Tien Hoang, Pham

Implementing IPv6 into Existing IPv4 Network on Cisco Devices

Helsinki Metropolia University of Applied Sciences

Bachelor of Engineering

Information Technology

Thesis

12 January 2013

Helsinki Metropolia University of Applied Sciences

Author(s) Title Number of Pages Date	Tien Hoang, Pham Implementing IPv6 into Existing IPv4 Network on Cisco Devices 55 pages + 12 appendices 12 January 2013
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Network Computing
Instructor(s)	Erik Pätynen, Senior Lecturer

The context of this project was a target organization which was planning on deploying a new IPv6 network system on their current IPv4 resources. The system administrator would like to have a document that would provide general knowledge of the new addressing system including its features, resource requirements, implementation guidance, and possible network models.

The Internet world has been growing rapidly during recent years and the IPv4 addressing space resource is used out. An upgrade to the new IPv6 addressing system is becoming a vital requirement for organizations. Despite the fact that documentations about IPv6 can be easily found but there are still very few organizations have done the upgrade. This is caused by many factors such as costs for the upgrade (labour, resources, and time), interruption, devices that do not support IPv6 are still in use, as well as troubleshooting skills and knowledge for the new problems that might come with the new system.

For the above reasons, this project was carried out. Within this project, six Cisco Catalyst 3750 routers were used for building the testing network. The implementation process was done both on real devices and the virtual environment of GNS3. The testing network topology was built to create a sample that involved setting up a new environment at the target organization, allowing the existing IPv4 network to operate optimally, while also allowing the introduction of the new system of IPv6 network.

The project managed to achieve the goals of analysing the concept, new features, and the structure of IPv6 as well as comparing IPv6 with IPv4 to help understand the necessity of an upgrade. The testing outputs showed successful results. The resulting document will serve as a reference for implementation for system engineers and product engineers for the deployment of the IPv6 protocol on a production network.

Keywords	network, computing, IPv6

Abbreviations

AH	Authentication Header
APAC	Asia-Pacific
APNIC	Asia Pacific Network Information Centre
AfriNIC	African Network Information Center
BGP	Border Gateway Protocol
CAR	Committed Access Rate
CatOS	Catalyst Operating System
CIDR	Classless Inter-Domain Routing
CM	Contract Manufacturing
DHCP	Dynamic host Configuration Protocol
DHCPv6	Dynamic Host Configuration Protocol for IPv6
DNS	Domain Name Server
DUAL	Diffusing Update Algorithm
EIGRP	Enhanced Interior Gateway Routing Protocol
EMEA	Europe, the Middle East and Africa
ESP	Encapsulating Security Protocol
EU	European Union
FDDI	Fiber Distributed Data Interface
FHRP	First Hop Redundancy Protocol
GRE	Generic Routing Encapsulation
HQ	Head Quarters
HSRP	Hot Standby Router Protocol
ICMP	Internet Control Message Protocol
IETF	Internet Engineering Task Force
IOS	Internetwork Operating System
IP	Internet Protocol
IPNG	Internet Protocol Next Generation
IPsec	Internet Protocol Security
IPT	Internet Protocol Telephony
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ISP	Internet Service Provider
ISP	Internet Service Provider
IT	Information Technology
LAN	Local Area Network

LLQ	Low Latency Queuing
LTE	Long Term Evolution
MAC	Media Access Control
MLS	Multi-Layer Switching
MPLS PIP	Multi-Protocol Label Switching based Private IP
MPLS	Multi-Protocol Label Switching
NA	North America
NAT	Network Address Translation
OS	Operating System
PPL	People
PPP	Point-to-Point Protocol
QoS	Quality Of service
R&D	Research and Development
RIB	Router Information Base
RIR	Regional Internet Registry
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
ToS	Type of Service
UDP	User Datagram Protocol
VLAN	Virtual Local Area Network
VOIP	Voice over Internet
VPN	Virtual Private Network
WAN	Wide Area Network
WLAN	Wireless Local Area Network
WS	Work Station

Contents

1 Introduction 1

2 Theoretical Background 1

2.1 The Transition of The Internet World to IPv6 1
2.2 Limitation of IPv4 3
2.3 Managing IPv4 Addresses 3
2.4 IPv6 Architecture 4
2.4.1 Improvement in Addressing Space 4
2.4.2 Global IP Address Hierarchy 5
2.4.3 New Features in IPv6 6
2.5 Comparison of IPv4 Header and IPv6 Header 9
2.6 IPv6 Packet Format 12
2.6.1 Formatting 12
2.6.2 Using IPv6 to Access An URL 13
2.7 Different Types of IPv6 Addresses 13
2.7.1 Unicast Address 13
2.7.2 Multicast Address 14
2.7.3 Anycast Address 15
2.8 Special IPv6 Addresses 16
2.8.1 Unspecified Address 16
2.8.2 IPv4-Compatible IPv6 Address 16
2.8.3 IPv4-Mapped IPv6 17

3 Implementation of IPv6 on the Basis of the IPv4 Network 18

3.1 Current Global Situation of IPv6 Implementation 18
3.2 Methods to Implement Internet Protocol Address Version 6 (IPv6) 19
3.2.1 Extended Unique Identifier-64 (EUI-64) Format 19
3.2.2 Stateless Auto-configuration 20
3.2.3 Dynamic Host Configuration Protocol Version 6 (DHCPv6) 22
3.3 Mobile IPv6 23
3.4 Routing for IPv6 24
3.4.1 Routing Table for IPv6 24
3.4.2 Static Routing 26
3.4.3 Dynamic Routing Protocols in IPv6 28
3.5 Open Shortest Path First Version 3 (OSPFv3) for IPv6 30

3.5.1 Function of Open Shortest Path First Version 3 (OSPFv3) 31
3.5.2 Link-State Advertisement (LSA) Packets of IPv6 32
3.6 Transition Mechanism Between IPv4 and IPv6 34
3.6.1 Dual Stack 35
3.6.2 Tunnelling 35
3.6.3 Network Address Translation/Protocol Translation (NAT-PT) 36
3.7 Network Models 36
3.7.1 Dual Stack Model 36
3.7.2 Hybrid Model 36
3.7.3 Service Block Model 37

4 Simulating the IPv6 Network 37

4.1 GNS installation and configuration 37
4.2 Testing Network 38

5 Implementation Results 41

6 Project Evaluation and Conclusion 45

References 46

Appendices

Appendix 1. Configuration of R1

Appendix 2.Configuration of R2

Appendix 3.Configuration of R3

Appendix 4.Configuration of R4

Appendix 5.Configuration of R5

Appendix 6.Configuration of R6

1 Introduction

Internet Protocol version 6 - IPv6 is the next generation of network addressing methodology, with vast addressing space, and will replace the old addressing system of IPv4 within the near future. Demand for deployment of IPv6 is thus gradually increasing and is becoming a mandatory task, especially for growing organizations with future expansions.

The target organization is planning to upgrade their campus network, which is running on IPv4, to IPv6. The system administrator would like to know whether it is possible to implement a new addressing system, without having any interruption to network, upgrades to their current resources if needed, evaluation of the advantages and disadvantages of the new IPv6 system, possible network models and a demonstration of the implementation process of a practical sample network.

The objective of this project is multi-fold. The project aims to evaluate new features supported in the new addressing system of IPv6, common methodology for upgrading from IPv4 to IPv6, resource requirements as well as sample configuration scripts. The project also demonstrates a practical sample of the implementation process of an IPv6 campus network, which is currently functioning on the base of existing IPv4 resources, presenting step-by-step of the configuration process in order to bring the new IPv6 system into production.

2 Theoretical Background

2.1 The Transition of the Internet World to IPv6

As Figure 1 shows, on 31 January 2011, the Internet Assigned Number Authority IANA assigned the last block of the IPv4 address for the Regional Internet Registries after 30 years since the release of IPv4. However, the IT world has been aware of this problem and has prepared IPv6 as a substitution since the early 2000s, which would be the base of the next generation of the modern Internet world. IPv6 addressing is an update of IPv4

to overcome the limitations of IPv4 and also aims to provide the Internet world with more advanced features which are and will be needed. [1.]

In 1973, TCP/IP was first introduced and was applied to ARPANET. During this time, ARPANET had around 250 inter connected sites, with approximately 750 workstations being unaware of the rapid growth of the Internet that would cause exhaustion in addressing later. According to collected statistical information, the modern Internet is connecting hundreds of thousands of sites and hundreds of millions of workstations globally. The numbers continue to grow and will require significant improvements via upgrades on the existing technologies and resources. [2.]

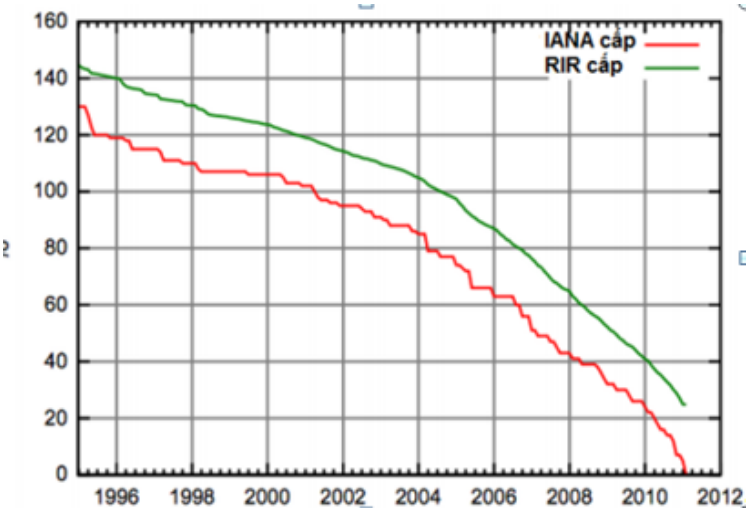

Figure 1. The exhaustion of IPv4 addresses over years (reprinted from [21]).

During the first few years of the 21st century, Internet applications started to provide services on new devices to users: notebooks, cellular modems, and tablets, smart phones, and smart TVs. To bring these new conceptual products to life, the addressing system TCP/IP should extend. IPv6 will not substitute IPv4 addressing instantly. Being an entirely new version of IP technology, researching and application of the new technology is very challenging. Among the challenges are the capability of IPv6 to be able to interact with neighbouring networks still running on IPv4, and how users are able to experience IPv6 advantages without having to upgrade the whole system (LAN, WAN, Internet, etc.) to IPv6. [2.]

2.2 Limitations of IPv4

IPv4 was built to support the 32 bits addressing method. IPv4 nowadays can no longer handle the desire of using the Internet. There are two major problems that IPv4 is facing: the exhaustion of addressing blocks (especially from the class B), and the dangerous growth of the size of the routing table on the Internet [3, 9].

Moreover, as technology grows, demand for the automatic configuration feature becomes more important. IPv4 addressing is divided into five layers: A, B, C, D, E. The first three layers are most widely used. The layers differ from each other in number of bits that are used to determine network ID [3, 9]. For example:

Addresses of the B layer have the first 16 bits assigned to determine the Network ID and the last 16 bits are used for determining the Host ID. The C layer addresses have 21 bits to determine the Network ID and the other 8 bits are used for determining the Host ID, etc. As the matter of fact, the capacity of these addresses from different layers is different.

2.3 Managing IPv4 Addresses

Besides those problems that were mentioned in section 2.2, the addressing technique has another limitation, the lost or lack of optimization in the usage of addresses. However the number of IPv4 addresses available can meet the requirements of the Internet but the delivery of IPv4 cannot. [4.]. For example:

A company requires having IP addresses assigned to 300 Hosts. To assign an IPv4 block of address to this company, layer B addresses are assigned to them. However, a block of layer B address can be used for 65536 Hosts. Using B layer block of address for this company would waste 65000 addresses. Other companies or organization would not be able to use these left addresses. [4.]

During the 1990s, Classless Inter-Domain Routing (CIDR) was built based on the understanding of addressing a mask. CIDR temporarily solved the above mentioned problems. Hierarchical of CIDR was an improved extension in IPv4. This method helps the delivery of IPv4 become more flexible with the help of a subnet mask. The length of Network ID

to the Host ID depends on the number of bit 1 of the subnet mask, and as a result, the capacity of IP address increases. [4.]. For example:

Using layer C IP addresses with the length of Subnet Mask 23, e.g. x.x.x.x/23 for the previous company. This address has Host ID determined by 9 bits, equals to 512 Hosts. This address is suitable, however CIDR has a cons is Router can only detect Network ID and Host ID if it knows Subnet mask value.

Although there were more technologies created, such as the technique of subnetting (1985), VLSM (1987) and CIDR (1993), but those could not save IPv4 from a simple problem: there would not be enough addresses for everyone in the future. There are around four billion IPv4 addresses but still not enough for the future, once devices that will be connected to the Internet and other household electrical appliances start to need IP addresses. [5.]

A few short-term solutions have been suggested, for instance RFC 1918 (Address Allocation for Private Internets), in which the space of an address is made for dedicated addresses, and NAT is a tool that allows thousands of Hosts connected to the Internet with a few valid IP addresses. However, a long-term solution is to bring in an IPv6 address with 128-bit address structure. The greater addressing space of IPv6 not only provides more availability but also other advanced new features. [5.]

With 128 bits, 340 282 366 920 938 463 374 607 431 768 211 456 addresses are available. In 1994, IETF proposed IPv6 in RFC 1752 (The Recommendation for the Ip Next Generation Protocol). IPv6 helps improve the situation in various fields such as address space exhaustion, quality of service, auto configuration, verification, and security. [5.]

2.4 IPv6 Architecture

2.4.1 Improvement in Addressing Space

During the development of the new addressing version, IPv6 was built entirely on the base of IPv4. Some old features were left out and replaced by new better features. [6, 131.]

