TLEN 5460 – TELECOMMUNICATIONS SYSTEMS LABORATORY

Lab 9: IP Version 6

Summer 2014

[redacted]

INTRODUCTION

Internet Protocol Version 6 (IPv6), also called IPng (IP next generation), is a set of specifications from the Internet Engineering Task Force that's basically an upgrade of IPv4.

WHY IPV6?

The main reason for moving towards IPv6 is because of the shortage of IPv4 addresses. NAT, which is a way to save public IPv4 addresses, breaks end-to-end transparency architecture of Internet. NAT is not a solution to save IP address mainly because of the fact that “NATs do not support the original Internet promise of true peer-to-peer and any-to-any communication”.[1]

Several protocols don’t pass through NAT: -

- IPsec - NAT changes address in the packet header - loss of integrity.
- 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.

The right question to be asked before asking do we need IPv6 is:

“Are we interested in a network that allows any IP electronic devices to communicate transparently to each other regardless its location on THE global net?” - Viagénie

Other reasons why one should use IPv6 and not IPv4 are:

- Huge Address Space
- Address Renumbering/Hierarchy/Mobility
- Multicast/Anycast
- Security (IPsec, Source Route)
- Flow Labels
- High Performance Design
- Jumbograms (packets > 64 KB)

WHAT’S NEW IN IPv6?

IPv6 protocol is usually installed as a software upgrade in most device and operating systems; and needs to be only activated or configured. The main features (new in IPv6 compared to IPv4) are:

Extended address space:
Address format is changed from 32 bits to 128 bits.

[1] http://www.potaroo.net/ispcol/2001-01/2001-01-ipv6.html

DHCP changes:

In IPv6 (compared to IPv4), the address is dynamically configured in 2 ways: -

i) Stateful and ii) Stateless

i. **Stateful Configuration**: This way of dynamic learning the IPv6 address is similar to IPv4. It is called stateful because, like in DHCPv4, the DHCP server keeps the status of IP address leased. The only difference in the stateful configuration of DHCPv6 and DHCPv4 is that DHCPv4 uses broadcast address for sending out DHCP updates, while DHCPv6 uses multicast address (as IPv6 has no broadcast address). In stateful configuration v6 (v4), unlike stateless configuration, the host learns the whole 128 bits (32 bits) from the DHCP server.

ii. **Stateless Configuration**: In this dynamic configuration method, the host (requesting for IPv6 address) learns 64 bit prefix from the attached router (i.e. the DHCPv6 server) and calculates rest 64 bit using its MAC address (this is in EUI-64 format). The stateless configuration uses NDP (Neighbor Discovery Protocol) for learning about the 64 prefix bits.

NOTE: Stateless autoconfiguration - DHCPv6 stateless works in a plug-and-play fashion. In other words, as soon as a host (capable TCP/IPv6 suite) is connected to a router’s (i.e. a DHCPv6 server) link (and IPv6 address configured on it), the host will immediately learn an IPv6 address (i.e. the host actually learns the 64 bit prefix and calculates itself the rest 64 bits using the MAC address in EUI-format) even though one hasn’t configured any DHCP commands on the router.

NOTE: Stateless autoconfiguration is used to learn only the host IPv6 address and its default gateway, not DNS server’s IPv6 address. To learn about DNS server, IPv6 node can either use DHCP stateful or DHCP stateless configuration (in stateless DHCP, the DHCP server doesn’t store the state of the information about client).

Simplification of IP header:

The IPv6 header, being simpler than IPv4, has fixed 40 bytes length. This allows faster processing.

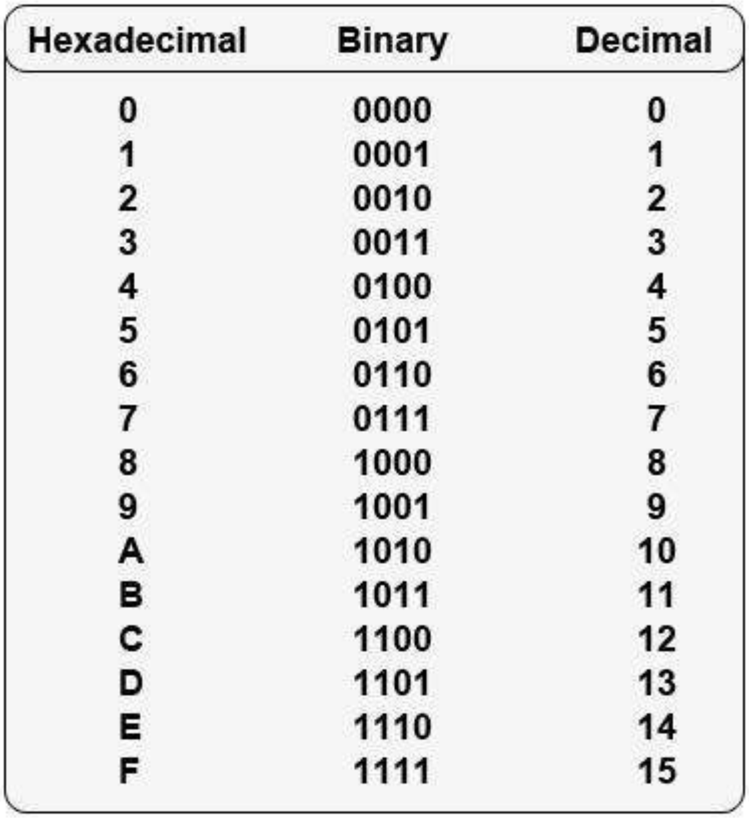
Improved Support in IPv6 header-Options and Extensions:

As compared to IPv4 (where the options are in the base header), IPv6 carries its options in an extension header which are inserted only if they are needed. This improves the processing speed of the headers. Using these extension headers, IPv6 can support Mobile IPv6, QoS and security.

IPv6 ADDRESSING

According to the RFC2460, the IPv6 address is represented as:
XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX, where XXXX is a hexadecimal number (representing 16 bits). X is a 4 bits value also called nibble. Below is a table showing

hexadecimal, its corresponding decimal and binary values.



Hexadecimal	Binary	Decimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

As this IPv6 address is too big to write, *following conventions* can be followed:

Omit the leading 0s in any give quartet (i.e. X).
e.g. ABCD:0056:0001:FE00 is similar to ABCD:56:1:FE00

Consecutive quartet of all 0s can be replaced with ‘**:**’
But only one such occurrence of ‘::’ can be present in a given
address. e.g. ABCD:0000: 0000: 0001: 0000: 0000: 0000:0056
is similar to ABCD::1:0:0:0:56 or ABCD:0:0:1::56

If you want to use the IPv6 address in the URL, for example for IPv4 it is
http://192.168.2.1/ For IPv6 it has to be http://[2001:1::3]/

IPv6 Prefix

In IPv4 - |Nw No.|Subnet No.|Host
no.| |Nw No.|Subnet No.| - Prefix
In IPv6 - |Prefix|host=Interface ID|

IPv6 – only classless addressing.

e.g 2000:1234:ABCD:1239:4:A:10A:9AC1/64 - IPv6 address.

