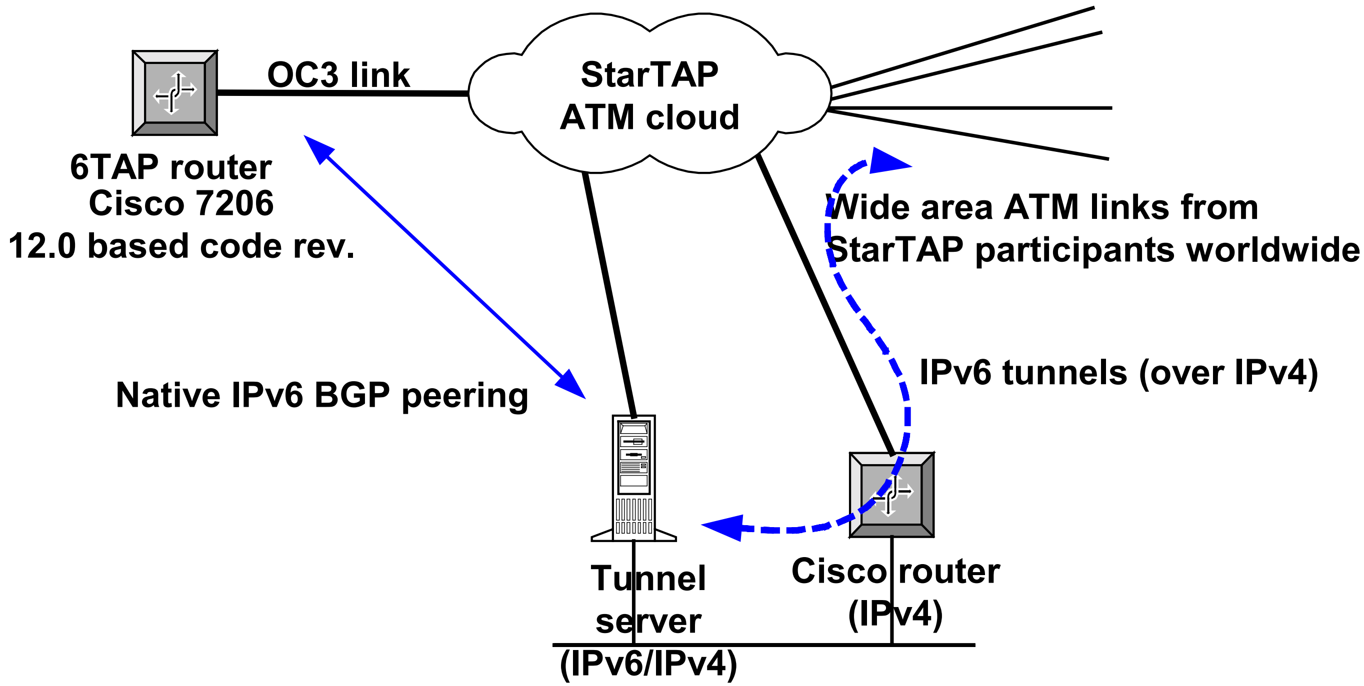
6TAP operations

- The 6TAP router supports native IPv6 peerings only: no tunneled IPv6 connections
- A dedicated server will provide IPv6 tunnels to sites that do not have ATM connectivity to the StarTAP

6TAP operations

- 6TAP: tunnel service
  - An extension of the 6TAP native link service
  - Enable IPv6 over IPv4 tunnel connections to the 6TAP
  - Project done by Viagénie

Tunnel peering to the 6TAP



- Cisco router provides IPv4 connectivity for the tunnel server
- Tunnel server has native IPv6 connectivity to the 6TAP router
- **Tunnel server provides IPv6 connectivity to sites without ATM links to the StarTAP**

6TAP participants

- Current participants

APAN Japan APAN Korea ESNET NTT-ECL CA*net 3	vBNS SingAREN, Singapore Univ. of Wisconsin-Madison CERN

Plan

- Overview of IPv6
- DNS configuration
- Routing protocols
- Transition strategies
- Router configurations
- Host installation and configuration
- How to connect to the IPv6
- IPv6 deployment on the Internet
- ***IPv6 industry support and trends***

Consortium

- IPv6 forum
  - Non-profit industry forum
  - Established in March 14th, 1999, at IETF in Minneapolis
  - Latif Ladid, VP Telebit.
  - Mission
    - to promote IPv6 (Internet Protocol version 6: the new Internet Protocol) by dramatically improving the market and user awareness of IPv6, creating a quality and secure Next Generation Internet and allowing world-wide equitable access to knowledge and technology, embracing a moral responsibility to the world.
- **The IPv6 FORUM will not develop protocol standards. The Internet Engineering Task Force has sole authority for IPv6 protocol standards.**

Consortium

- To this end the IPv6 FORUM will
  - Establish an open, international FORUM of IPv6 expertise
  - Share IPv6 knowledge and experience among members
  - Promote new IPv6-based applications and global solutions
  - Promote interoperable implementations of IPv6 standards
  - Cooperate to achieve an end-to-end quality of service
  - Resolve issues that create barriers to IPv6 deployment
- In order to achieve these objectives IPv6 FORUM will manage a set of projects that will contribute to the mission of the FORUM. The benefits of the FORUM will be shared on a fair, equitable and non-profit basis.