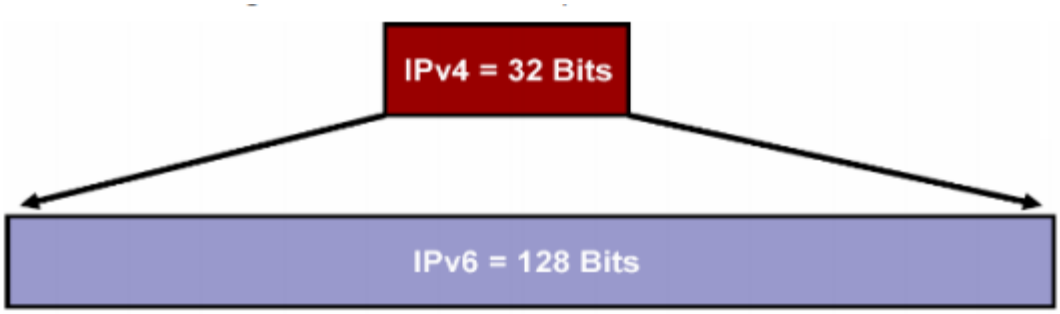

Figure 2. Number of bits in IPv4 compared to IPv6 (reprinted from [8]).

As can be seen from Figure 2, IPv6 uses 128 bits for addressing, which is four times more than IPv4 (32 bits). This means that while IPv4 has 2^{32} ~ 4.3 billion addresses, IPv6 can have 2^{128} ~ $3.4*10^{38}$ IP addresses, 2^{96} times more than IPv4. With the number of IP address IPv6 can provide, if delivered equally, 665 570 IPv6 addresses are available to each square meter on the Earth surface. [6, 131.]

IPv4 addressing is currently maintained by the NAT technology and a temporary address delivering method. The downsides of this are the peer-to-peer exchange, end-to-end security, and Quality of Service (QoS). With a vast number of available addresses in IPv6, these technologies would not be needed anymore. Each device can have a global IP address. The current demand would only consume 15% of the available IPv6 addresses, and the rest 85% would be saved for future extension. [6, 131.]

2.4.2 Global IP Address Hierarchy

The previous address hierarchy is illustrated in Figure 3.

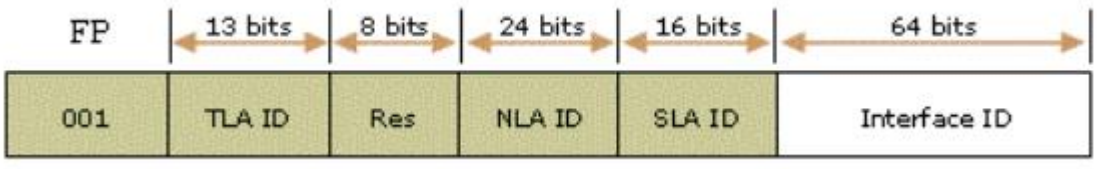

Figure 3. Previous IPv6 address hierarchy (reprinted from [7]).

As shown in the Figure 3:

- FP – Format Prefix: 3 bit 001 to recognize global IP address.
- TLA ID – Top Level Aggregate ID: Highest level of mass recognition.
- Res – Reserved: Reserved for future extension.
- NLA ID – Next level Aggregator ID: Mass recognition for the next level.
- SLA ID – Site Level Aggregator ID: Area mass recognition.
- Interface ID: interface naming address of a node in a child network. [7.]

The current address hierarchy is illustrated in Figure 4.

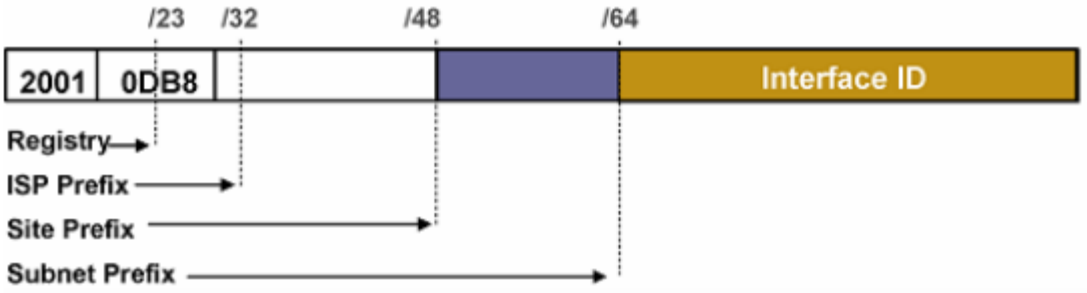

Figure 4. The current address hierarchy

Table 1. Global IPv6 delivery (data collected from [7])

Prefix	Number of bit	Features
/3	3 bits	Is always 001, and used for globally routable unicast – GRU
/23	20 bits	The highest level is IANA, IANA delivers to the next 5 RIR – the highest area IP address delivery, including: AfriNIC (Africa), ARIN (North America and Caribbean), APNIC (Asia-Pacific), RIPE (Europe, the Middle East and Middle Asia).
/32	9 bits	Area level or national level, is delivered by RIR to the highest level ISP in the system of service provider of each nation.
/48	16 bits	Regional service provider or big organizations.
/64	16 bits	Lowest level, is delivered by ISPs to customers.

The current IPv6 address is delivered by the IANA. Table 1 describes how IPv6 is delivered based on prefix.

2.4.3 New Features in IPv6

Simplified addressing process for Host means that IPv6 uses the last 64 bits for Host addressing utilizing a technology that is known as EUI-64, to simplify the Host addressing

process compared to IPv4. This technology utilizes 48 bits of the MAC address for the Host address, inserts FFFE into each 16-bit of the Mac address to complete 64 bits of the Host address. In this way, every Host will have only one Host ID within the network [9, 43-44].

Auto-configuration is used to simplify workstation configuration. IPv6 supports both Stateful Auto-configuration and Stateless Auto-configuration. With the Stateless Auto-Configuration, the workstations can automatically connect to the router and receive prefix for the network. Without the Router, the workstations connected to each other still can auto-configure and interact with each other without any assistance [9, 62].

Higher efficiency is achieved with IPv6 that uses private addresses to avoid addressing. NAT technology was invented to interchange addresses, leading to the increase in Overhead for data packages. In IPv6, due to the vast number of available addresses, the NAT can be left out. The performance is improved significantly because the Header processing time has been shortened by decreasing the number of Overhead. [9, 230].

IPv6 decreases the time to process routing: Many IPv4 address blocks are delivered but cannot be summarized, so they require entries into the routing tables to improve their sizes and implementation of Overheads. In contrast, IPv6 is delivered through ISP to decrease the number of Overhead [9, 231].

In IPv4, it is common to use many Broadcasts such as ARP Request, while IPv6 uses Neighbour Discovery Protocol to execute similar features such as in the auto configuration without having to use Broadcast. Besides, Multicast has limits in IPv6. A Multicast address contains a scope which can limit Multicast data packets within nodes, links, or within an organization [9, 230].

Mobility support is a very important feature of modern network systems. The mobile IP is a standard of IETF for both IPv4 and IPv6. Mobile IP allows devices to change their locations without being disconnected, and maintains the current connection. In IPv4, a mobile IP is a new feature that should be integrated in case it is needed. On the other hand, IPv6 has a mobility feature integrated, which means any IPv6 node can be used when needed. Figure 5 is a graphical demonstration of the mobility feature of IPv6 [9, 404].

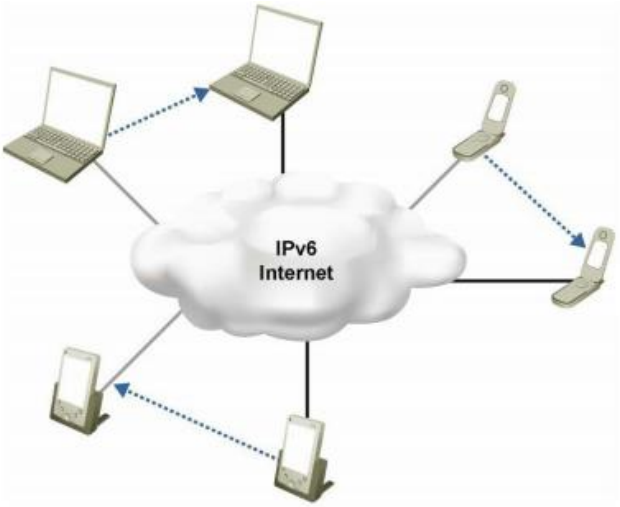

Figure 5. Mobility features of IPv6.

In addition, the routing headers of IPv6 help a mobile IPv6 to function with more efficient Mobile IPv4. In the future, portable devices such as laptops, tablets, and smartphones. will be using IPv6 integrated on the basis of telecommunication network resources [9, 408].

IPSec feature (IP Security) is a standard proposed by IETF for the field of security in IP networks, and is used in both IPv4 and IPv6. The basic features are similar within both environment. However, in IPv6, IPSec is a mandatory feature. IPSec is integrated and active on all IPv6 interfaces. The readiness of IPSec on all of the nodes help making IPv6 Internet more secured [9, 408].

Header of IPv6 was made to be simpler and more logical than that of IPv4. IPv6 has only six fields and two addresses. As a result, it takes less time for IPv6 data packets to be sent and received within the network, which helps to improve the connection speed. [4.]

Addressing aggregation is a technique similar to Address Summarize in IPv4. An ISP will summarize all of the prefixes of the customers into only one single prefix and announce this prefix to its higher level. [4.]

The process of summarizing addresses will help to shorten and simplify the routing table and provide possibility to increase routing on all of the routers. This will lead to the possibility to optimize and increase bandwidth, allowing other kinds of network services such as VoIP, Internet TV, high definition videos, real-time applications, online games, and

studying or conferencing over the Internet. Figure 6 briefly demonstrates the process of summarizing addresses. [4.]

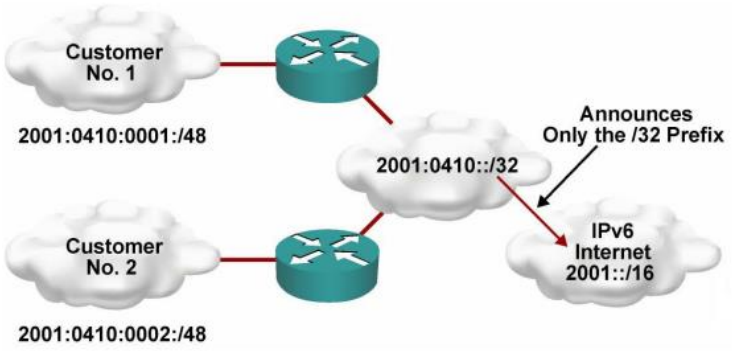

Figure 6. Summarization for the routing (reprinted from [17]).

Renumbering of IPv4 devices is a task that systems administrators do not want to deal with. It affects the functionality of the whole network and sometimes costs a vast amount of labour to reconfigure the information for the connected devices. [4.] IPv6 addresses are designed to possess a method for renumbering with ease. An IPv6 assigned to node will have two states: “preferred” and “deprecated” depending on the active time of that address. The computer always tries to utilize addresses in the “preferred” state. The living time of the address is configured from the broadcasted information of the routers, so computers running IPv6 can be renumbered by the announcement of routers for the expiring time for the use of a prefix. [4.]

2.5 Comparison of IPv4 Header and IPv6 Header

Header of IPv6 has 40 octets (40 bytes) while there are only 20 octets in IPv4. However IPv6 has fewer fields, so it improves the header processing time and flexibility. The addressing field is four times greater than in IPv4. [4, Chapter VI.]

The header checksum of IPv4 is left out because the current connection speed is higher and are more reliable and thus it only requires hosts to checksum while routers do not have to. Besides, the header checksum is a parameter which is used for checking errors within the header. It is calculated on the basis of numbers of the header. However, there is TTL (Time to Live), which value needs to be changed when a data packet is transferred over a router. So the header checksum needs to be recalculated. If a router is released from this task, the delay can be decreased. [4.]

In IPv4, when packets grow big, routers can separate them. This will increase the number of overhead for packets. In IPv6, only the source hosts can separate a packet according to a suitable MTU path that it finds. In order to support hosts, IPv6 includes a function to help finding an MTU from the source to its destination. [4.]

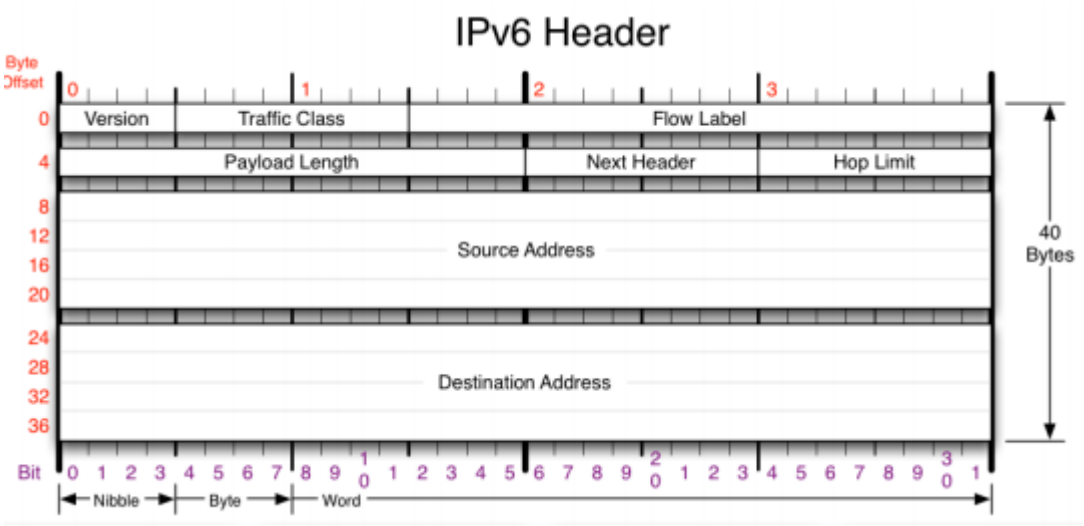

Figure 7. Details of IPv6 Header (reprinted from [22]).