2000:1234:ABCD:1239::/64 - Prefix. (also called network number).

Another example,
1111: 1111: 1111: 1111: 1111: 1111: 1111: 1111 /56 – IPv6 address

1111: 1111: 1111: 1100::/56 – Prefix (NOTE: always put the trailing 0s)

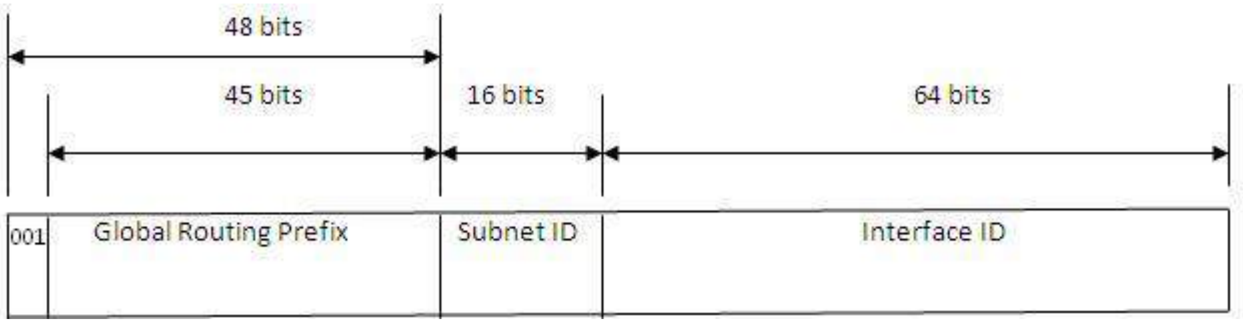
IPv6 address types

There are following address types in IPv6:

i) **Unicast IPv6 Address:**

a) Global Unicast

- Similar to public IPv4 address
- Global unicast address prefix assigned by ICANN
- Global unicast addresses have been designed to conform to the design plans for the IPv6 internet to support hierarchical addressing and hence, supportive of aggregated addresses for efficient routing.



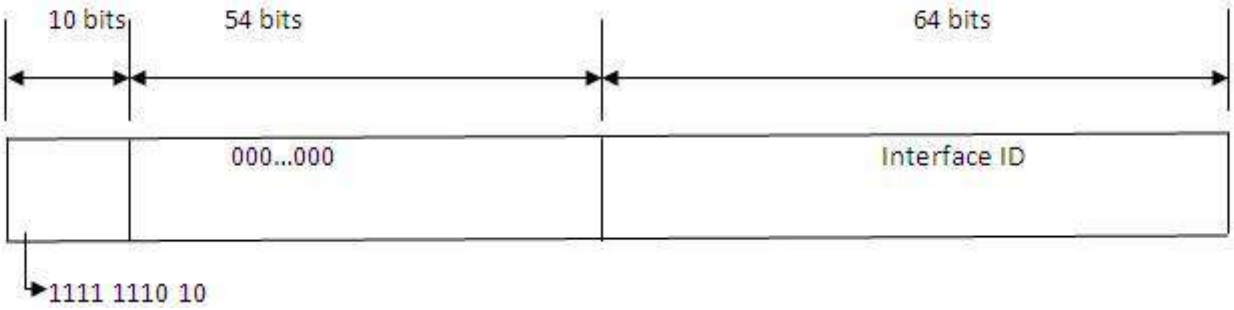
□ Header format:

i. 001 bits – 2(in Hex) is fixed.
ii. 48bit site prefix which is assigned to an individual by an organization. Routers on the IPv6 internet forward packets to organizations by matching requests with the 48-bit prefix. iii.
16-bits are collectively referred to as Subnet ID or Site topology and they are used for dividing a site into different subnets to prevent needless allocation of addresses.
iv. interface ID, which occupies the last 64-bits of the IPv6 Global unicast address. It often contains interface identifiers which are derived from the 48-bit MAC address of the network adapter.

□ 2XXX::/3 is the prefix for global unicast (like in IPv4 we have anything other than 10.0.0.0/8, etc) e.g 2001:0:4137:9e50:1885:102d:53ea:2dd5

b. Link Local Address

I) Similar to the automatic private IP range of addresses used in IPv4 (169.254.0.0/16) and their scope are confined to the local link i.e. they are never forwarded beyond the linkFE80::/64 is the prefix for link local address.



II) Format: 1) 1st 10 bits fixed (as FE8 in Hex)

2) 2nd rest 54 bits are 0's

3) 3rd the rest 64 bits are the EUI format Interface ID.

III) Used only within a subnet. This IP cannot leave outside its subnet. Example can router updates are sent with this address. Also while the host boots itself and gets an IPv6 address, host will initiate connection using NDP protocol where the packets are sent with source of link local address.

IV) A router will never send packet with an IP address of link local address into another subnet.

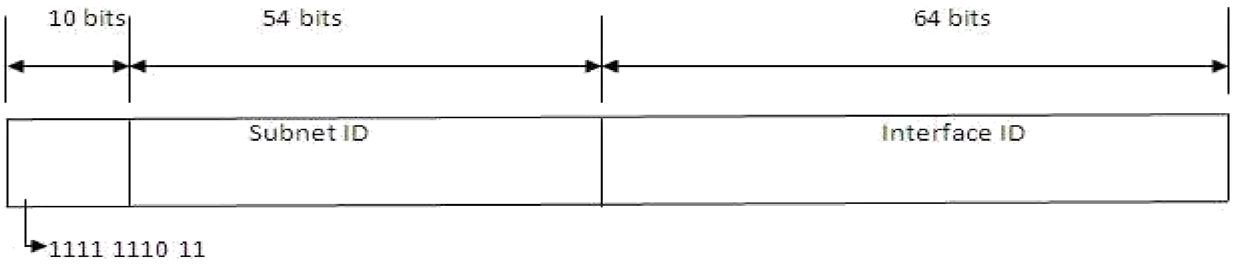
V) Link local addresses are always auto configured by the interface and they also require the neighbor discovery protocol.

VI) This IP is used only in places where packets need to be sent within the same subnet. Example, fe80::5efe:192.168.92.1 (with IPv4 address as Interface ID)
fe80::225:56ff:fe7e:4269 (with MAC EUI- format as Interface ID)

c) Site Local Address

I) Similar to Private IPv4 address.

II) FECX::/10 is the prefix for site local address. (similar to 10.0.0.0/8 for IPv4)



III) Format: Only has 1st 10 bits fixed, and then we have subnet bits and finally 64 bits of Interface ID.

IV) Supposed to be private IPv6. Therefore this IPv6 address should not go out of its autonomous system.