Consortium

- About 75 IPv6 Forum members
  - AT&T, MCI, Sprint, Sun, Cisco, IBM, Microsoft, 3Com, Compaq, Canarie, NTT, Nortel, Teleglobe, Thomson-CSF…
- http://www.ipv6forum.com/

IPv6 initiatives

- Nokia initiative to put IPv6 in the cellular stack
- Commcercial IPv6 exchanges in Japan and Holland
- 6init: European initiative (industrial members)

Plan

References

Some links on IPv6

- IPng wg: http://playground.sun.com/pub/ipng/html/
- 6Bone: http://www.6bone.net
- IPv6 users site: http://www.ipv6.org
- IPv6 Forum: http://www.ipv6forum.com
- 6ren: http://www.6ren.net
- 6Tap: http://www.6tap.net
- 6Bone registry: http://whois.6bone.net/~david/6bone/whois.html
- Viagénie 6Bone registry mirror and Web interface: http://www.viagenie.qc.ca/en/ipv6/registry

Some links on IPv6

- Kame distribution: http://www.kame.net/
- IPv6 applications: ftp://ftp.kame.net/pub/kame/misc/
- MRTd routing daemon: http://www.mrtd.net/
- Tunnel server: http://www.freenet6.net
- Tunnel broker: http//carmen.cselt.it/ipv6/download.html
- Viagénie’s IPv6 project: http://www.viagenie.qc.ca/

RFC

- "Core spec"
  - RFC2460, Internet Protocol, Version 6 (IPv6) Specification, S. Deering, R. Hinden, Draft standard, 1998-12-01.
  - RFC2461, Neighbor Discovery for IP Version 6 (IPv6), T. Narten, E. Nordmark, W. Simpson, Draft standard, 1998-12-01.
  - RFC2462, IPv6 Stateless Address Autoconfiguration, S. Thomson, T. Narten, Draft Standard, 1998-12-01.
  - RFC2463, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification, A. Conta, S. Deering, Draft standard, 1998-12-01.
  - RFC1981, Path MTU Discovery for IP version 6, J. McCann, S. Deering, J. Mogul, Proposed standard, 1996-08-01.
- Addressing
  - RFC2373, IP Version 6 Addressing Architecture, R. Hinden, S. Deering, 1998-07-01.
  - RFC2374, An IPv6 Aggregatable Global Unicast Address Format, R. Hinden, M. O'Dell, S. Deering, Proposed standard, 1998-07-01.
  - RFC2450, Proposed TLA and NLA Assignment Rule, R. Hinden, Informational, 1998-12-01

RFC

- Routing
  - RFC2080, RIPng for IPv6, G. Malkin, R. Minnear, Proposed Standard, 1997-01-01
  - RFC2283, Multiprotocol Extensions for BGP-4, T. Bates, R. Chandra, D. Katz, Y. Rekhter, Proposed Standard, 1998-02-01
  - RFC2545, Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing, P. Marques, F. Dupont, Proposed standard, 1999-03-01.
- Transition scenarios
  - RFC1933, Transition Mechanisms for IPv6 Hosts and Routers, R. Gilligan, E. Nordmark, Proposed standard, 1996-04-01.
  - RFC2185, Routing Aspects of IPv6 Transition, R. Callon, D. Haskin, Informational, 1997-09-01
  - RFC2529, Transmission of IPv6 over IPv4 Domains without Explicit Tunnels, B. Carpenter, C. Jung, Proposed standard,1999-03-01.
- API
  - RFC2292, Advanced Sockets API for IPv6, W. Stevens, M. Thomas, Informational, 1998-02-01
  - RFC2553, Basic Socket Interface Extensions for IPv6, R. Gilligan, S. Thomson, J. Bound, W. Stevens, Informational, 1999-03-01.

RFC

- IPv6 over “foo”
  - RFC2472, IP Version 6 over PPP, D. Haskin, E. Allen, Proposed standard, 1998-12-01
  - RFC2464, Transmission of IPv6 Packets over Ethernet Networks, M. Crawford, Proposed standard, 1998-12-01.
  - RFC2467, Transmission of IPv6 Packets over FDDI Networks, M. Crawford, Proposed standard, 1998-12-01.
  - RFC2470, Transmission of IPv6 Packets over Token Ring Networks, M. Crawford, T. Narten, S. Thomas, Proposed standard, 1998-12-01.
  - RFC 2491, IPv6 over Non-Broadcast Multiple Access (NBMA) networks, G. Armitage, P. Schulter, M. Jork, G. Harter, Proposed standard, 1999-01-01.
  - RFC2492, IPv6 over ATM Networks, G. Armitage, P. Schulter, M. Jork, Proposed standard, 1999-01-01.
  - RFC2497, Transmission of IPv6 Packets over ARCnet Networks, I. Souvatzis, Proposed standard, 1999-01-01.
- DNS
  - RFC1886, DNS Extensions to support IP version 6, S. Thomson, C. Huitema, proposed standard, 1995-12-01