As shown in Figure 7, the IPv6 header includes the *Version field* which contains 4 bit 0110 corresponding to the 6 numbers of IP version. The Traffic class: contains 8 bits corresponding to the Type of Service (ToS) in IPv4. This field is used for demonstrating the priority of packets, for instance whether or not to deliver a packet with high priority or normal priority, allowing devices to process packets correspondingly. The *Flow Label* contains a complete new field in IPv6, and contains 20 bits. This field performs the flow of data packets and is used in multilayer switching, so data packets can be switched faster than previously. By using this field, the source of the data packet and the current device can track out a stream of data, for instance VoIP. In IPv4, some interactive devices can also detect the flowing stream of data and attach a priority tag. However, these devices do not only check information from the content of the IP level such as source and destination but also port number and other information belonging to the higher level. Flow Label IPv6 aims to combine the important information and provide it as the IP level. The *Payload Length* contains 16 bits. Similarly to the Total Length in IPv4, determines total size of IPv6 packets (excluding header). The Next Header field contains 8 bits. This field will determine whether the extension header exists. If it is not in use, the basic header contains all of the IP level information. It will be followed by the header of the

higher level header, which is header of TCP or UDP, and the next header field will point out the type of the following header. The *Hop Limit* field contains 8 bits. This field is similar to the Time to Live field of IPv4. It carries information of the maximum number of hops that an IP packet is allowed to encounter. The *Source Address* field contains 16 octets (128 bits), carry information of the source address of the packet. The *Destination Address* field contains 16 octets (128 bits), carry the destination address of the packet. [4.]

Last but not least, the IPv6 Header also contains the Extension Header, which is an additional header, as shown in Figure 8. IPv6 applies a separate system of the additional services and locate them in an extension header, and categorizes the extension headers correspondingly to their features. This will increase flexibility and improve the functionality of routers. [4.]

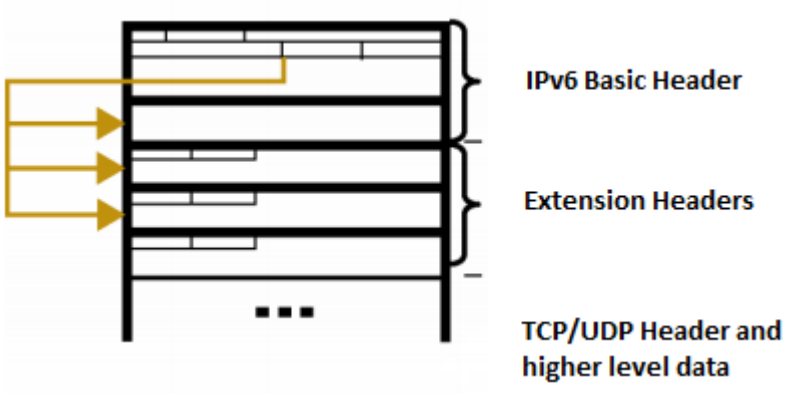

Figure 8. Order of headers in IPv6 data packet

There are six kinds of Extension headers. When there are multiple Extension Headers in use, there is usually a suggestion to arrange them in the following order: the *Hop-by-Hop Options Header* (value = 0) defines a process that needs to be done every time a packet travels through a router. The *Destination Options Header* (value = 60) is used in case there is a Routing Header, in order to determine a process that needs to be done by the destination node. It is possible to determine from here any process. Usually only the destination node processes the Extension Header of IPv6. So other Extension Headers for example the Fragment header can be called Destination Option Header. However, the Destination Option header is different from other headers because it can define many type of processes. Mobile IP uses this Header frequently. The *Routing Header*

(value = 43) is used to determine the Routing Path, for example: In order to determine which Service Provider will be used, and security options, the source node uses Routing header to list the address of the Routers so that a data packet would go through. The *Fragment Header* is used when the source sends IPv6 data packet greater than Path MTU, to guide how to recover the data packet from its fragments. The MTU (Maximum Transmission Unit) is the size of the biggest data packet that can be sent over a specific path. The *Authentication Header (value = 51) and Encapsulating Security Payload Header (value = 50)* are used in IPSec to verify the wholeness and security of a data packet and it is used for determining the information concerning data encryption. The *Upper-Layer Header* is considered the header that defines fields on the IP level and defines the method to transfer packets. Two main protocols are TCP (value = 6) and UDP (value = 17). [4.]

2.6 IPv6 Packet Format

2.6.1 Formatting

128 bits of IPv6, are divided into 8 octets, each octet needs 2 bytes (4 bits), includes 4 hexadecimal values, each group is separated from the others by the colons.

IPv6 is a new addressing system, most of the 128 bits are still not used, so there are many zeros for the first bits, and it is possible to leave out these zeros for simplification. Thus it is common to encounter an IPv6 address as:

1088:0000:0000:0000:0008:0800:200C:463A

In order to simplify the IP address, 0000 is replaced by 0, 0008 is replaced by 8, 0800 is replaced by 800. The simplified address would look like:

1088:0:0:0:8:800:200C:463A.

Another possibility when presenting an IPv6 address is to group one or more blocks of zeroes and represent them with double **"::"**. However, within one IPv6 address, the syntax allows to have only one double colon. The previous sampled IPv6 address can thus be shortened as follows:

1088::8:800:200C:463A.

2.6.2 Using IPv6 to Access An URL

It is possible to access a website by the domain name or an IP address. For example, instead of typing www.metropolia.fi, it is possible to type 195.148.144.10.

Similarly, it is possible to access a website by its IPv6 address, but it must be placed within square brackets **[ ]**. For example:

[1088::8:800:200C:463A]

2.7 Different Types of IPv6 Addresses

2.7.1 Unicast Address

Global Unicast Address is provided by ISPs to users. Global Unicast Addresses is similar to the public address of IPv4. The structure of this type of address was explained previously in the Section 1.4.2.

Link-local Address is the address used for Hosts when there is demand to interact with other Hosts within the LAN. All IPv6 interfaces within the LAN network have link-local addresses [9, 44].

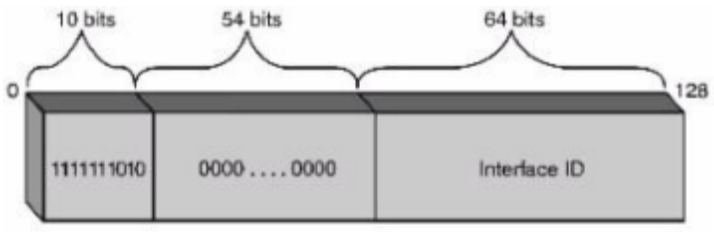

Figure 9. Structure of the Link-local address.

According to Figure 9, the first 10 bits are unchangeable values: 1111 1110 10 (Prefix FE80::/10), the following 54 bits contain value 0, the last 64 bits are interface addresses.

```
type ?.

:\Users\NTLong>netsh interface ipv6 show addresses

Interface 1: Loopback Pseudo-Interface 1

ddr Type  DAD State   Valid Life Pref. Life Address
-------- ---------- ---------- ---------- ------------------------
)ther     Preferred     infinite   infinite ::1

Interface 10: Wireless Network Connection

ddr Type  DAD State   Valid Life Pref. Life Address
-------- ---------- ---------- ---------- ------------------------
)ther     Preferred     infinite   infinite fe80::d16d:c70d:e6a9:9775%10

Interface 16: Teredo Tunneling Pseudo-Interface

ddr Type  DAD State   Valid Life Pref. Life Address
-------- ---------- ---------- ---------- ------------------------
)ther     Deprecated    infinite   infinite fe80::100:7f:fffe%16

:\Users\NTLong>
```

Figure 10. Checking link-local address

Figure 10 demonstrates the method to check the link-local address of the workstation. Routers cannot transfer any packet that has either source or destination address as a link-local address.

2.7.2 Multicast Address

In IPv6, there is no broadcast address. The features that were previously integrated into IPv4 are replaced by IPv6 Multicast.

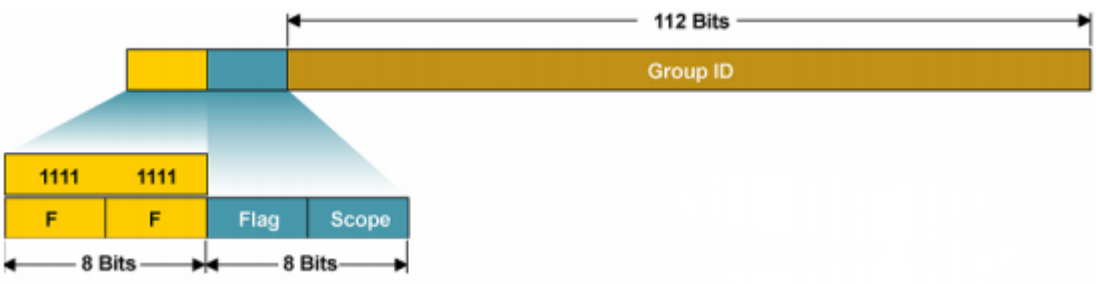

Figure 11. Structure of multicast address [8].

According to Figure 12:

The IPv6 Multicast address is defined with prefix of FF::/8

From FF00:: to FF0F:: are the dedicated addresses defined by IANA used for multicast purposes.

The second Octet carries the flag and scope of the Multicast address.

- The flag value defines the life span of the address. The flag receives either 0 (permanent) or 1 (temporary) value.
- The scope defines the range of the address. Scope receives seven possible values: Interface-local, Link-local, Subnet-local, Admin-local, Site-local, Organization, Global. [9, 47.]

Table 2 gives some examples of the Multicast address.

Table 2. Example of IPv6 Multicast

Address	Category	Range
FF02::/16	permanent	Link-local
FF08::/16	permanent	Organization
FF14::/16	temporary	Admin-local
FF1E::16	temporary	Global

Table 3 describes different types of Multicast addresses in IPv6.

Table 3. Description of different types of IPv6 Multicast addresses(data collected from [25, 15-16])

Address	Last bits	Subject	Range
FF02::1	1	Every nodes	Link-local
FF03::2	2	Every routers	Subnet-local
FF04::9	9	Every RIP routers	Admin-local
FF02::1:FFXX:XXXX	FFXX:XXXX	Solicited-nodes	Link-local
FF05:101	101	Every NTP servers	Site-local

FF02::1:FFXX:XXXX is a type of Multicast address with the role of Solicited-node (replacing ARP in IPv4) in order to derive IPv6 to the MAC address of the nodes within an area (the area in this example is link-local).

2.7.3 Anycast Address

Anycast is a new kind of address in IPv6, also known as one-to-nearest.

Figure 12. Structure of anycast address (reprinted from [8]).

As Figure 12 illustrates, Anycast address is a Global Unicast address assigned to many interfaces of many different Routers within a WAN scope. Data packet forwarded to Anycast Address will be transferred by the routing system to a router that has the best metric/closest router. At the moment, Anycast address has limited use. There are very few documents about the use of this kind of address. Anycast Address is mostly used for Router, for the purpose of balancing load. For example: When a service provider has many customers who want to connect from different geographical locations, in order to save costs, the service provider should assign a single Server to serve all. They build many routers to connect customers with the central Server. The service provider assigns anycast address for the Routers connected to the central Servers, so each customer only needs to memorize and connect to one single Anycast address. They will be automatically connected to the Server through the closest router. An Anycast address is never used as the source address of a data packet. [9, 100.]

2.8 Special IPv6 Addresses

2.8.1 Unspecified Address

IPv6 uses the following special addresses:

0:0:0:0:0:0:0:0 - simplified as "::" is an unspecified address that is used by IPv6 node to indicate that it does not have an address. The address "::" is used as a source address for data packet during the working process of an IPv6 node. This address is never assigned to an interface or used as the destination address.

0:0:0:0:0:0:0:1 - or "::1" is used as the address to determine loopback interface, corresponding to the address of 127.0.0.0 of IPv4. This address is used for checking whether a workstation can work with IPv6. Besides, to routers, an address of "::1" is never sent on the same connection path. This address is used within a node. [9, 203.]

2.8.2 IPv4-Compatible IPv6 Address

IPv4-Compaitble IPv6 address is a compatible address of an IPv4 with an IPv6 node. When using IPv4-compatible as an IPv6 destination address, the data packet will be packed together with an IPv4 header to be transferred within the IPv4 environment. [11, 267.]

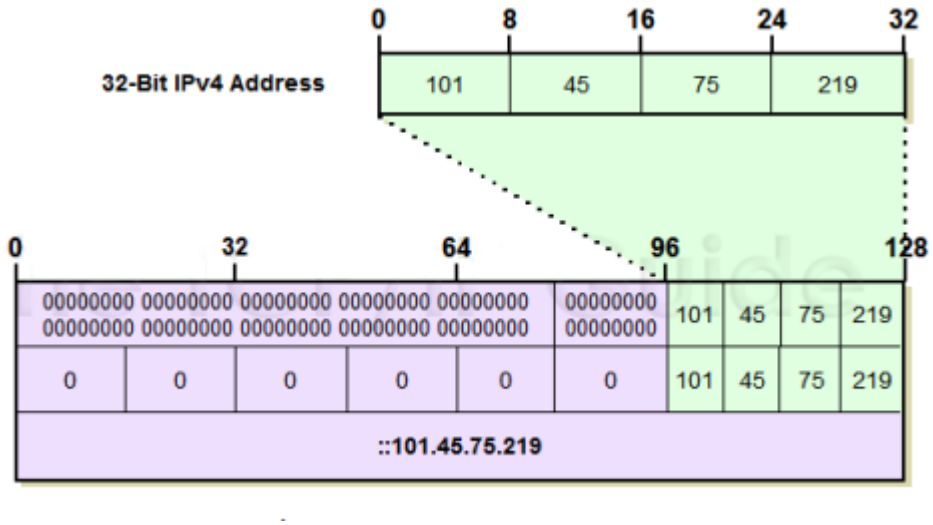

Figure 13. Structure of IPv4-Compatible IPv6 address [20]

As seen in Figure 12, the format of IPv4-Compatible IPv6 address is: 0:0:0:0:0:0:w.x.y.z, whereas w,x,y,z are IPv4 addresses. The address type IPv4-Compatible is used within the technology of creating an automatic tunnel. When an IPv6 data packet has the source address of this type, the IPv6 packet will be included a packet with an IPv4 header and sent to a destination environment that uses IPv4 addressing. [11, 267.]

2.8.3 IPv4-Mapped IPv6

IPv4-Mapped IPv6 is constructed from 32 bits of IPv4 address using the method of extending the first 80 bits, and the following are 16 bits of hexadecimal value FFFF with 32 bits of IPv4 address. The IPv4-Mapped address is used to show IPv4 address to an IPv6 node to serve the address translation technology from IPv4 to IPv6, for instance NAT-PT technology. IPv4-Mapped address is never used as the source or destination address of an IPv6 data packet. [9, 568.]