V) Came in 1995, RFC 1884

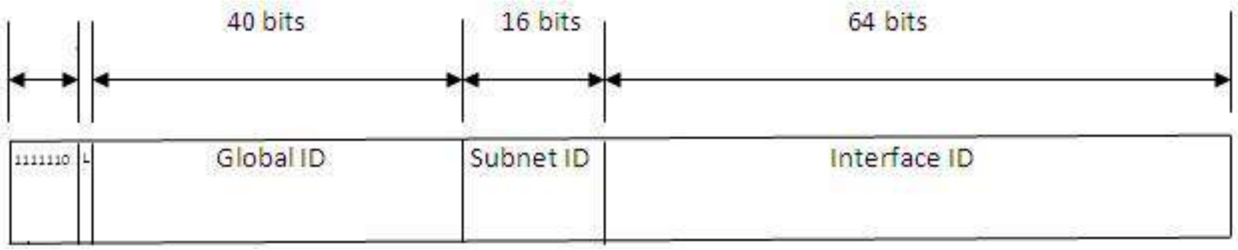
d) <u>Unique Local Address</u>

I) Similar to private IPv4 address

II) Came because of the same reason of site local address in 2005, RFC 4193; that is for private IPv6 address. Because the original defined site local addresses are not unique, this can lead to major problems, if two former independent networks would be connected later (overlapping of subnets). This and other issues lead to a new address type called Unique Local Address.

III) Just like in Site Local, this address is also supposed to be private IPv6. Unlike Site Local, Unique Local is globally unique due to the Global ID.

IV) fcXX::/7is the prefix for unique local (similar to 10.0.0.0/8 for IPv4)



VI) Format: 1) Fixed 1st 7 bits (1111 110)

2) 1bit = Local (L) – 1 means local, 0 not defined (therefore FD in Hex). Therefore we finally end up with FDXX::/8 as prefix.

3) The next 40 bits are referred to as the global ID which gains randomly assigned values enabling a unique 48bit prefix assigned to the site of the organization.

4) Similar to the global ID, the unique local offers a proceeding set of 16 bits used for subnetting within the organization.

5) Last 64 bits of Interface ID.

ii) <u>Multicast IPv6 Address:</u>[2]

I) Purpose is similar to that of IPv4 multicast.
II) Prefix is FFXX::/8

a) Multicast scopes

Multicast scope is a parameter to specify the maximum distance a multicast packet can travel from the sending entity. Currently, the following regions (scopes) are defined:

ffx1: node-local, packets never leave the node.
ffx2: link-local, packets are never forwarded by routers, so they never leave the specified link. ffx5: site-local, packets never leave the site.
ffx8: organization-local, packets never leave the organization (not so easy to implement, must be covered by routing protocol).
ffxe: global scope. others are reserved

b) Multicast types

There are many types already defined/reserved (see RFC 4291 / IP Version 6 Addressing Architecture for details). Some examples are:

- All Nodes Address: ID = 1h, addresses all hosts on the local node (ff01:0:0:0:0:0:0:1) or the connected link (ff02:0:0:0:0:0:0:1).
- All Routers Address: ID = 2h, addresses all routers on the local node (ff01:0:0:0:0:0:0:2), on the connected link (ff02:0:0:0:0:0:0:2), or on the local site (ff05:0:0:0:0:0:0:2)

c) Solicited node link-local multicast address

Special multicast address used as destination address in neighborhood discovery, because unlike in IPv4, ARP no longer exists in IPv6. An example of this address looks like ff02::1:ff00:1234 Used prefix shows that this is a link-local multicast address. The suffix is generated from the destination address. In this example, a packet should be sent to address “fe80::1234”, but the network stack doesn't know the current layer 2 MAC address. It replaces the upper 104 bits with “ff02:0:0:0:0:1:ff00::/104” and leaves the lower 24 bits untouched. This address is now used `on-link' to find the corresponding node which has to send a reply containing its layer 2 MAC address. This is how the work of ARP is done by NDP in IPv6 (for more information visit http://www-uxsup.csx.cam.ac.uk/courses/ipv6_basics/x84.html)

iii) Anycast Addresses:[3]

a) This type of IPv6 address is not present in IPv4

[2] http://tldp.org/HOWTO/Linux+IPv6-HOWTO/x513.html
[3] http://tldp.org/HOWTO/Linux+IPv6-HOWTO/x513.html

b) Anycast addresses are used only as destination addresses (like in multicast)

c) Anycast addresses main idea is: one-to-those who are nearest (like fore multicast is one-to-many or many-to-many)

d) These addresses are used along with a routing protocol that decides which nodes are closest. So for example, if we assign interfaces of different routers with same anycast address, then the source will reach these interface based on the routing protocol (the the nearest one gets the packet 1st).

e) Anycast address are no special address. They are taken from the same IPv6 Unique address pool.

f) They are used in place where one has to reach the nearest DNS server or DHCP server or similar dynamic groups.

g) Example: Subnet-Router Anycast Address. Assume that a node's address is 2001:db8:100:f101:210:a4ff:fee3:9566/64. Then, the anycast address could be created as 2001:db8:100:f101::/64

iv) Transition address

The following 5 types of addresses are made to ensure that there is a smooth transition from IPv4 to IPv6.

a) IPv4-Compatible address

This is an IPv6 address which is made by 1st 96 bits of IPv6 and last 32 bits of IPv4. This is in a case where a host (example Windows XP) is compatible with IPv4 and IPv6. Example fe80::5efe:192.168.36.1

b) IPv4-mapped address

These addresses are used to represent an IPv4 as a 128 IPv6 address. Prior to the IPv4 lies an IPv6 prefix::FFFF. Example::FFFF:192.168.2.1

c) 6to4 address

The 6to4 address is used for communicating between two nodes running both IPv4 and IPv6 over an IPv4 routing infrastructure. The 6to4 address is formed by combining the prefix 2002::/16 with the 32 bits of the public IPv4 address of the node, forming a 48-bit prefix. For example, for the IPv4 address of 131.107.0.1, the 6to4 address prefix is 2002:836B:1::/48 (131 decimal = 83 Hex, 107 = 6B Hex)

d) ISATAP address

Intra-Site Automatic Tunnel Addressing Protocol and it uses an address of the type 64bit prefix :0:5EFE:x.x.x.x where x.x.x.x represents the private IPv4 assigned to the node. E.g. Connection-specific DNS Suffix . : int.Colorado.EDU Link-local IPv6 Address . . . . . : fe80::**5efe**:172.21.210.42

e) Teredo address

This address is for the Teredo tunneling protocol (this protocol essentially will encapsulate an IPv6 packet into a IPv4 UDP datagram so that any hosts behind a NAT (or a firewall) capable to IPv6 can communicate with other IPv6 hosts in an IPv4 environment).

Example, 2001:0:4137:9e76:8000:f534:bc41:fb79[6]

v) Other addresses

a) Loopback - ::1 (similar to IPv4 127.0.0.1)
b) All 0s - :: (similar to IPv4 0.0.0.0)

IPv6 INTERFACE ID

The last 64 bits of the IPv6 address are said to be Interface ID. The reason being this Interface ID is being made with the help of the IPv6 address of the Interface (that interface to which the host is connected to) for which the Interface ID is being calculated. Most commonly, EUI format is used to calculate this Interface ID-
64 bit EUI- format: The ID is made by inserting FFFE perfectly between the 48 bit MAC address. Also, the 7th bit (from left e.i MSB) is set to 1 (this 7th bit is called U/L bit- U for Universal (when 0) meaning that the MAC address is burned and L for Local (when 1) meaning that MAC address has been configured).
Example, AB:CD:12:34:56:78 is a MAC address. To get the Interface ID for IPv6 address for the host’s interface that has this MAC address, we use the EUI format to get the following Interface ID – ABCD:12FFFE34:5678. Note that as the 7th bit was already 1, there was no change to it.