RFC

- MIBs
  - RFC2465, Management Information Base for IP Version 6: Textual Conventions and General Group, D. Haskin, S. Onishi, Proposed standard, 1998-12-01.
  - RFC2466, Management Information Base for IP Version 6: ICMPv6 Group, D. Haskin, S. Onishi, Proposed standard, 1998-12-01.
  - RFC2452, IP Version 6 Management Information Base for the Transmission Control Protocol, M. Daniele, Proposed standard, 1998-12-01.
  - RFC2454, IP Version 6 Management Information Base for the User Datagram Protocol, M. Daniele, Proposed standard, 1998-12-01.
- Others
  - RFC2147, TCP and UDP over IPv6 Jumbograms, D. Borman, Proposed standard, 1997-05-01
  - RFC2375, IPv6 Multicast Address Assignments, R. Hinden, S. Deering, 1998-07-01.
  - RFC2473, Generic Packet Tunneling in IPv6 Specification, A. Conta, S. Deering, Proposed Standard, 1998-12-01
  - RFC2526, Reserved IPv6 Subnet Anycast Addresses, D. Johnson, S. Deering, Proposed standard, 1999-03-01.

Internet-drafts

- Addressing
  - Initial IPv6 Sub-TLA ID Assignments, draft-ietf-ipngwg-iana-tla-03.txt, Bob Fink, Bob Hinden, Steve Deering, Tony Hain, 2000-01-14.
  - A method for flexible IPv6 address assignments, draft-ietf-ipngwg-ipaddressassign-00.txt, Marc Blanchet, 1998-10-22.
  - Site prefixes in Neighbor Discovery, draft-ietf-ipngwg-site-prefixes-03.txt, Erik Nordmark, 1999-07-01
  - Separating Identifiers and Locators in Addresses: An Analysis of the GSE Proposal for IPv6, draft-ietf-ipngwg-esd-analysis-05.txt, Lixia Zhang, Allison Mankin, J. Stewart, Thomas Narten, M. Crawford, 1999-10-19
- Routing
  - OSPF for IPv6, draft-ietf-ospf-ospfv6-05.txt, Author(s) : Rob Coltun, John Moy, D. Ferguson, Organization : IETF, Working group : ospf, State : unknown, Status : unknown, Date : 1997-11-26, Size : 176250 bytes
  - Router Renumbering for IPv6, draft-ietf-ipngwg-router-renum-09.txt M. Crawford, 1999-07-01
  - Routing of Scoped Addresses in the Internet Protocol Version 6 (IPv6), draft-ietf-ipngwg-scoped-routing-03.txt, Brian Haberman, 2000-03-03

Internet-drafts

- DNS
  - DNS Extensions to Support IP Version 6, draft-ietf-ipngwg-dns-lookups-07.txt, Christian Huitema, Susan Thomson, M. Crawford, 1999-03-08.
- Transition scenarios
  - Categorizing Translators between IPv4 and IPv6, draft-ietf-ngtrans-translator-02.txt, K. Yamamoto, Munechika Sumikawa, 1999-10-19.
  - A SOCKS-based IPv6/IPv4 Translator Architecture, draft-kitamura-socks-ipv6-01.txt, Hiroshi Kitamura, 1999-02-08.

Internet-drafts

IPv6 over “foo”

- Transmission of IPv6 Packets over IEEE 1394 Networks, draft-fujisawa-ip1394-ipv6-03.txt, Kenji Fujisawa, 1999-05-07.
- Transmission of IPv6 Packets over Frame Relay Networks Specification, draft-ietf-ion-ipv6-fr-02.txt , Andy Malis, A. Conta, Martin Mueller, 1999-02-01.

Others

- Protocol Independent Multicast Routing in the Internet Protocol Version 6 (IPv6), draft-ietf-pim-ipv6-02.txt, Hal Sandick, Garry Kump, Brian Haberman, 1999-11-12.
- Mobility Support in IPv6, draft-ietf-mobileip-ipv6-10.txt, C Perkins, D. Johnson, 2000-02-10.
- Dynamic Host Configuration Protocol for IPv6 (DHCPv6), draft-ietf-dhc-dhcpv6-14.txt, C Perkins, Jim Bound, 1999-02-26.
- Extensions for the Dynamic Host Configuration Protocol for IPv6, draft-ietf-dhc-v6exts-11.txt, C Perkins, Jim Bound, 1999-02-26.

Acknowledgements

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

- Implementing IPv6 over CA*net2 has been done together with Dalhousie University (http://www.dal.ca)
- Some slides were inspired from Bob Fink's presentations

Conclusion

- Sections: Overview, Specifications, Deployment on the Internet, Registries, Transition strategies, DNS configuration, Router configurations, Host installation and configuration, References, Links
- Neither complete or exhaustive. Tried to be as accurate as possible.
- Hope this helps to understand IPv6