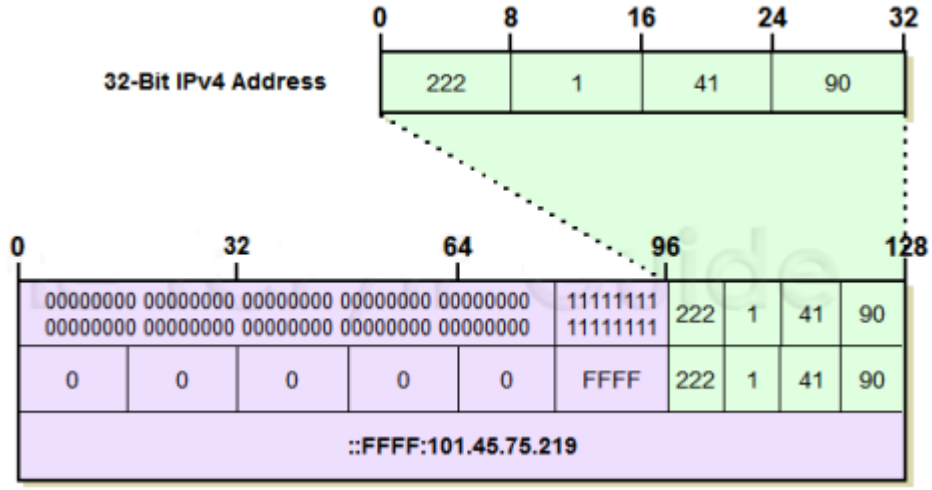

Figure 14. Structure of IPv4-Mapped IPv6 (reprinted from [20]).

Figure 15 shows the structure of IPv4-Mapped IPv6 address which has the format of 0:0:0:0:0:FFFF:w.x.y.z.

3 Implementation of IPv6 on the Basis of the IPv4 Network

3.1 Current Global Situation of IPv6 Implementation

In Asia, the limitations of IPv4 has laid a certain obstacle to the development of the Internet for the important economic areas such as China, Taiwan, Japan, and South Korea. These nations have determined that IPv6 is the next-generation and promising technology. The work of developing IPv6 and leading the next generation of network computing technology is carefully planned. China plans to build the greatest IPv6 network all over the world. [12.]

Planning of deployment of IPv6 is being rapidly developed by big researching projects, trying to build IPv6 networks to connect the European nations, connecting Europe with other continents. The US where the Internet originates from, is also the nation that owns most of the space in IPv4. So the demand for addresses is not under such a rush. However, due to the great improvements in IPv6 in security, in the year 2008, the US Ministry of Defence decided to implement IPv6 for its network system. [12.]

3.2 Methods to Implement Internet Protocol Address Version 6 (IPv6)

3.2.1 Extended Unique Identifier-64 (EUI-64) Format

The 64-bit protocol in an IPv6 protocol is used to define a single interface in a link. A link is a network environment within which the nodes communicate by using the connecting layers (the second layer in the OSI hierarchy model – data link layer). Interface can define the singularity of itself on a larger scale. In many contexts, an interface is recognized by the connecting layer (the MAC address of the interface). Similarly in IPv4, a subnet prefix in IPv6 is related to a link. [13, 127.]

The Identified interface is used in global unicast and other kinds of IPv6 addresses and must have a length of 64 bits. It is built using a format created by IEEE which is the Extended Universal Identifier (EUI) - 64. EUI-64 format ID interface originates from 48 bits of the MAC address of the interface. Because the MAC address is unique, by adding a string of the hexadecimal FFFE in the middle of 3 bytes of the MAC address to create 64 bits of the interface ID. [13, 127.] This method can be seen in the Figure 14.

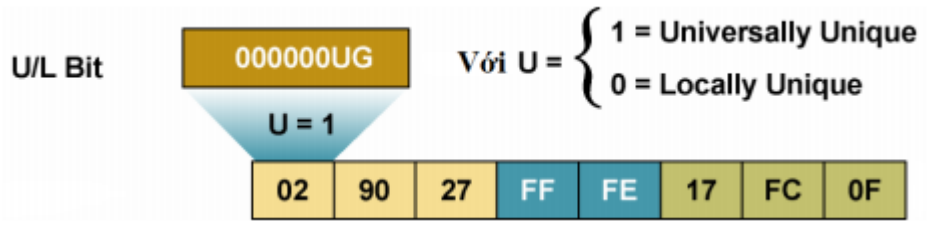

Figure 15. EUI-64 format for IPv6 (reprinted from [8]).

In order to make sure the address made from the MAC Ethernet address is unique, the seventh bit from the first octet (U bit) is either 1 or 0 stands for the unique value of the whole or the unique value locally, in order to organize the groups. [13, 57.]

3.2.2 Stateless Auto-Configuration

Definition

IPv6 is designed on the model of plug-and-play. Within a local area network, if the computers are connected to a Router then process called Stateless Auto-configuration is initiated. [13, 127.]

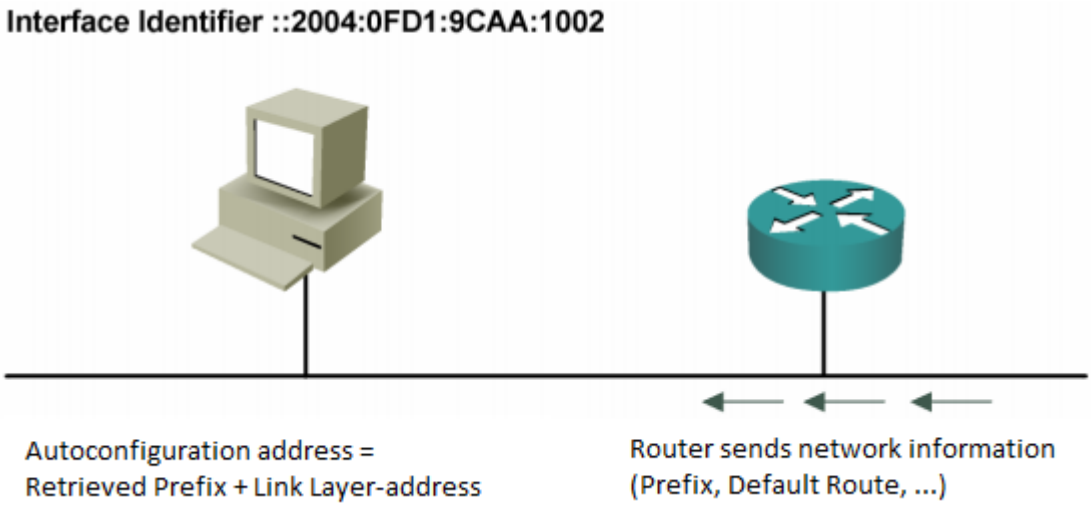

Figure 16. Stateless Auto-configuration [19]

As demonstrated in Figure 17, a router in a local area network sends information to its network, such as a 64-bit prefix of the network and the default route of the network. Router will send this information to all of the nodes within the network it is connected to. A computer can initiate auto-configuration by using the 64 bit prefix of the network that the Router sends together with the EUI-64 technology to construct the host 64 bits. This process leads to an unique 128-bit address that can be used globally. [13, 127.]

A process called *duplicated address translation* is triggered in case a duplicated address is detected. Auto-configuration helps making the plug-and-play feature optimized. In fact, devices are connected to the network without any configuration and do not require any server (DHCP). This feature allows connecting new devices to the Internet, such as mobile phones, wireless devices, household appliances, as well as family monitor devices. [13, 127.]

The Stateless Auto-configuration process is initiated through the following three steps:

Step 1: The device will send a data packet called router solicitation to the router to request network information like shown in Figure 18.

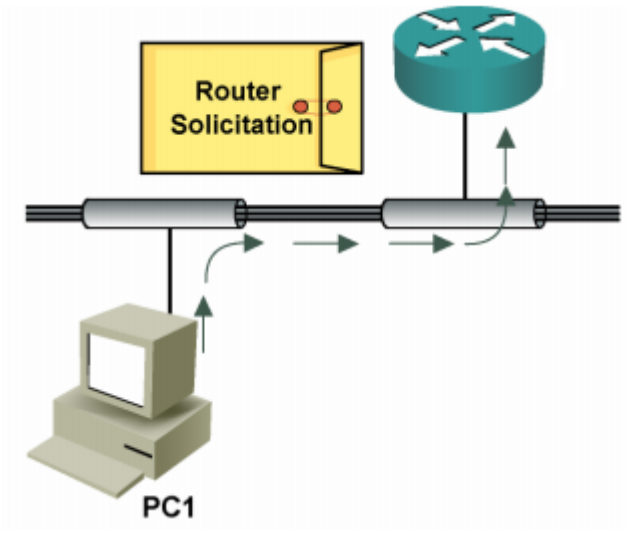

Figure 17. Step 1 of Stateless Auto-configuration (reprinted from [19]).

Step 2: The router replies with a router advertisement packet containing the necessary information (including the 64-bit prefix of the network and a default route).

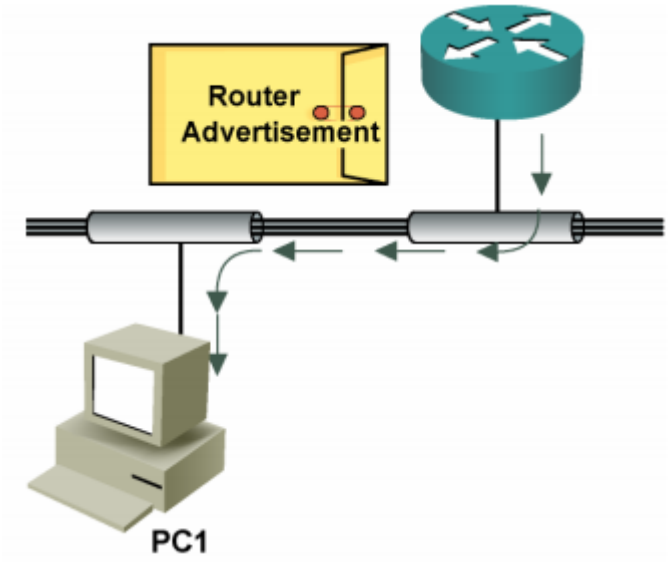

Figure 18. Step 3 of the Stateless Auto-configuration [19].

Step 3: The device uses 64-bit prefix of the network retrieved from the router together with the EUI-64 method to obtain a 64 bits host address. The result is the 128 bits of the IPv6 address. [13, 127.]

3.2.3 Dynamic Host Configuration Protocol Version 6 (DHCPv6)

During the process of Stateless Auto-configuration, each node is responsible for constructing its own address and saving the interface ID and its information using the neighbour discovery protocol. Within a small network, this process benefits from its simplicity and ease of use. However it also has some disadvantages. It is on the multicast technology, cannot use the range of address effectively and has a lack of security, and lack of control in policy and enrolment. [14, 150.]

In order to support the interaction between bigger scale networks and complex networks, it is better to use Stateful Auto-configuration. This concerns studies in Stateful Auto-discovery, DHCPv6, DHCPv6 client, and relay agent. [14, 150.]

Stateful Auto-Configuration is built based on the servers to provide information about configuration. These servers are known as DHCPv6 servers. However, from the point of view of system administrators. Stateful Auto-configuration is more complicated than Stateless Auto-configuration because it requires additional information of the DHCPv6 servers. Hence, Stateful Auto-configuration possesses a better possibility for future extension for large scale networks. [14, 150.]

Stateful Auto-configuration can be used simultaneously with Stateless Auto-configuration, i.e. a node can follow a stateless process during the starting process in order to obtain the local address. After that it can use Stateful Auto-configuration to obtain additional information from the DHCPv6 server. [14, 150.]

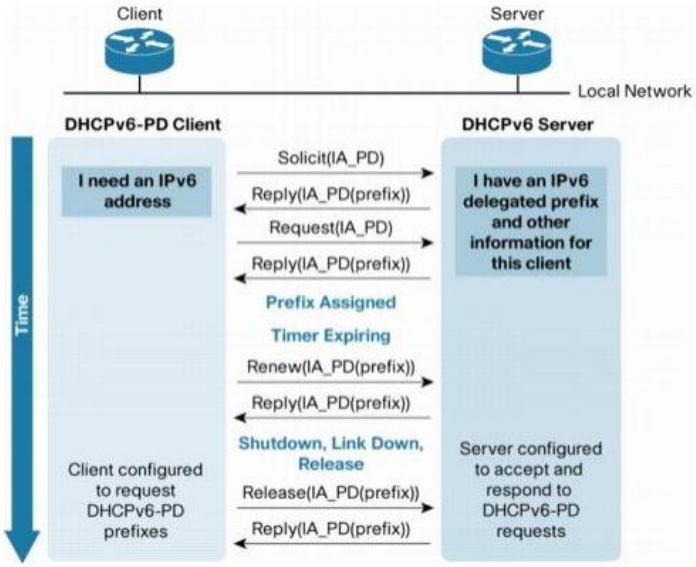

Figure 19. DHCPv6 process (reprinted from [23]).

In order to obtain configuration information, client needs to determine a DHCPv6 server by sending a DHCP solicit message by listening to a DHCP advertisement. The client later will send a unicast DHCPv6 Request. If the DHCPv6 server is not from the same subnet with the Client, then a DHCP relay or agent will forward the request to another servers. Server will reply with a DHCPv6 Reply containing configuration information for the client. [14, 151.] An illustration of this process is shown in Figure 18.

Using DHCPv6 brings many advantages. DHCPv6 helps controlling the delivery and assigning addresses from a central monitoring station. The delivery of addresses process is based on a hierarchy, so it is possible to summarize addresses. When a new ISP is chosen to replace the old one, then the new addresses can be easily delivered with the DHCPv6 service. A registered host can be used in a service of DHCPv6. The registration system can provide selective services to the registered Hosts and refuse to provide services to unregistered Hosts. [13, 165.]

3.3 Mobile IPv6

The Mobile IPv6 is a standard which allows IPv6 nodes to interchange from one network to another remaining connected. When an IPv6 node change its positions, its connection is possibly changed accordingly. When IPv6 changes its connection, the IPv6 address

can also be changed to retain the connection. However, once the address is changed, the connection might not be able to retain. [13, 165.]