Zone ID

This is an integer that is attached to the IPv6 address with a % so as to inform the sending packet of its scope (e.i the zone). The ZoneID option specifies “the scope or zone of the destination for the ICMPv6 Echo Request messages”7. For example, if I am sending a packet from a zone 3 to a

[4] http://www.laynetworks.com/ipv6.htm
[5] http://msdn.microsoft.com/en-us/library/aa921071.aspx

zone 4, I will send a packet with zone 3. The zone is basically to define a site. In terms of link local address, an interface will have its zone. Zone is only useful in cases of link local/site local addresses to define their context. The place where zone is mainly useful is when we have many interfaces on a host connecting to different sites. Example: Link-local IPv6 Address . . . . . : fe80::1885:102d:53ea:2dd5%14

IPv6 Address Allocation

As mentioned earlier, the global unicast address is assigned by ICANN. The procedure of address allocation is as follows:

2001::/16	---	IANA issues this
2001:0400:: /24	---	IANA gives this particular address space to ARIN
2001:0410::/32	---	ARIN gives this space to an ISP
2001:0420::/32	---	ARIN gives another space to 2nd ISP (ISP 2)
2001:0420:0001::/48	---	ISP issues to first organization (Customer 1)
2001:0420:0002::/48	---	ISP issues to second organization (Customer 2)

NOTE Organization subnets with the 16bits (subnet ID) between the first 48 to 64bits.

2001:0400:0001:XXXX:YYYY:YYYY:YYYY:YYYY (where X reps the subnets and Y the interface identifier).

IPV6 TRANSISTION MECHANISMS

Now that we understand the basic concepts of IPv6, it is important to understand how IPv4 to IPv6 co-exists. In the present Internet scenario, we can have following types of nodes (hosts or routers):

b. IPv6 only – capable of IPv6 only
c. IPv4 only – capable of IPv4 only
d. IPv4/IPv6 – capable of IPv6 and IPv4

Also in the present Internet world, we can have IPv6 site (capable of only IPv6) that need connectivity to other IPv6 sites (also called islands) via IPv4 existing infrastructure. This can be achieved by following 2 ways:

a) Dual stack mechanism

In this mechanism basically all the devices in the site are IPv4/IPv6 capable.

[6] http://ipv6-or-no-ipv6.blogspot.com/2009/02/teredo-ipv6-on-windows-xp-better-than.html
[7] http://www.dslreports.com/forum/remark,13479280

b) **Tunneling**

This method is used in cases where one has IPv6 sites connected via IPv4 sites. In such a case, the IPv6 packets are encapsulated in IPv4 packets and the IPv6 can traverse through IPv4 infrastructure. Based on the type of tunneling and configuring the tunnels, we can have following types of tunnels:

c) **Manual**

- Generic routing encapsulation (GRE)
- IPv4-compatible
- 6to4
- Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)

Please read the following documents for knowing tunneling concepts and how to configure them:

For configuring: http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-tunnel.html
http://www.cisco.com/en/US/tech/tk872/technologies_configuration_example09186a00800b49a5.shtml

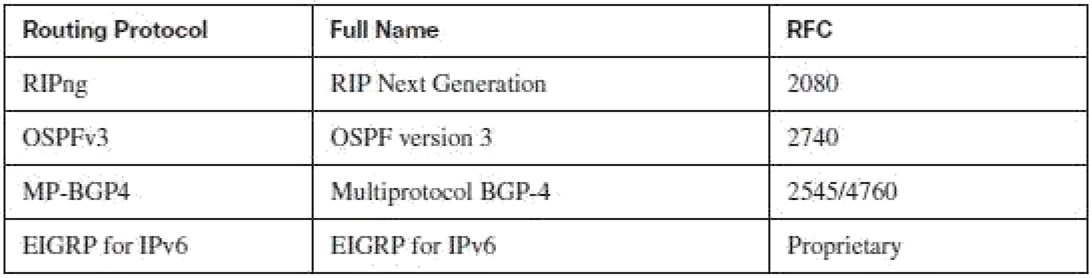
For concepts: http://www.cciecandidate.com/?p=592, http://www.faqs.org/rfcs/rfc1933.html

IPv6 NAT-PT:[8]

NAT Protocol Translation is a technique that is used for migrating from IPv4 into IPv6. NAT-PT is should only be used in case where one node with only IPv6 compatibility wants to connect to a node with only IPv4 compatibility. This mechanism is basically a last resort for migration from IPv4 to IPv6. For example, tunneling should be preferred over NAT-PT when one needs IPv6-IPv6 connectivity via an IPv4 infrastructure.
NAT-PT, as compared to NAT in IPv4, does bi-directional translation. For example, source and destination IPv6 addresses of a packet sent from IPv6-only node are both translated into IPv4 addresses. This is achieved via static mapping. Please read the following document for knowing how to configure static NAT-PT mapping:
http://blog.ine.com/2008/04/18/understanding-ipv6-nat-pt/

IPv6 ROUTING PROTOCOLS

The IPv4 routing protocols have been updated for IPv6. The basic idea behind all the routing protocols of IPv6 is same as that of IPv4 ones. Following is a table taken from Cisco Press ICND 2 to show the names and RFCs for IPv6 routing protocols:



Routing Protocol	Full Name	RFC
RIPng	RIP Next Generation	2080
OSPFv3	OSPF version 3	2740
MP-BGP4	Multiprotocol BGP-4	2545/4760
EIGRP for IPv6	EIGRP for IPv6	Proprietary

Please **read** the CCNA ICND2 IPv6 chapter for knowing how to configure RIPng and OSPFng. In short the major change in RIPng and OSPFv3 is that, you must configure the protocol globally (not for OSPFv3) and also doing the same in the interface mode.

http://www.potaroo.net/ispcol/2001-01/2001-01-ipv6.html 8

DHCPv6: -

The current development of the IPv6 Internet and the IPv6 address allocation recommendation RIPE267 has led to the need for Internet Service Providers (ISPs) to offer relatively large address blocks to an increasing number of customers. Every site with justification for more than one link is entitled to receive a /48 prefix allocation. The DHCPv6 prefix delegation feature can be used to manage link, subnet, and site addressing changes. DHCPv6 can be used in environments to deliver stateful and stateless information:

Stateful—Address assignment is centrally managed and clients must obtain configuration information not available through protocols such as address autoconfiguration and neighbor discovery.