An advantage of the Mobile IPv6 is even when the moving nodes change their places and addresses, the connection is preserved. Connections to the mobile nodes are usually approved. The Mobile IPv6 provides a connection for the nodes at the transport layer. [13, 165.]

3.4 Routing for IPv6

Similarly to IPv4 nodes, IPv6 nodes use a local IPv6 routing table to specify how a data packet is routed. The entries into the routing table are created with default settings when IPv6 is initiated. Other entries will be added to the table, when receiving router advertisement data packets containing prefixes and routes, or can also be manually configured. [24.]

3.4.1 Routing Table for IPv6

Routing tables in IPv6 will appear on all the nodes running IPv6 protocol. The routing table saves the information of the subnets of the network and a next hop in order to reach those subnets. Before the routing table is checked, the destination memory will be checked in order to find out the entries that match with the destination address in the IPv6 header of the data packet. Otherwise, the routing table will be used. [24.]

An Interface is used to deliver a data packet. The interface determines the physical or logical interface to be used to send the packet to its destination or the next hop router. The next hop address is the instant destination address of data packet. With the destinations that are not in the same subnet, the next-hop-address is the address of a router. After the interface and the next-hop-address are determined, the node will update its cache memory. The next data packets will be transferred to their destinations using cache memory without having to check the routing table. [24.]

Entries of an IPv6 routing table can be used to save the following routes:

- Direct connection routes which are prefixes of subnets that are directly connected and have the size of 64 bits.
- Remote routes which are prefixes of the indirect connected network. These routes are prefixes of a subnet (usually have a prefix of /64) or a prefix of a range of address (a prefix smaller than 64)
- Route of hosts: a host route is a route for a certain determined IPv6 address with prefix of 128 bits.
- Default route: Is used when a network cannot find the route from the routing table, with a prefix of ::/0. [24.]

Routing process: In order to define which entry to be used from the routing table, for each entry of the routing table. It will compare the bits within the network prefix with the corresponding bits from the destination address. The number of bits determined by the prefix of route. If they match then the route will be chosen for destination. Routes that match will be reprocessed. The route that has longest prefix length will be chosen according to the longest match rule. The longest match route will be the best route for the destination. If there is more than one entry to satisfy this condition, the router will choose the route with the lowest metric value. If they still match, then the router will choose the route with the lower index to use. [24.] With a certain given destination, a route that totally matches with the destination address, a network route with longest prefix that matches with the destination route, or a default route will produce the same result.

The chosen route will have the interface and address of the next hop. If the path determination process fails, the IPv6 will assume that the destination can be reached locally. If the routing process on the router fails then the IPv6 will send an ICMP Destination-Unreachable-No-Route packet to the sending Host and dispose the data packet. [24.]

For example is the routing table of a PC running on a Windows environment. In order to check the routing table on Windows 7, the `netsh` command is used as in Figure 21.

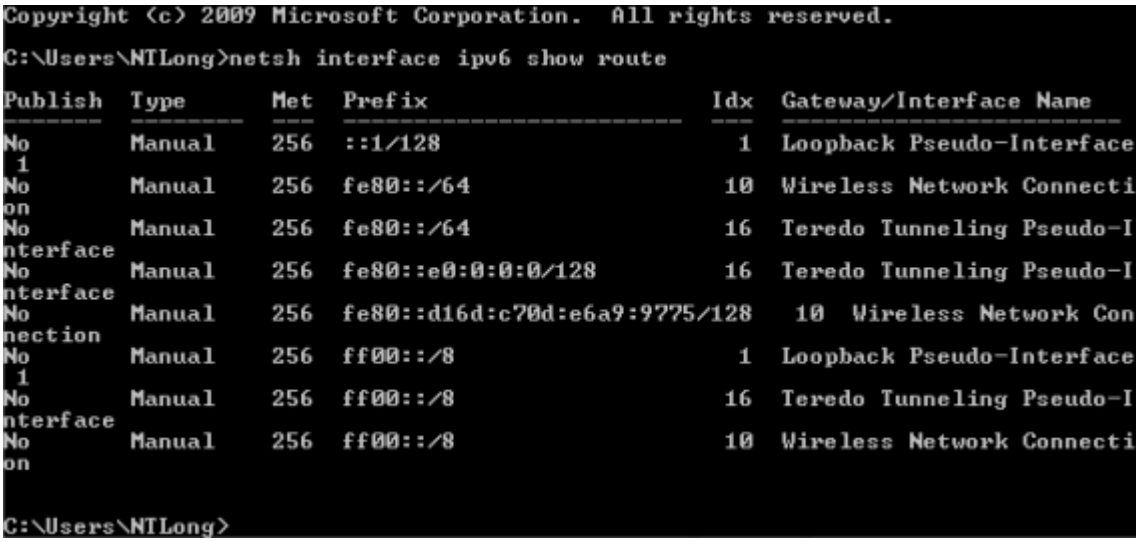

Figure 20. IPv6 routing table on Windows

Each entry of the routing table on a Windows computer contains six fields. The *Publish Field* indicates whether the route is broadcasted (through Router advertisement). The *Type Field* determines routing type (dynamic or static). The *Metric Field* defines which metric is to be chosen when there are routers with the same prefix. The *Prefix Field* contains Network bits. The *Idx Field* determines interface indexes through which packets are sent to. The index can be checked by issue command: `netsh` interface IPv6 show interface. The *Gateway/Interface Name Field* contains next hop address or next hop interface.

With routes of remote networks, an IPv6 of the next hop will be listed. With routes that are directly connected, the name of the interface will be listed. [24.]

With routes that are configured by users' applications, there will be a Manual value assigned to the route type. Routes are that configured by the IPv6 protocol will have the routing type of Auto-configuration. The routing table of IPv6 is built automatically based on the current configuration of Host. Routes for a prefix that has a local connection (starting with FE80::/64) will not appear in the routing table. [24.]

3.4.2 Static Routing

Static routing on IPv6 is not different from static routing on IPv4. Static routing is manually configured and determines a routing path between two nodes within the network. Unlike

dynamic routing, static routing is not updated automatically but must be updated by the system administrator in case there is a change within the network. [14, 813.]

A benefit of using static routing is that it is secured and the efficiency in using router resource. Static routing uses less bandwidth compared to dynamic routing and does not require much performance from CPU to calculate the best route. [14, 813.] A disadvantage of using static routing is the inability of automatically reconfigured in case there are changes within the network structure. The second disadvantage is that there is no algorithm to avoid looping in static routing. [14, 813.]

Static routing is also used for small-scale networks with a single outgoing connection to other networks, and for providing security for a greater-scale network in order to guarantee a certain amount of bandwidth for more control. In fact, most of the networks use dynamic routing for interaction among the nodes but for some special purposes.
[14, 813.]

Configuring static route for IPv6

On Cisco devices, the command is `IPv6 route` within `mode config` to configure static route.

```
IPv6 route IPv6-prefix/prefix-length {IPv6-address | inter-
face-type
interface-number [IPv6-address]} [administrative-distance]
[administrative-multicast-distance | unicast | multicast]
[tag tag]
```

Listing 1. Syntax of the IPv6 route command

For example, a command for configuration of IPv6 route to the destination of 2001:0DB8::/32 from the serial interface 0/1/1 is done as below:

```
IPv6 route 2001:0DB8::/32 serial 0/1/1
```

Listing 2. Configuring static route for data packet to destination of 2001:0DB8::/32, though s0/1/1 interface

Different kinds of static route in IPv6 includes the following

- A directly attached static routes is the kind of static route with a single interface assigned to be the output destination
- A recursive static routes directly points out the next-hop address.
- A fully specified static routes points out both the output and input next hope addresses.
- A floating static routes is backup routing protocol for the dynamic routing protocols. Parameter AD of a Floating Static Route will be higher than AD of the dynamic backup routing protocol. If the dynamic routing path is lost, the floating static route will be immediately used alternatively for that routing path. [14, 813.]

3.4.3 Dynamic Routing Protocols in IPv6

RIPng

Routing Information Protocol next generation (RIPng - RFC 2080) is a routing protocol based on a distance vector with limited amount of hop, using split-horizon, poison reverse, hold-down timer, triggered updates to avoid looping. Similarly to RIP and RIPv2 of IPv4, RIPng uses routing protocol base on Bellman-Ford algorithm, IPv6 for transportation, includes IPv6 prefixes and next-hop IPv6 addresses. FF02::9 is used as multicast address for all of the RIP-Router. FF02::9 is considered as destination address for all of the RIP updates data packets; deliver update information on UDP port 521.

OSPFv3

OSPFv3 is a routing protocol according to the state of the connection route (RFC 2740). OSPFv3 is used for routing in the IPv6 environment, designed to run as a self-monitoring system. It is built based on the OSPFv2 of IPv4. OSPFv3 still uses the algorithm of Dijkstra to construct the routing table. This algorithm is to find the shortest path first to reach the destination. LSA carries router information and the status of neighbouring routers. Depending on the information from LSA, OSPF will build a network topology.[9, 615.]

EIGRP for IPv6

Enhanced Interior Gateway Routing Protocol (EIGRP) is the advanced version of IGRP (Interior Gateway Routing Protocol) developed by Cisco, so it can only be used on Cisco

devices. EIGRP uses the Distance Vector algorithm similarly to IGRP. However EIGRP obsesses higher convergence and performance compared to IGRP. [9, 230.] This convergence technology is researched at the SRI International and it uses an algorithm called the Diffusing Update Algorithm (DUAL). This algorithm ensures a loop free functionality during the process of route calculating, and allows other devices to attend the topology synchronization at the same time. The routers that are affected by the topology changes will not attend the recalculating process. [9, 230.]

With RIP, the maximum length of the network is 15 hops. When EIGRP starts, the maximum length of the network is updated to 224 hops. Because of the metric of EIGRP is enough to support thousands of hops, the only obstacle to extend the network resource is the transport layer. Cisco solves this problem by increasing transport control. The DUAL algorithm allows information to quickly converge routing information like other protocols. EIGRP will send update information when the status of the destination is changed instead of sending all information. Another feature is neighbouring routers discovery because EIGRP is used for big network systems. Route filtering system is executed by using the command `distribute-list prefix-list`. [9, 230-231.]

EIGRP for IPv6 contains the following basic components:

Neighbour discovery is a process in which the router automatically studies other routers that it are directly connected within the network. The router also discovers neighbour routers that it cannot connect to or neighbour routers that are not functional. The EIGRP neighbour also discovers the neighbouring routers that are functioning again and resends them hello packets. With hello packets, the IOS of Cisco can determine whether the neighbouring router is alive and functioning. Once this status is determined, the neighbour routers can exchange routing information.

The *Reliable transport protocol* is a protocol that can be trusted in exchanging EIGRP packets to neighbouring routers. It supports exchanging both multicast and unicast data packets. Some EIGRP packets should be trusted while others. For efficiency, reliability is provided when necessary.

The *DUAL finite state machine* is a methodology that represents the process of issuing decisions for metric calculation. It tracks every route broadcasted by every neighbouring router. DUAL uses the number of metric including distance and cost information to

choose the efficiency not to be looped. When there are many paths leading to a single router exist, DUAL will determine the path with lowest metric and save this into the routing table. The other routes with a higher metric, DUAL will determine distance and inform this to the network. When there is no router feasible successor, neighbouring router broadcast routes, there will be a vote. This is a process that DUAL decides a new successor depending on the required amount of time to calculate the affects the convergence process. The recomputation process is an advanced process. It is an advantage to avoid unnecessary computation. When the topology table changes, DUAL will check for a feasible successor. If a feasible successor is found, DUAL will use them to avoid unnecessary recomputations.

The protocol-dependent relies on a certain network layer. An example is the EIGRP modules are responsible for sending and receiving EIGRP packets in IPv4 and IPv6. It is also responsible for analysing EIGRP packets and informs DUAL about the received packets. EIGRP requests DUAL to release routing decision, and the result will be saved in the routing table of IPv6. Besides, EIGRP is responsible for redistributing other paths learned from the IPv6 routing protocols. [9, 231.]

IS-IS

The Intermediate System-to-Intermediate System is an IGP developed in 1980 by Digital Equipment. Later IS-IS was recognized by ISO as a standard routing protocol. IS-IS was created for building a standard routing protocol, wide area routing, structural routing method, efficiency, fast convergence and low cost.[9, 361.]

In the beginning, IS-IS was built in a way that every system can use it. However, to ensure an opening property, ISO has tried to integrate all of the persuasive properties of other protocols in IS-IS resulting in a complicated protocol. The majority of ISP use IS-IS from the years before IS-IS was created. This is because IS-IS is an independent protocol, extendable and can define a service type during the routing process (ToS routing). IS-IS feature in IPv6 is similar to and provides many advantages as in IPv4. IPv6 upgraded IS-IS, allowing IS-IS to advertise IPv6 prefix beside IPv4. IS-IS in IPv6 supports two network states, which are single topology and multiple topologies. [9, 361.]

3.5 Open Shortest Path First Version 3 (OSPFv3) for IPv6

OSPF is a routing protocol based on the status of the connected paths implemented in open standards; OSPF is described in RFC of IETF (Internet Engineering Task Force). Open standard means that OSPF is used on every routing device of many different producers and has not monopoly. [9, 613.]

Compared with RIPv1 and RIPv2, OSPF is an Interior Gateway Protocol (IGP) for its extendibility and it is sometimes slow for its route selection process regardless of other important properties such as bandwidth. OSPF solves the problems of RIP and it is a strong routing protocol, and extendable, and suitable for the modern network systems. [9, 613.]

3.5.1 Function of Open Shortest Path First Version 3 (OSPFv3)

OSPFv3 works is based on the previous OSPFv2 plus some additional features. OSPF is a link-state routing protocol, opposite to distance vector protocols. A link is considered as a network component. [9, 614.] A link-state decides the path base on the state of the link from source to destination.

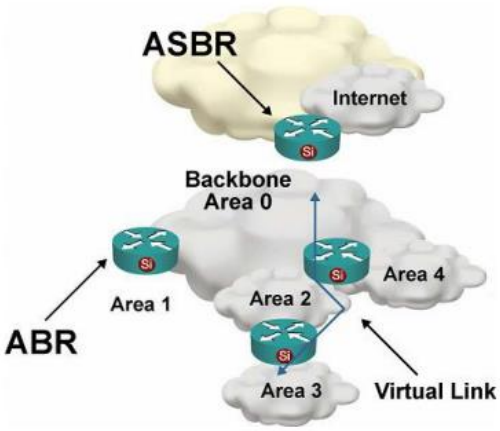

Figure 21. Hierarchy model in OSPFv3.

The state of a link is described as the neighbouring relation of that interface with its neighbouring network devices. Interface information includes IPv6 prefixes of the interface, types of networks and the routing system it is connected to. This information is packed and transferred through data packets called Link-state Advertisements (LSAs). A folder of LSA information in each router is saved within the Link-state information system. The content of the information system uses the Dijkstra algorithm, and the final

result is the routing table of OSPF. The difference between LSDB and the LSDB routing table is a folder of raw information, while the routing table contains a list of the lowest metric path to the destinations which are acknowledged through certain interfaces on routers. [9, 614]

In order to minimize the size of LSDB, OSPF allows calculation and generates areas. An area in OSPF is a group of continuous segments. In all of the subnets of OSPF, there is at least one backbone area which is area 0. The other areas are required to be directly connected to the backbone area, or an inter-connected area to the backbone area. OSPF areas allow summarization or routing information from the boundary OSPF areas. The router at the boundary areas is known as the Area Border Router (ABR). The router located outside the OSPF areas is known as the Autonomous System Boundary Router (ASBR). [9, 614]

3.5.2 Link-State Advertisement (LSA) Packets of IPv6

Every LSAs contains a header of 20 bytes. This header contains enough information to determine a single LSA (LS type, Link State ID, and Advertising Router). In many cases, LSAs can simultaneously exist in routing fields. This can be checked by initiating a check on LS age, LS sequence number and LS checksum fields in LSA header. [9, 616]

Table 4. OSPFv3 LSA header

LS age	LS type
Link State ID	
Advertising Router	
LS sequence number	
LS checksum	Length

As Table 4 demonstrates, the OSPFv3 LSA header contains the following fields:

- LS age: indicates the moment from which the LSA packet is generated up to present.
- LS type: indicates the responsibility that LSA performs; the first 3 bits in LS type indicate the general encryption type of LSA.
- Link state ID: together with LS type and advertisement router ensures the singularity of the LSA in the information system of the link-state.

- Advertisement Router: contains the Router ID of the source router that generated LSA.
- LS sequence number: is the index number of the LSA packet in order to detect out-dated LSA packets and duplicated LSA packets.
- LS checksum: verifies the sum of the LSA packet
- Length: indicates the length of 20 bytes for the LSA packet. [9, 616.]

Basic OSPFv3 multi-area structure

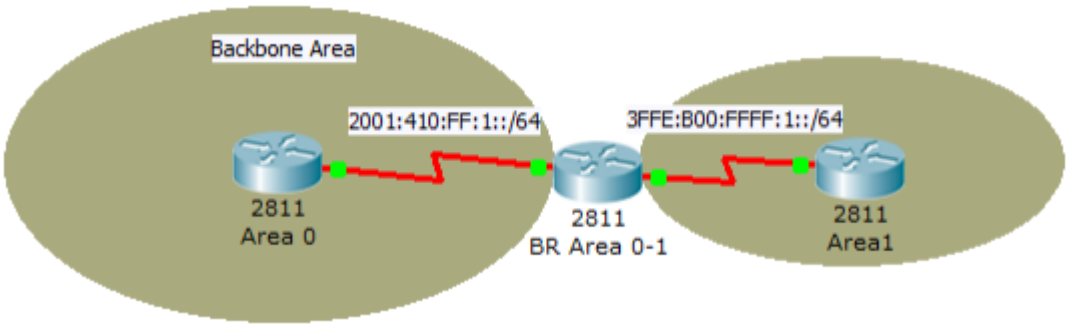

Figure 22. Basic multi-area OSPFv3 structure [16]

As shown by Figure 23, the structure has two areas (area 0 and area 1), three routers, each area has two routers and a boundary router. Once the routers are connected through the Serial port, the DCE ports will belong to the Boundary router. The EUI-64 technology is used on the interfaces.

Configurations applied to the routers are as follows:

```
Area0(config)#ipv6 unicast-routing
Area0(config)#ipv6 router ospf 1
Area0(config-rtr)#router-id 1.1.1.1
Area0(config-rtr)#exit
Area0(config)#interface Serial0/0/0
Area0(config-if)#ipv6 enable
Area0(config-if)#ipv6 address 2001:410:FF:1::/64 eui-64
Area0(config-if)#ipv6 ospf 1 area 0
Area0(config-if)#no shutdown
Area0(config-if)#end
```

Listing 3. Area 0 router configurations

```
BR(config)#ipv6 unicast-routing
BR(config)#ipv6 router ospf 1
```

```
BR(config-rtr)#router-id 2.2.2.2
BR(config-rtr)#exit
BR(config)#interface s0/0/0
BR(config-if)#ipv6 enable
BR(config-if)#ipv6 address 2001:410:FF:1::/64 eui-64
BR(config-if)#ipv6 ospf 1 area 0
BR(config-if)#clock rate 128000
BR(config-if)#no shutdown
BR(config-if)#exit
BR(config)#interface s0/0/1
BR(config-if)#ipv6 enable
BR(config-if)#ipv6 address 3FFE:B00:FFFF:1::/64 eui-64
BR(config-if)#ipv6 ospf 1 area 1
BR(config-if)#clock rate 128000
BR(config-if)#no shutdown
BR(config-if)#end
```

Listing 4. Boundary router configuration

```
Area1(config)#ipv6 unicast-routing
Area1(config)#ipv6 router ospf 1
Area1(config-rtr)#router-id 3.3.3.3
Area1(config-rtr)#exit
Area1(config)#interface Serial0/0/1
Area1(config-if)#ipv6 enable
Area1(config-if)#ipv6 address 3FFE:B00:FFFF:1::/64 eui-64
Area1(config-if)#ipv6 ospf 1 area 1
Area1(config-if)#no shutdown
Area1(config-if)#end
```

Listing 5. Area 1 router configuration

The commands shown in Listings 3 - 5 results of the routing process can be checked by issuing the following commands `show ipv6 route, show ipv6 ospf; show ipv6 ospf<1-65535>, border-routers, database, interface, neighbour`

3.6 Transition Mechanism Between IPv4 and IPv6

The exchanging process from using IPv4 and changing to IPv6 addressing is not an easy task that can be done quickly. Once the IPv6 addressing processes have been standardized, completed and can function properly, the exchanging process can be initiated within a certain limited amount of time for small-scale networks. However, this can hardly be done quickly on a big scale network. With the global Internet system, an instant transformation from IPv4 to IPv6 is not possible. [17.]

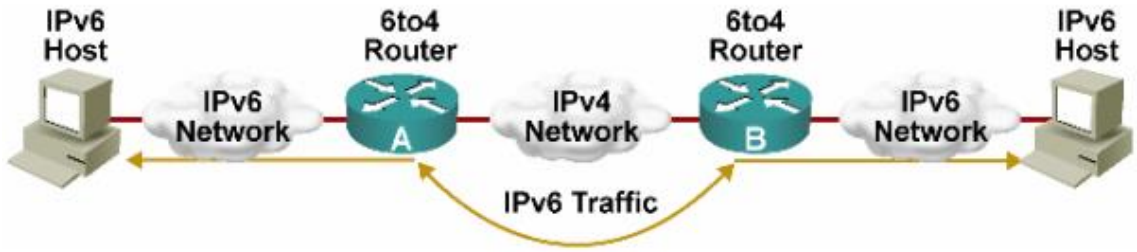

Figure 23. The exchanging process between IPv4 and IPv6 (reprinted from [17]).

IPv6 was developed when IPv4 was already widely used, and the IPv4 network was already completed and performed constantly. During the development process of the IPv6 addresses on the Internet, it is not possible at a certain time to eliminate IPv4 will, IPv6 and IPv4 will continue to exist simultaneously for years, and IPv6 will be developing on the basis of IPv4 resources. In fact, there have been technologies that were developed to support the exchanging between IPv4 and IPv6 such as Dual Stack, Tunnelling, and NAT-PT. [17.]

3.6.1 Dual Stack

The Dual stack model is a method to initiate TCP/IP including both the IP layer of IPv4 and the IP layer of IPv6. This method supports both IPv4 and IPv6 protocols, allowing the operation system or application to select one of the two protocols for each communication (according to the default standard is giving priority to IPv6 where it is possible to use IPv6). Dual stack is widely used on Windows OS, and Linux, and the operation systems on routing devices of Cisco, and Juniper. [14, 567]

3.6.2 Tunnelling

Tunnelling is a technology that utilizes IPv4 resources to transmit IPv6 packets, serving IPv6 communications. IPv6 addressing develops when the Internet IPv4 is widely used. IPv6 at the beginning would be used as islands, isolated within the great IPv4 network system, and in order to connect these nodes, the tunnelling technology is used. [18.]

Tunnelling utilizes devices that can initiate dual stack connectivity from both ends. These devices pack IPv6 packets in IPv4 and transmit them in the IPv4 network. Once they arrive, these devices once again would remove the IPv4 headers and retrieve the original

IPv6 packets at the destination point. In other words, tunnelling functions as a virtual connection of IPv6. [18.]

There are various kinds of tunnelling due to different requirements and demands from different groups of user, such as Manual Tunnel, Automatic Tunnel (Intra-site Automatic Tunnel Addressing Protocol, Teredo Tunnelling, 6to4 Tunnelling), Configured Tunnel, Tunnel Broker, and Tunnel Server. [18.]

3.6.3 Network Address Translation/Protocol Translation (NAT-PT)

In order to allow a device that only supports IPv6 to communicate with another device that only supports IPv4, the Network Address Translation-Protocol Translation is a solution that makes an important contribution to assist users to switch from IPv4 to IPv6. This solution is described in RFC 2766. The exchange from IPv4 to IPv6 allows hosts belonging to different aspects of the network to connect to each other. The device that provides the NAT-T service will retranslate headers and address, which allows IPv6 networks to interact with IPv4 networks. [14, 566]

3.7 Network Models

3.7.1 Dual Stack Model

The Dual Stack model is a model that is based entirely on the transition mechanism of Dual Stack. Devices or networks within this model have two protocols enabled and operate at the same time. In the deployment of the IPv6 network on an existing IPv4 environment, the Dual Stack model is most preferred because it allows using IPv4 on connections in which devices do not support IPv6. Additionally IPv6 can be enabled on other connections in which IPv4 is no longer needed. [15, 4]

3.7.2 Hybrid Model

The Hybrid Model strategy approaches in such a way that it can utilize the existing network infrastructure, employing two independent transition mechanisms. Transition mechanisms are selected based on multiple criteria, such as hardware capabilities with IPv6,

number of hosts, purpose, location, as well as network infrastructure feature support for different transition mechanisms. The three main IPv6 transition mechanism leveraged by Hybrid Model are Dual Stack, ISATAP, and manually-configured tunnels. [15, 6]

3.7.3 Service Block Model

The Service Block model is different from the other two models mentioned in sections 3.7.1 and 3.7.2 even though the concept of a service block design is not new. The model offers unique capabilities to users who require access to IPv6 services in a short time. The Service Block Model is unique so that it can be implemented rapidly as an overlay network without causing any impact on the existing IPv4 network while allowing high availability of IPv6 services, QoS, and restriction access to IPv6 resources. [15, 13-14.]

The key to maintain highly scalable and redundant configuration in this model is to ensure a high-performance switch, and supervisor. The modules are used to handle the load of the ISATAP, manually-configured tunnels, and dual-stack connections for an entire campus network. [15, 13-14.]

4 Simulating the IPv6 Network

4.1 GNS installation and configuration

GNS3 is software that simulates the real operation system running on routers and switches utilizing its core of dynamips. GNS3 can simulate Cisco and Juniper devices, the software was created by Christophe Fillot (according to gns3.net).

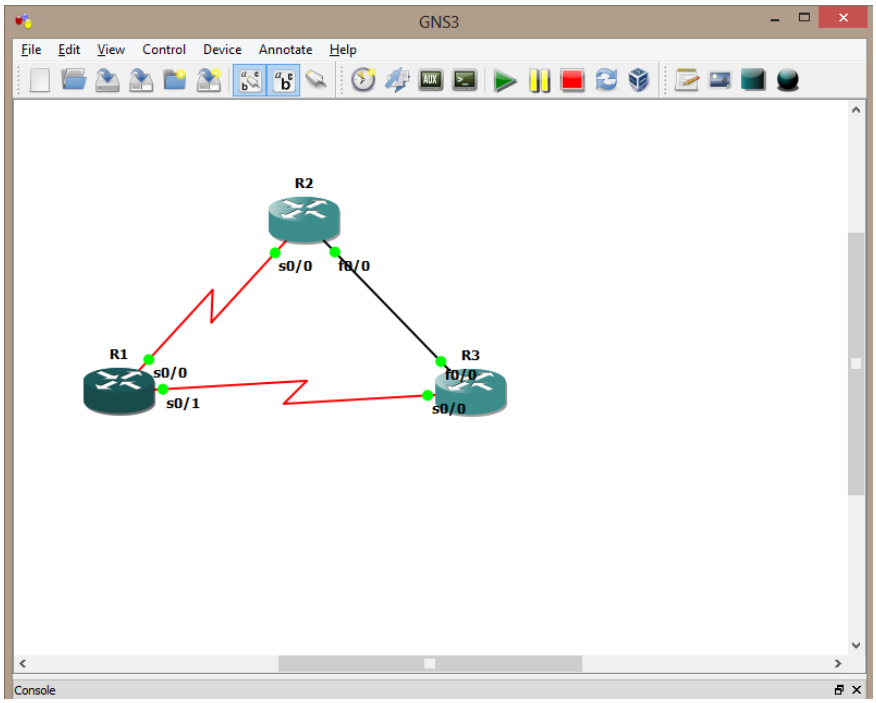

Figure 24. GNS3 user interface

GNS3 can support designing simple and complex network topologies, and simulating different iOS version of Cisco, IPS, PIX firewall, ASA firewall, and JunOS of Juniper routers. It is also possible to simulate switches or using Wireshark to analyse packages.