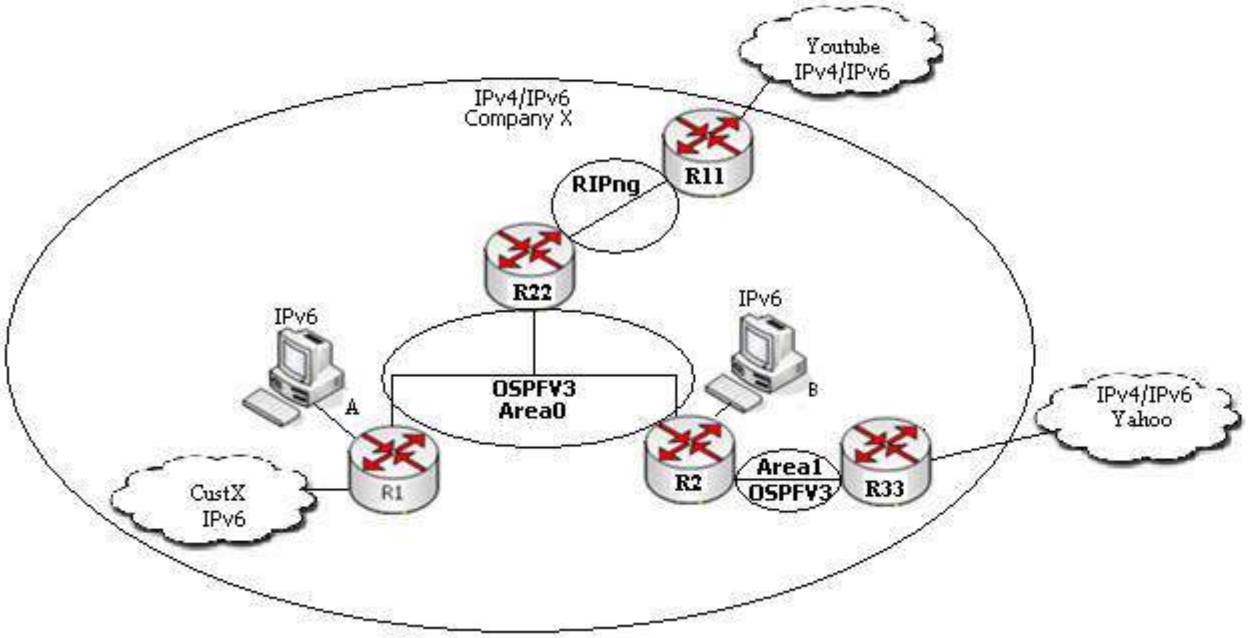
Stateless—Stateless configuration parameters do not require a server to maintain any dynamic state for individual clients, such as Domain Name System (DNS) server addresses and domain search list options.

Extensions to DHCPv6 also enable prefix delegation, through which an Internet service provider (ISP) can automate the process of assigning prefixes to a customer for use within the customer's network. Prefix delegation occurs between a provider edge (PE) device and customer premises equipment (CPE), using the DHCPv6 prefix delegation option. Once the ISP has delegated prefixes to a customer, the customer may further subnet and assign prefixes to the links in the customer's network.

IPv6 LAB NETWORK DIAGRAM:

During you LAB, please refer to the last pages for references and commands.

Objective 1:OSPFv3



ARIN has assigned 2005::/48 for your company (Company X). Read the following information about Company X and keep them in mind:

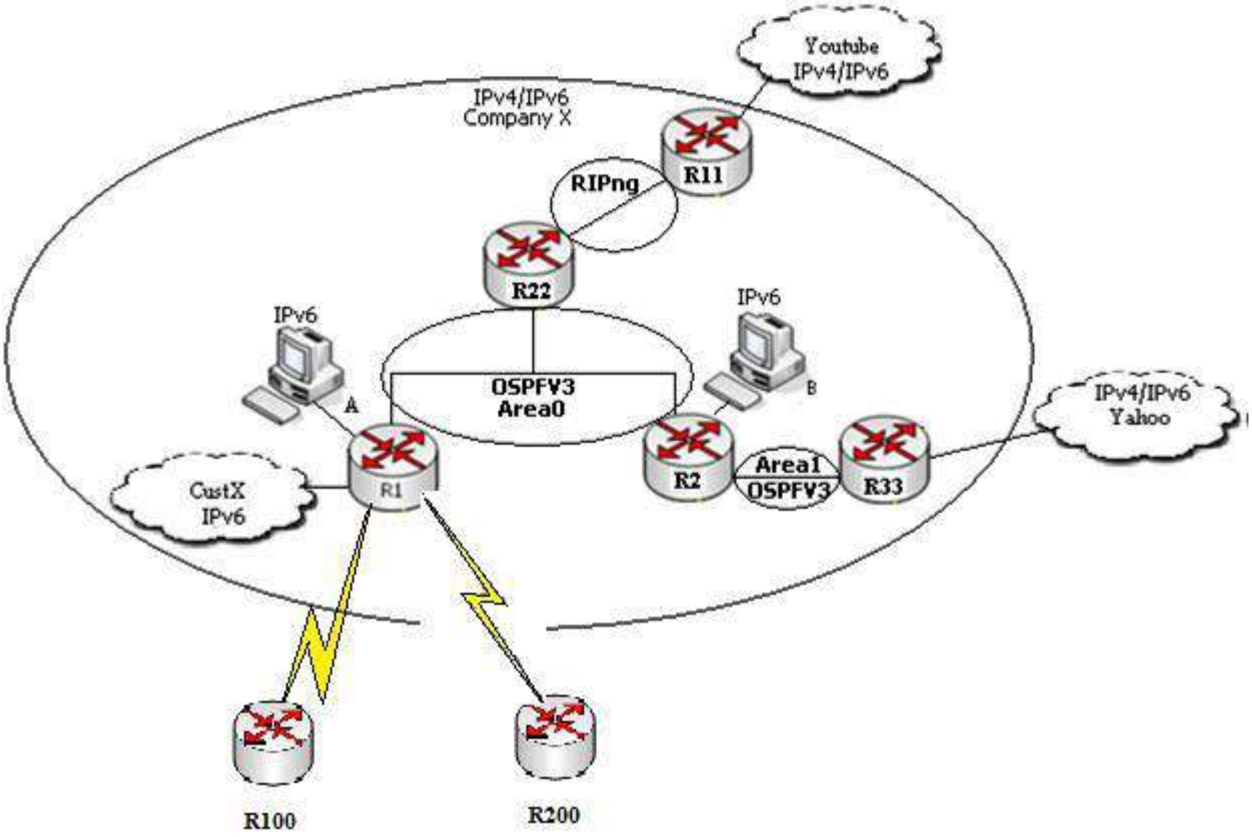
1) All the routers have dual stack (v4/v6) implemented in them.
2) R1, R2 and R22 are running OSPFv3 - area0.
3) R2 and R33 are running OSPFv3 – area1.
4) R22 and R11 are running RIPng
5) Youtube and Yahoo (loopbacks) are dual stack - IPv4/IPv6
6) CustX, PC A and PC B have only IPv6.

Following objectives must be met:

1) Assign only IPv6 addresses to PC A, PC B and custX and all the interfaces from the block provided by ARIN.
2) Use some other IPv6/IPv4 block (than Company X) for Yahoo and Youtube.