4.2 Testing Network

The testing network this project intended to build was the core system of a sample campus network among the routers. Within the core network, the following listed technologies which are useful for the process of implementation IPv6 addressing on the basis of existing IPv4 resources (six Cisco Catalyst 3750 routers, switches, workstations, EIGRPv2 routing protocol) were implemented:

- IPv4 and IPv6 addresses for every serial and loopbacks interfaces
- A default link-local address in R1 and R2
- EUI-64 format addresses for the Fast Ethernet connection between R3 and R2
- OSPFv3 routing protocol for the IPv6 network between R1-R4
- EIGRP routing protocol for the IPv4 network for all routers
- A manual IPv6 Tunnel 0 between R4 and R2

- OSPFv3 routing for Tunnel 0
- A 6-to-4 Tunnel 1 between R4 and R5, R6 plays the role of an old router that is not able to support IPv6 routing.
- Static routes through Tunnel 1.

The topology of the testing network is as shown in figure 25.

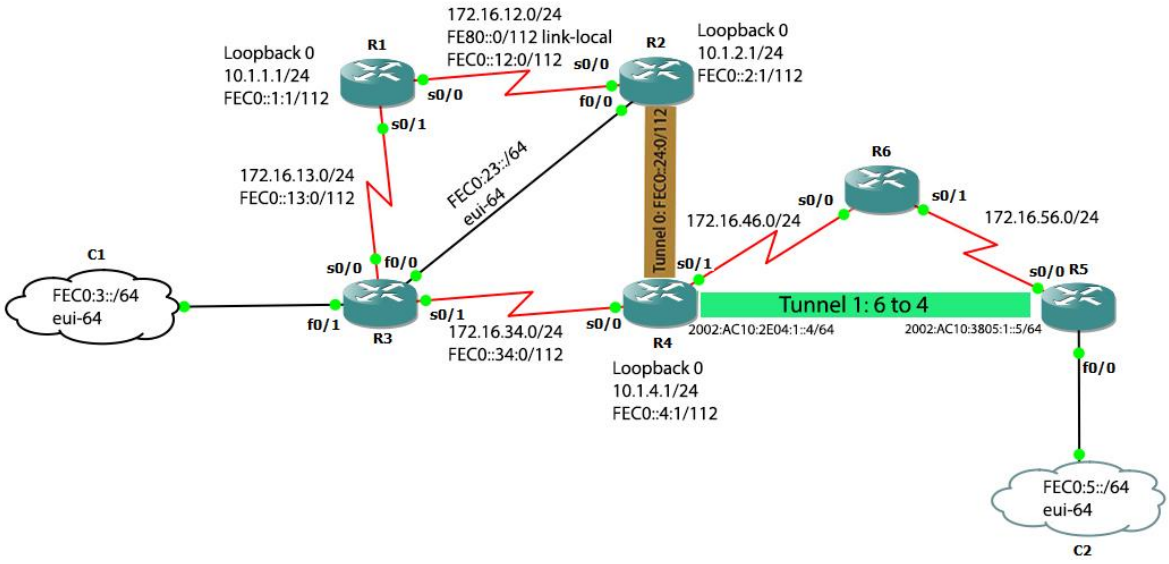

Figure 25. Topology of the testing network

The concept of the testing network is to give an example of an IPv4 network, with IPv4 addresses assigned to each of the nodes. The IPv4 network uses the EIGRP routing protocol. A network specialist of the university X will configure the stateless autoconfiguration IPv6 on R3, which will automatically allow a new device connected to Cloud 1 to receive an IPv6 address in the eui-64 format. The new IPv6 configuration will function simultaneously with the old IPv4 network on the same resources. The OSPFv3 routing protocol is used for IPv6 routing. A manual IPv6 tunnel is configured from R4 to R2. R5 and R6 represent a remote connection with a medium that might or might not support IPv6, so I configured a 6-to-4 tunnel from R4 to R5. The static route is configured in order to route a package from R4 to Cloud 2 under R5. The static route is redistributed into the OSPFv3 routing system and can be confirmed by issueing trace route commands from any of the hosts in Cloud 1 to a host in Cloud 2.

Table 5. Configuration of the testing network

Devices	Interface	IPv4	IPv6
R1	Serial 0/0	172.16.12.1/24	FE80::1/112 link-local
			FEC0::12:1/112
	Serial 0/1	172.16.13.1/24	FEC0::13:1/112
	Loopback 0	10.1.1.1/24	FEC0::1:1/12
R2	Serial 0/0	172.16.12.2/24	FE80::2/112 link-local
			FEC0::12:2/112
	Serial 0/1	172.16.25.2/24	FEC0::25:2/112
	FastEthernet 0/0		FEC0:23::/64 eui-64
	Loopback 0	10.1.2.1/24	FEC0::2:1/112
	Tunnel 0		FEC0::24:2/112
R3	Serial 0/0	172.16.13.3/24	FEC0::13:3/112
	Serial 0/1	172.16.34.3/24	FEC0::34:3/112
	FastEthernet 0/0		FEC0:23::/64 eui-64
	FastEthernet 0/1	10.1.3.1/24	FEC0:3::/64 eui-64
R4	Serial 0/0	172.16.34.4/24	FEC0::34:4/112
	Serial 0/1	172.16.46.4/24	
	Tunnel 0		FEC0::24:4/112
	Tunnel 1 (6 to 4)		2002:AC10:2E04:1::4/64
	Loopback 0	10.1.4.1/24	FEC0::4:1/112
R5	Serial 0/0	172.16.56.5/24	FEC0::25:5/112
	Tunnel 1 (6 to 4)		2002:AC10:3805:1::5/64
	FastEthernet 0/0	10.1.5.1/24	FEC0:5:1::/64 eui-64
R6	Serial 0/0	172.16.46.6/24	
	Serial 0/1	172.16.56.6/24	

Table 5 shows the detailed configuration for the testing network. The configurations for each router of the testing networks are as shown in Appendix 1 - 5.

5 Implementation Results

The performance was tested to be correctly functional. The IPv6 routing table on R4 was checked by issuing the command `show ipv6 route`. The routing table on R4 and R3 was as shown in Listing 6 and Listing 7.

```
R4#show ipv6 route
IPv6 Routing Table - 17 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS -
ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2
- OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
S   2002::/16 [1/0]
via ::, Tunnel1
C   2002:AC10:2E04:1::/64 [0/0]
     via ::, Tunnel1
L   2002:AC10:2E04:1::4/128 [0/0]
via ::, Tunnel1
O   FEC0::1:1/128 [110/128]
via FE80::C602:1CFF:FE6C:0, Serial0/0
O   FEC0::2:1/128 [110/74]
via FE80::C602:1CFF:FE6C:0, Serial0/0
C   FEC0::4:0/112 [0/0]
via ::, Loopback0
L   FEC0::4:1/128 [0/0]
via ::, Loopback0
O   FEC0::12:0/112 [110/138]
via FE80::C602:1CFF:FE6C:0, Serial0/0
O   FEC0::13:0/112 [110/128]
via FE80::C602:1CFF:FE6C:0, Serial0/0
C   FEC0::24:0/112 [0/0]
     via ::, Tunnel0
L   FEC0::24:4/128 [0/0]
     via ::, Tunnel0
C   FEC0::34:0/112 [0/0]
via ::, Serial0/0
L   FEC0::34:4/128 [0/0]
via ::, Serial0/0
O   FEC0:3::/64 [110/74]
via FE80::C602:1CFF:FE6C:0, Serial0/0
S   FEC0:5::/64 [1/0]
via 2002:AC10:3805:1::5
O   FEC0:23::/64 [110/74]
via FE80::C602:1CFF:FE6C:0, Serial0/0
L   FF00::/8 [0/0]
via ::, Null0
```

Listing 6. Routing table on R4

```
R3#show ipv6 route
```

```
IPv6 Routing Table - 16 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS -
ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2
- OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
OE2  2002::/16 [110/20]
via FE80::C603:1CFF:FE6C:0, Serial0/1
O    FEC0::1:1/128 [110/64]
via FE80::C600:1CFF:FE6C:0, Serial0/0
O    FEC0::2:1/128 [110/10]
via FE80::C601:1CFF:FE6C:0, FastEthernet0/0
O    FEC0::4:1/128 [110/64]
via FE80::C603:1CFF:FE6C:0, Serial0/1
O    FEC0::12:0/112 [110/74]
via FE80::C601:1CFF:FE6C:0, FastEthernet0/0
C    FEC0::13:0/112 [0/0]
via ::, Serial0/0
L    FEC0::13:3/128 [0/0]
via ::, Serial0/0
O    FEC0::24:0/112 [110/11121]
via FE80::C601:1CFF:FE6C:0, FastEthernet0/0
C    FEC0::34:0/112 [0/0]
via ::, Serial0/1
L    FEC0::34:3/128 [0/0]
via ::, Serial0/1
C    FEC0:3::/64 [0/0]
via ::, FastEthernet0/1
L    FEC0:3::C602:1CFF:FE6C:1/128 [0/0]
via ::, FastEthernet0/1
OE2  FEC0:5::/64 [110/20]
via FE80::C603:1CFF:FE6C:0, Serial0/1
C    FEC0:23::/64 [0/0]
via ::, FastEthernet0/0
L    FEC0:23::C602:1CFF:FE6C:0/128 [0/0]
via ::, FastEthernet0/0
L    FF00::/8 [0/0]
via ::, Null0
```