3) Run OSPFv3, RIPng according to above information so that the company has **end-to-end intra company connectivity** (includes all loopbacks and PCs)

Objective 2:DHCPv6: -



In the above objective you assigned your IPv6 addresses manually. Here you would need to use DHCPv6.

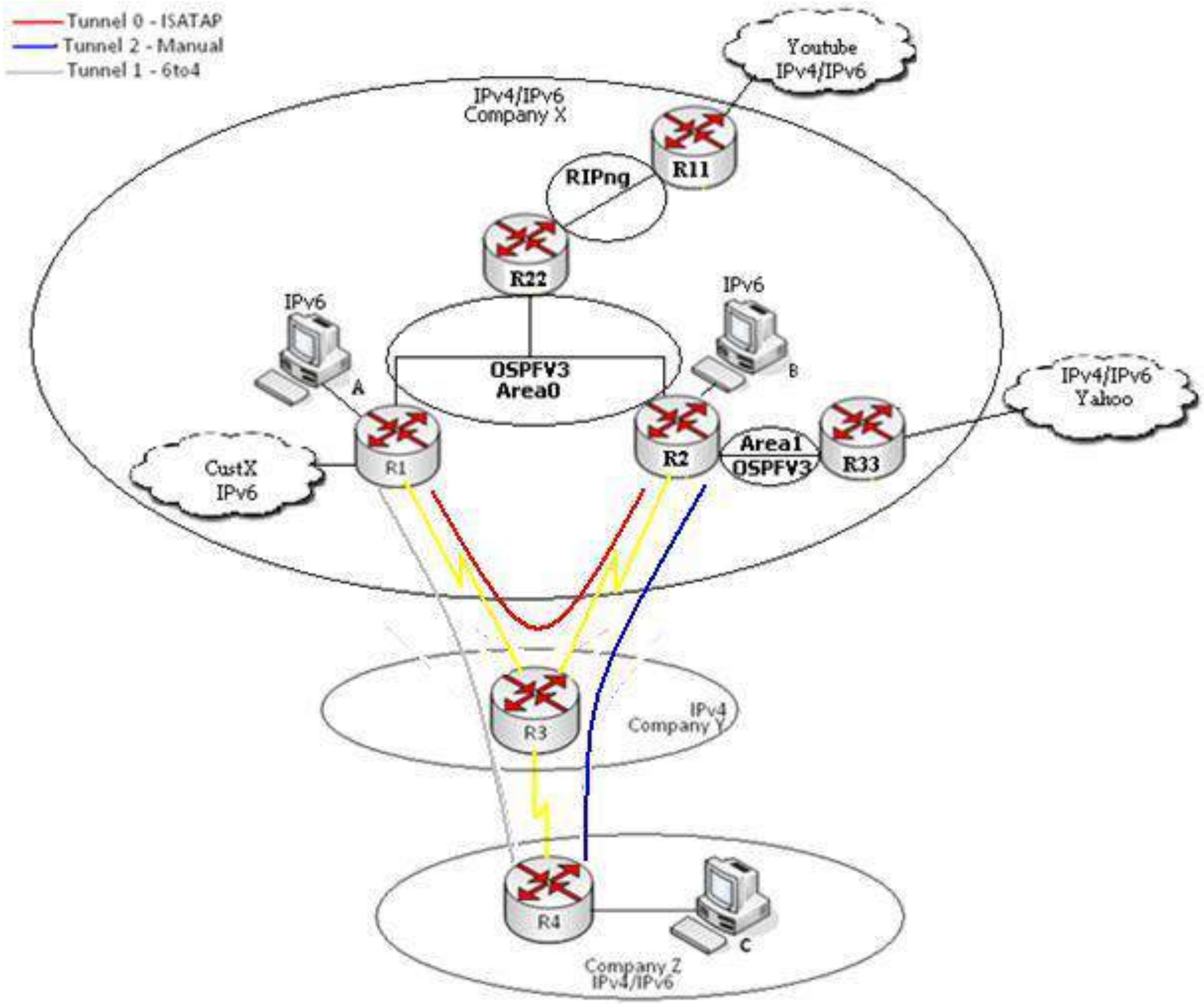
- Assume R100 belongs to the ISP X. ISP X has allocated the subnet 2005::/48 to Company X (remove the static ip addresses from R1 and PC A that you had assigned in the first objective) and will be assigning this subnet through DHCPv6.
- You need to get R100 to provide an ip from the 2005::/48 subnet to your R1 router's public interface.
- Also get the prefix to propagate on to PC A. Observe what ip is given to PC A. Is it the ip you anticipated the PC to get or is it something else. List your observations. Give a short summary of what you observe.
- R200 belongs to ISP Y. ISP Y will provide the subnet 2006::/48 to company X. Imagine Company X has terminated ISP X's contract and assigned it to ISP Y. Shut down the R1-R100 link and get R200 to propagate 2006::/48 to R1 and eventually to PC A. This should be plug and play (i.e. no new configurations on R1 are required, since it is already configured for DHCP at the first attempt itself when it was getting the IPs through R100). Achieve the same results as you did for ISP X.

The following document will help you set up the DHCP configurations: -

http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/whitepaper_c11-689821.html

Note: -Before continuing with Objective 3, remove the DHCPv6 configurations, get rid of R100 and R200 and configure it back to static IPs, the way it was done in objective 1.

Objective 3:IPv6 Tunneling



Company X wishes to connect to Company Z. But as company Y is only IPv4 compatible, company X can talk to company Z only via IPv6 tunneling. ARIN has given 2010::/48 to company Z. NOTE: routers only in company X and company Z are dual stack.

Achieve the following objectives in order.

1) Objective 3.1: ISATAP tunneling

As company X is connected to company Y now, company X wishes that PC A should always be able to connect to PC B even when the switch link fails. This will be achieved via setting up a tunnel 0 (ISATAP tunnel) between R1 and R2 through R3. Follow the conditions given below to achieve PC A and PC B connectivity via tunnel0:

i) Use only 10.0.0.0/16 IPv4 addresses for all the links coming out from R3.
ii) **R3 (being in company Y) should never have any IPv6 configurations.**

iii) Use RIPv2 to get IPv4 connectivity between all the routers –R1, R2, R3, R4.

iv) Tunnel 0 must be of mode – ISATAP. Therefore, assign carefully IPv6 addresses to the tunnel interfaces. (HINT: go through the IPv6 addressing section again to know the proper addressing for ISATAP)

v) Run OSPFv3 on tunnel 0.

Once your tunnel is up, **make sure that PC A should still talk to PC B via tunnel 0 (fail switch link between R1 and R2).**

2) Objective 3.2: 6to4 tunneling

Now that Company X has link redundancy, it wishes to take a new step by getting connected to company Z. Somebody decides that the connectivity should be achieved via 6to4 tunneling (called as tunnel 1). Follow the conditions (which will help you) to get end-to-end connectivity between the two companies:

i) 1st get complete IPv4 connectivity between Company X, Y and Z. HINT: this IPv4 connectivity is needed only for tunneling purpose like in previous objective.

ii) Again, R3 being in Company Y should not have any IPv6 configurations.

iii) As ARIN assigns a 2010::/16 prefix for company Z, use it for intra network (e.i link between PC C and R4).

vi) Tunnel 1 must be of mode – 6to4. Therefore, assign carefully IPv6 addresses to the tunnel interfaces. (HINT: go through the IPv6 Addressing section again to know the proper addressing for 6to4)

After your tunnel is up, **get complete connectivity. i.e. between companies X and Z (Yahoo, Youtube, and all other PCs).**

What do you notice? Do you still have connectivity between PC A and PC B via tunnel?

3) Objective 3.3: Manual tunnel

Now that the companies realize that, one cannot run ISATAP and 6to4 over same interface, they decide to run a manual tunnel from R2 to R4 for connecting company X and Z via company Y.

Again, follow the conditions to create a manual tunnel between R2 and R4:

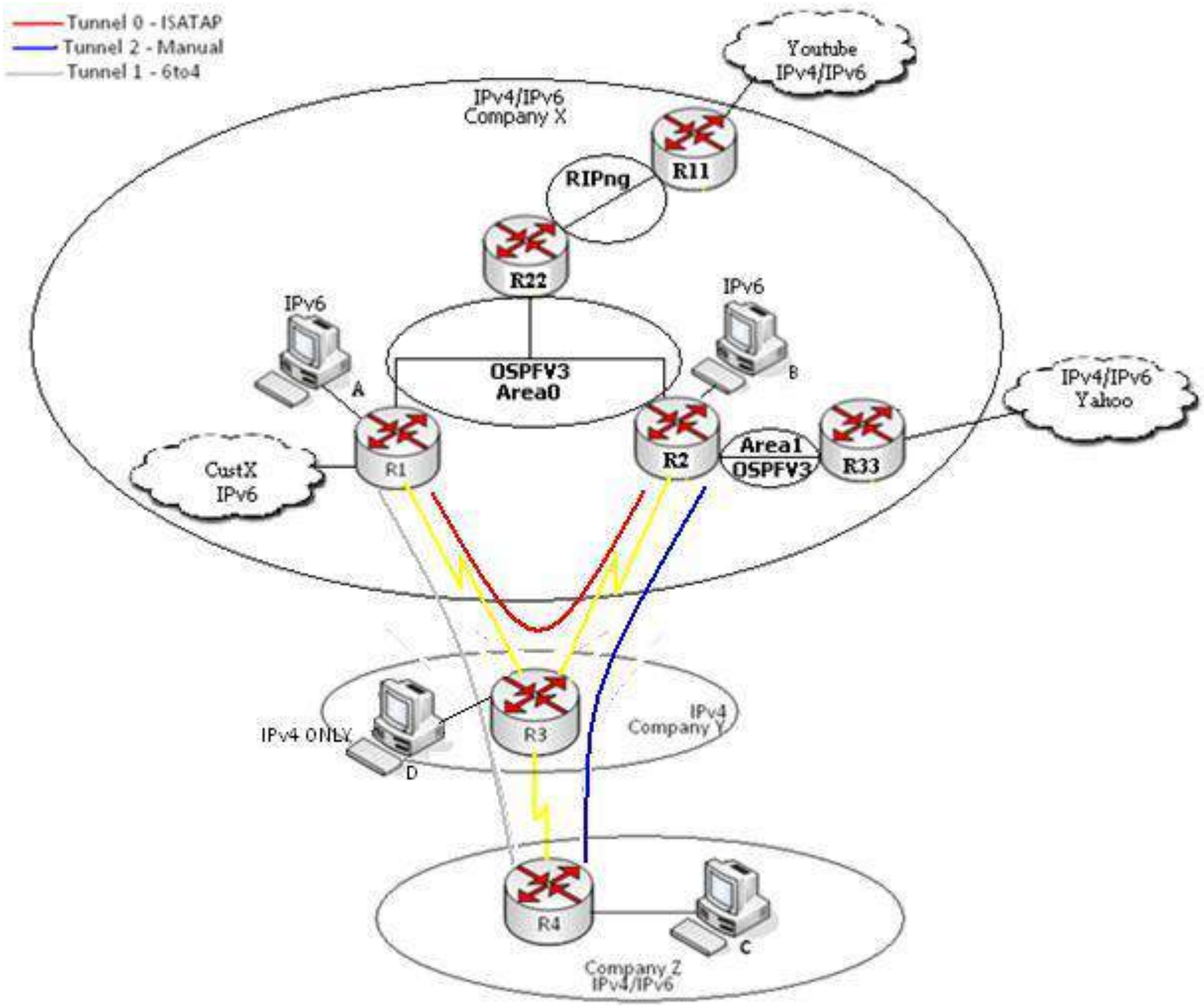
i. Create a manual tunnel 2.

ii. Use IPv6 address for tunnel 2 from Company X"s IPv6 pool.

iii. Run OSPFv3 on tunnel2.

Finally, get full v6 connectivity between all the PCs (and loopbacks).

Objective 4:IPv6 NAT-PT



After ages (company Y has received a lot of revenue from other companies), the company Y wishes to get connectivity‟s PCwithD company(having onlyZ.ToIPv4be more specific, company Y compatibility) should now connect to PC C (which is having only IPv6 address). This can only be achieved by NAT-PT. Follow the conditions given below to achieve connectivity between PC D and PC C:

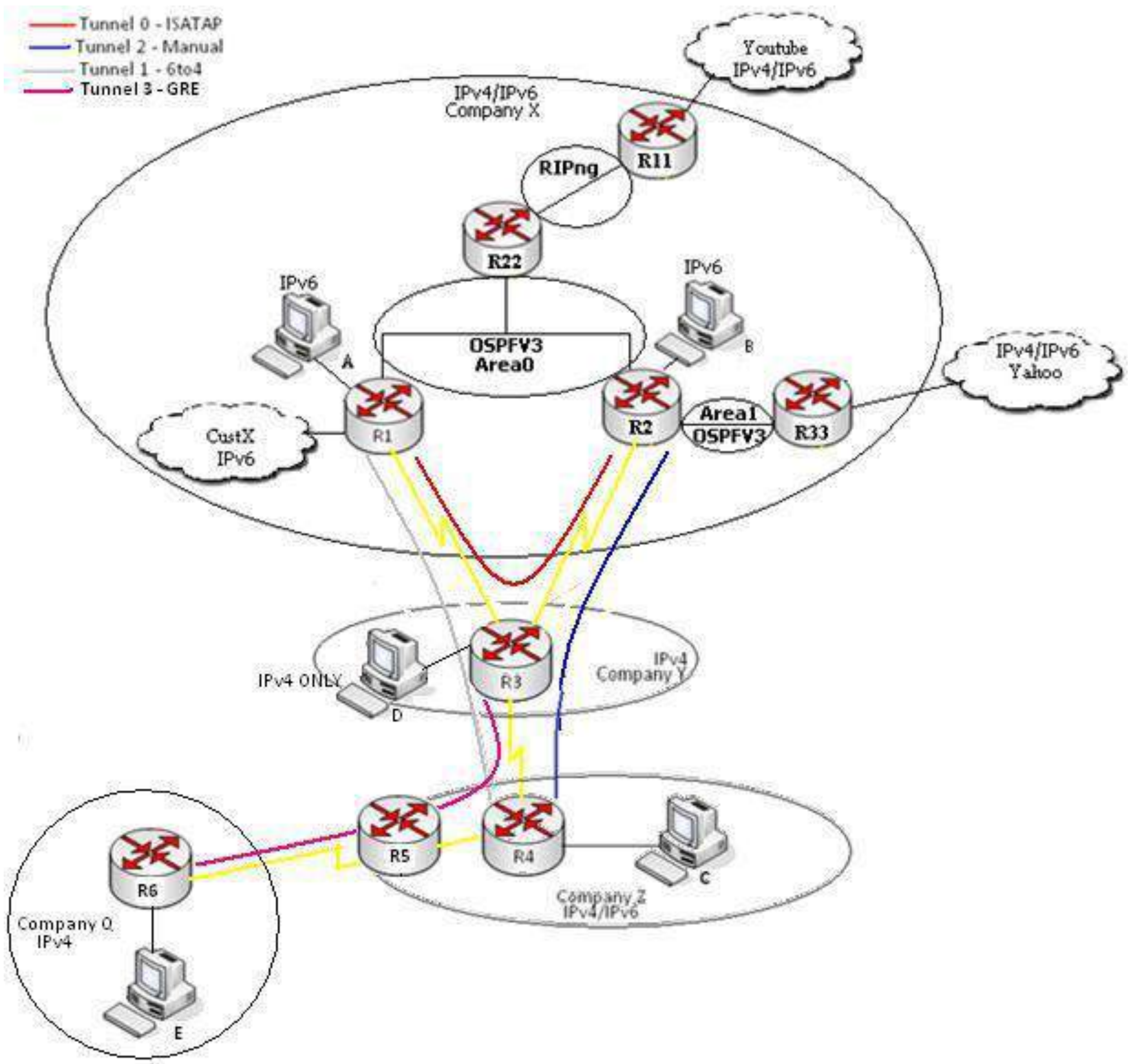
a) Use NAT-PT (decide which router you will make the NAT-PT router) HINT: only routers having dual stack can run NAT-PT.
b) Use a NAT prefix of 2050::/96
c) Any intra devices within company Y should take IP address from pool 10.0.10.0/24 iv) PC C should be identified by 50.0.0.1 IPv4 address. (But, PC C is still only IPv6 compatible)

- Finally make sure PC C can connect to PC D
- Also, using NAT-PT, make sure that PC D can reach Youtube. (both v4 and v6 connectivity).
- Check finally that Youtube can be reached by a v4 client (in company Y) and v6 client (in company Z).
- HINT: Run EIGRP for IPv4 connectivity in Company X

Extra Credit 1(important for final)

Get full connectivity (v4-v4-v4-v6,v6-v6) between all the devices and loopbacks using NAT-PT and IPv6/IPv4 routing protocols.
Intra-network IPv4 routing protocols - Company X - EIGRP, Company Y– RIPv2. For other tunneling purposes (e.i inter IPv4 routing protocol) use what you have been using before, i.e. RIPv2

Objective 5:IPv4 over IPv6 tunneling – GRE



Great, now that company Y is making a lot of money, it decides to connect to company Q which is just like company Y being only IPv4 compatible. So Company Z (who is getting paid by Company Y and Q for using its resources) decides to run a GRE tunnel between R3 and R6 through company Z (which is IPv6 compatible), particularly via R4 and R5. Follow the conditions to achieve connectivity between PC E and PC D (i.e to achieve IPv4 to IPv4 connectivity via IPv6):

- Link between R4 and R5 has only IPv6 addresses. (Run OSPFv3 between them)
- R4 and R5 being dual stack are connected to R3 and R6 respectively using only
- IPv4 addresses. (RIPv2 is running between R5-R6 and R3-R4)
- Company Q is running only IPv4 RIPv2 routing protocol.
- Again, PC E and PC D have only IPv4 addresses from their respective pools.
- Company Y has a pool of 10.0.0.0/8 and Company Q has a pool of 20.0.0.0/8
- Use GRE tunnel to get connectivity between PC E and PC D

Extra credit 3: (important for final)

Get full connectivity (v4-v4,v4-v6,v6-v6) between all the devices and loopbacks using NAT-PT and IPv6/IPv4 routing protocols.

Useful links for LAB troubleshooting purpose:

All required configurations:
http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-tunnel.html

ISATAP – http://blog.ine.com/2009/10/17/ipv6-transition-mechanisms-part-4-isatap-tunnels/

6to4 tunnel – http://blog.ine.com/2009/09/09/ipv6-transition-mechanisms-part-3-6to4-tunnels/ http://packetlife.net/blog/2010/mar/15/6to4-ipv6-tunneling/ http://ieoc.com/forums/t/5419.aspx http://cciethebeginning.wordpress.com/2009/07/24/automatic-6to4-transition/

NAT-PT –
http://blog.ine.com/2008/04/18/understanding-ipv6-nat-pt/

Useful Commands:

How to configure a static IPv6 address on Windows XP:

PC:

Go to command prompt and type netsh netsh> interface ipv6 install netsh>interface ipv6

netsh interface ipv6>add address “local area connection” 2000::1 netsh interface ipv6>show address

netsh interface ipv6>delete address *{interface no}* 2000::1

How to configure a static IPv6 address on a Router:

Router:

```
Router(config)#ipv6 unicast-routing
Router(config)# interface Fa 1/0
Router(config-if)# ipv6 address 2000:1:1:1:1:1:1:1112/112
Router(config-if)# no shut
```

How to configure OSPFv3:

http://www.cisco.com/en/US/tech/tk872/technologies_configuration_example09186a0080b40d8a.sh tml

How to configure tunnels:

Refer “Useful links for LAB troubleshooting purpose”

How to configure NAT-PT:

Refer “Useful links for LAB troubleshooting purpose”

Other useful commands for troubleshooting purpose: show ipv6 route

show ipv6
interface brief
ping *ipv6 address*
show ipv6 ospf
neighbors show
interface tunnel *number*
show ipv6 interface tunnel *number*

LAB Questions:

All the answers should not exceed more than 5-6 lines.

1. **Explain how DNS and ICMP have changed in IPv6 compared to IPv4.**
2. **Is there any ARP in IPv6? – Yes/No! Explain**
3. **What do you mean by dual stack? Is it used in tunneling? Explain**
4. **What is the basic change in RIPng compared to RIPv2 (with respect to configuration and concepts)**
5. **How does DHCPv6 work? How is it different from DHCPv4?**
6. **What is NDP - Explain? What IPv6 addresses are used by NDP?**
7. **How will you ping to ipv6.google.com from a IPv4 compatible PC (take Windows XP). Give me the necessary commands.**
8. **Does ISATAP tunnel support OSPFv3? Why/Why Not? Explain**
9. **How is Teredo Tunneling better than 6to4? Explain the major difference between the two tunnels?**
10. **What is function of a Zone ID? Explain**