Listing 7. Routing table on R3

Confirmation for the configurations of Tunnel 0 - Manual IPv6 Tunnel and Tunnel 1 - 6 to 4 can be checked by issuing the command `show interface tunnel [id]`. The results are demonstrated in Listing 8.

```
R4#show int tunnel 0
Tunnel0 is up, line protocol is up
  Hardware is Tunnel
  MTU 1514 bytes, BW 9 Kbit, DLY 500000 usec,
reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
Keepalive not set
```

```
Tunnel    source    172.16.34.4    (Serial0/0),    destination
172.16.12.2
Tunnel protocol/transport IPv6/IP
  Tunnel TTL 255
  Fast tunneling enabled
Tunnel transmit bandwidth 8000 (kbps)
Tunnel receive bandwidth 8000 (kbps)
  Last input 00:00:07, output 00:00:05, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total out-
put drops: 10
Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     666 packets input, 82732 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0
abort
     676 packets output, 71368 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out


R4#show int tunnel 1
Tunnel1 is up, line protocol is up
  Hardware is Tunnel
  MTU 1514 bytes, BW 9 Kbit, DLY 500000 usec,
reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
Keepalive not set
Tunnel source 172.16.46.4 (Serial0/1), destination UNKNOWN
Tunnel protocol/transport IPv6 6to4
  Tunnel TTL 255
  Fast tunneling enabled
Tunnel transmit bandwidth 8000 (kbps)
Tunnel receive bandwidth 8000 (kbps)
  Last input 00:11:55, output 00:11:55, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total out-
put drops: 0
Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     88 packets input, 10976 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0
abort
     179 packets output, 19976 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Listing 8. Tunnel 0 and Tunnel 1 functional configurations.

Figure 26 shows the IPv4 and IPv6 configuration on a host machine within Cloud 1 and Cloud 2 of the testing network.

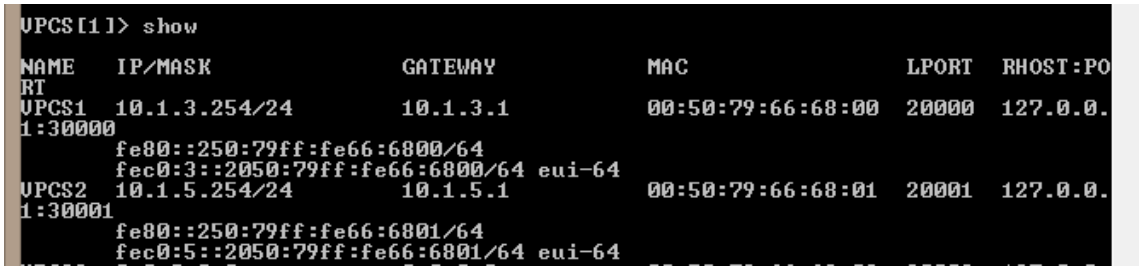

Figure 26. IPv4 and Auto-configured IPv6 eui-64 format addresses for hosts in C1 and C2

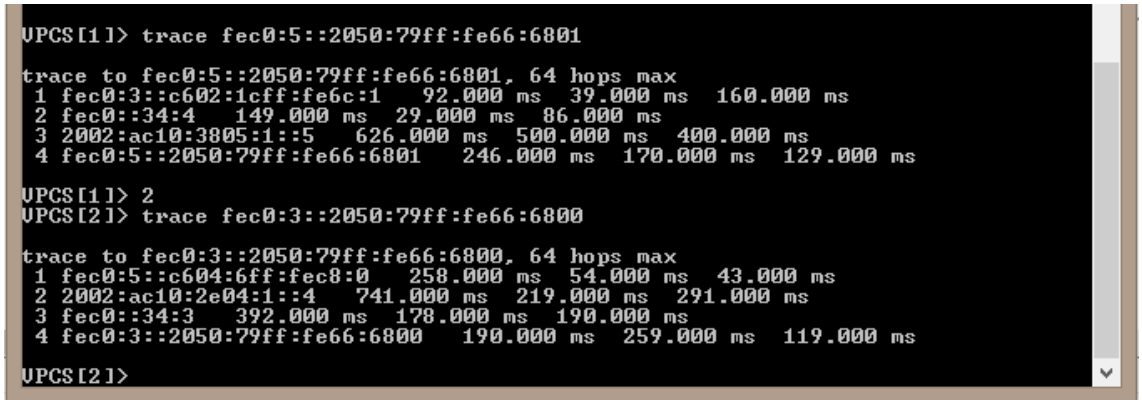

Figure 27. Trace route results from hosts in C1 and C2

Figure 27 shows the results of the trace route tested on the host machine from Cloud 1 to Cloud 2 and from the host machine from Cloud 2 to Cloud 1.

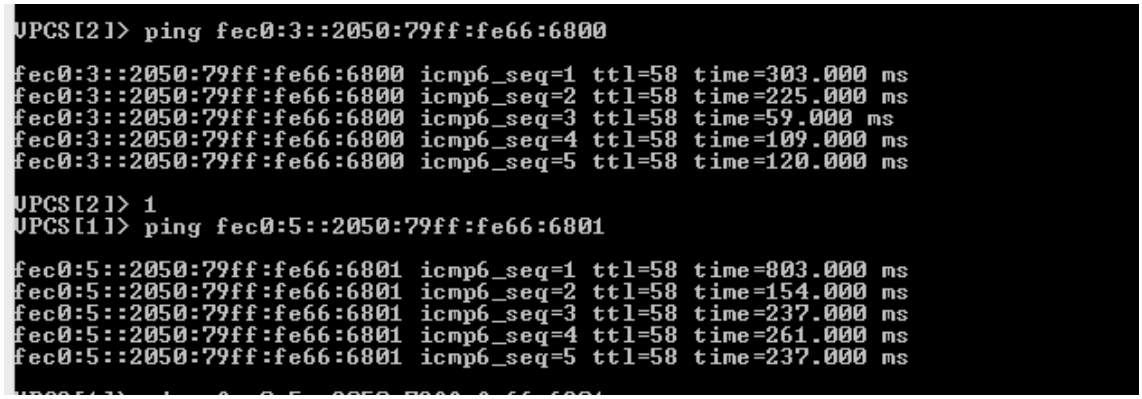

Figure 28. Pinging from end to end result.

Figure 28 shows successful pinging results from end to end of the network.

6 Project Evaluation and Conclusion

Throughout this project, I improved my understanding and skills in the IPv6 addressing system. The project achieved the goals of analysing the concept, analyzed the concept, new features, and the structure of IPv6 as well as comparing IPv6 with the previous IPv4 to help understanding the necessity for upgrade. The testing output showed successful result. The resulting document reached the goal of being capable to serve as a reference of implementation for system engineers and product engineers for the deployment of IPv6 protocol on a production network.

The deployment of IPv6 into the testing network was the major result of this project. Overall, the result was very positive, since all the technologies that were integrated into the network were to be correctly functional. Despite the achievements of the project, there are still some drawbacks. The project managed to describe routing and conversion technologies and but they are still at the introductory level. In this thesis, some of the given figures have general descriptions. Furthermore, the testing network was not built based on a practical network model.

Further development of the project could be to implement and systematically build an IPv6 campus network from the scale of enterprise up to a national-scale and global network, creating the backbone for development of a new generation of services on IPv6 addressing.

To conclude, I would like to emphasize the fact that IPv4 addressing has existed for three decades, the addressing space exhaustion is spreading globally and the upgrade to IPv6 is required for every organization in particular and for the Internet world in general.

References

1. IPv4 Address Exhaution and IPv6 [online]. Office of the Government Chief Information Officer.
   URL: http://www.ogcio.gov.hk/en/business/tech_promotion/IPv6/IPv4_address_exhaustion.htm. Accessed 14 February 2013.

2. ARPANET timeline [online]. Technical Histories of the Internet& Other Network Protocols.
   URL: http://www.cs.utexas.edu/~chris/think/ARPANET/Timeline/
   Accessed 14 February 2013

3. McFarland Shannon, Sambi Muninder, Sharma Nikhil, Hooda Sanjay. IPv6 for Enterprise Network. First Edition. Cisco Press; 2011.

4. Teare Diane, Paquet Catherine. CCNP Self-study: Advanced IP Addressing [online]. Second Edition. Cisco Press; 11 June 2004.
   URL: http://www.ciscopress.com/articles/article.asp?p=174107
   Accessed 3 March 2013.

5. Teare Diane, Paquet Catherine. Building Scalable Cisco Internetworks (BSCI) (Authorized Self-Study Guide). Third Edition. Cisco Press; 2006.

6. Hogg Scott,Vyncke Eric. IPv6 Security. Information assurance for the next-generationInternet Protocol. Cisco Press; December 2008.

7. Unicast IPv6 Addresses [online].
   URL: http://technet.microsoft.com/en-us/library/cc759208(v=ws.10).aspx.
   Accessed 14 February 2013

8. Internet Address Lab [online].
   URL: http://blog.naver.com/PostView.nhn?blogId=hsyoo2k&logNo=60105077543.
   Accessed 14 February 2013

9. Cisco IOS IPv6 Configuration Guide, Cisco IOS Release 15.2MT. Cisco Press; 2012

10. McQuerry Steve. Interconnecting Cisco Network Devices (Cisco Career Certifications). Second Edition. Cisco Press; 2003.

11. Bruno.Anthony, Jordan Steve. CCDA Offcial Exam Certification Guide. Cisco Press; June 2007.

12. Government launches 10-year infocomm masterplan [online]. Singapore update.
    URL: http://www.singaporeupdate.com/previous2006/220606_governmentlaunches10yearinfocommmasterplan_more.htm
    Accessed 17 February 2013.

13. KarlssonBjörn. Cisco Self-Study: Implementing IPv6 Networks (IPv6). Cisco Press; 2003.

14. IPv6 Configuration Guide, Cisco IOS. Release 12.4T Cisco Press; 2012.

15. Deploying IPv6 Campus Network. Cisco Press; 2006.

16. OSPFv3 (RFC 2740) [online].
URL: http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/prod_presentation0900aecd80311e31.pdf. Accessed 3 March 2013.

17. Implementing IPv6 Routing (Workshop) [online].
URL: http://www.itc-trainingcenter.net/training/networking-course/Implementing_for_IPv6_Routing_Workshop.php
Accessed 3 March 2013.

18. Das, Kaushik. IPv6 Transition Technologies [online]. IPv6.com Tech Spotlight.
URL: http://www.ipv6.com/articles/gateways/IPv6-Tunnelling.htm
Accessed 3 March 2013.

19. IPv4 to Ipv6 Translations: Comparing network-ready routing solutions. [online]
URL: http://searchtelecom.techtarget.com/tip/IPv4-to-IPv6-translation-Comparing-network-ready-routing-solutions. Accessed 3 March 2013.

20. Kevin. CCIE 123 [online].
URL: http://ccie123.wordpress.com/2009/12/22/ipv6-address-types/
Accessed 3 March 2013

21. IPv4 to IPv6 Migration Strategies and Challenges [online].
URL: http://www.telefocal.com/macrosite/en/resource/research-articles/198-ipv4-to-ipv6-migration-strategies-and-challenges.
Accessed 4 March 2013

22. Celtdra Aragoen. BSCI: IPv6 Configuration Exercises [Dynamips Lab] [online]. Route my world ; A CCNA/CCNP Blog.
URL: http://routemyworld.com/category/ipv6/. Accessed 4 March 2013

23. DHCPv6 Based IPv6 Access Services [online]. Cisco; October 2011.
URL: http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/whitepaper_c11-689821.html. Accessed 4 March 2013

24. Configuring IPv6 Routing [online]. Catalyst 3750 Software Configuration Guide, Release 12.2(55)SE. Cisco.
URL: http://www.cisco.com/en/US/docs/switches/lan/catalyst3750/software/release/12.2_55_se/configuration/guide/swipv6.html#wp1305314
Accessed 4 March 2013.

25. R. Hinden, S. Deering. IPv6 Addressing Architecture [online]. Network Working Group; February 2006
URL: http://www.ietf.org/rfc/rfc4291.txt. Accessed 5 March 2013.

Appendix 1: Configuration of R1

```
R1(config-if)#do show run
Building configuration...

Current configuration : 1460 bytes
!
version 12.4
service timestamps debug datetimemsec
service timestamps log datetimemsec
no service password-encryption
!
hostname R1
!
boot-start-marker
boot-end-marker
!
noaaa new-model
memory-sizeiomem 5
ipcef
!
noip domain lookup
ip domain name lab.local
ipauth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
ipv6 unicast-routing
ipv6cef
multilink bundle-name authenticated
!
archive
logconfig
hidekeys
!
interface Loopback0
ip address 10.1.1.1 255.255.255.0
ipv6 address FEC0::1:1/112
ipv6ospf 1 area 0
!
interface FastEthernet0/0
noip address
shutdown
duplex auto
speed auto
!
interface Serial0/0
ip address 172.16.12.1 255.255.255.0
ipv6 address FE80::1 link-local
ipv6 address FEC0::12:1/112
ipv6ospf 1 area 0
no fair-queue
clock rate 64000
!
interface FastEthernet0/1
noip address
shutdown
duplex auto
speed auto
```

```
!
interface Serial0/1
ip address 172.16.13.1 255.255.255.0
ipv6 address FEC0::13:1/112
ipv6ospf 1 area 0
clock rate 64000
!
routereigrp 1
network 10.0.0.0
network 172.16.0.0
no auto-summary
!
ip forward-protocol nd
!
noip http server
noip http secure-server
!
ipv6 router ospf 1
log-adjacency-changes
!
control-plane
!
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous
line aux 0
exec-timeout 0 0
privilege level 15
logging synchronous
linevty 0 4
login
!
end
```

Appendix 2: Configuration of R2

```
R2(config-if)#do show run
Building configuration...

Current configuration : 1605 bytes
!
version 12.4
service timestamps debug datetimemsec
service timestamps log datetimemsec
no service password-encryption
!
hostname R2
!
boot-start-marker
boot-end-marker
!
noaaa new-model
memory-sizeiomem 5
ipcef
!
noip domain lookup
ip domain name lab.local
ipauth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
ipv6 unicast-routing
ipv6cef
multilink bundle-name authenticated
!
archive
logconfig
hidekeys
!
interface Loopback0
ip address 10.1.2.1 255.255.255.0
ipv6 address FEC0::2:1/112
ipv6ospf 1 area 0
!
interface Tunnel0
noip address
ipv6 address FEC0::24:2/112
ipv6ospf 1 area 0
tunnel source Serial0/0
tunnel destination 172.16.34.4
tunnel mode ipv6ip
!
interface FastEthernet0/0
noip address
duplex auto
speed auto
ipv6 address FEC0:23::/64 eui-64
ipv6ospf 1 area 0
!
interface Serial0/0
ip address 172.16.12.2 255.255.255.0
ipv6 address FE80::2 link-local
ipv6 address FEC0::12:2/112
```

```
ipv6ospf 1 area 0
no fair-queue
clock rate 64000
!
interface FastEthernet0/1
noip address
shutdown
duplex auto
speed auto
!
interface Serial0/1
noip address
shutdown
clock rate 2000000
!
routereigrp 1
network 10.0.0.0
network 172.16.0.0
no auto-summary
!
ip forward-protocol nd
!
noip http server
noip http secure-server
!
ipv6 router ospf 1
log-adjacency-changes
!
control-plane
!
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous
line aux 0
exec-timeout 0 0
privilege level 15
logging synchronous
linevty 0 4
login
!
end
```

Appendix 3: Configuration of R3

```
R3(config-if)#do show run
Building configuration...

Current configuration : 1574 bytes
!
version 12.4
service timestamps debug datetimemsec
service timestamps log datetimemsec
no service password-encryption
!
hostname R3
!
boot-start-marker
boot-end-marker
!
noaaa new-model
memory-sizeiomem 5
ipcef
!
noip domain lookup
ip domain name lab.local
ipauth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
ipv6 unicast-routing
ipv6cef
multilink bundle-name authenticated
!
archive
logconfig
hidekeys
!
interface Loopback0
ip address 10.1.3.1 255.255.255.0
shutdown
ipv6 address FEC0::3:1/112
ipv6ospf 1 area 0
!
interface FastEthernet0/0
noip address
duplex auto
speed auto
ipv6 address FEC0:23::/64 eui-64
ipv6ospf 1 area 0
!
interface Serial0/0
ip address 172.16.13.3 255.255.255.0
ipv6 address FEC0::13:3/112
ipv6ospf 1 area 0
no fair-queue
clock rate 2000000
!
interface FastEthernet0/1
ip address 10.1.3.1 255.255.255.0
duplex auto
```

```
speed auto
ipv6 address FEC0:3::/64 eui-64
ipv6nd prefix FEC0:3::/64
ipv6ospf 1 area 0
!
interface Serial0/1
ip address 172.16.34.3 255.255.255.0
ipv6 address FEC0::34:3/112
ipv6ospf 1 area 0
clock rate 64000
!
routereigrp 1
network 10.0.0.0
network 172.16.0.0
no auto-summary
!
ip forward-protocol nd
!
noip http server
noip http secure-server
!
ipv6 router ospf 1
log-adjacency-changes
!
control-plane
!
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous
line aux 0
exec-timeout 0 0
privilege level 15
logging synchronous
linevty 0 4
login
!
end
```

Appendix 4: Configuration of R4

```
R4(config-if)#do show run
Building configuration...

Current configuration : 1831 bytes
!
version 12.4
service timestamps debug datetimemsec
service timestamps log datetimemsec
no service password-encryption
!
hostname R4
!
boot-start-marker
boot-end-marker
!
noaaa new-model
memory-sizeiomem 5
ipcef
!
noip domain lookup
ip domain name lab.local
ipauth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
ipv6 unicast-routing
ipv6cef
multilink bundle-name authenticated
!
archive
logconfig
hidekeys
!
interface Loopback0
ip address 10.1.4.1 255.255.255.0
ipv6 address FEC0::4:1/112
ipv6ospf 1 area 0
!
interface Tunnel0
noip address
ipv6 address FEC0::24:4/112
ipv6ospf 1 area 0
tunnel source Serial0/0
tunnel destination 172.16.12.2
tunnel mode ipv6ip
!
interface Tunnel1
noip address
noip redirects
ipv6 address 2002:AC10:2E04:1::4/64
tunnel source Serial0/1
tunnel mode ipv6ip 6to4
!
interface FastEthernet0/0
noip address
shutdown
```

```
duplex auto
speed auto
!
interface Serial0/0
ip address 172.16.34.4 255.255.255.0
ipv6 address FEC0::34:4/112
ipv6ospf 1 area 0
no fair-queue
clock rate 64000
!
interface FastEthernet0/1
noip address
shutdown
duplex auto
speed auto
!
interface Serial0/1
ip address 172.16.46.4 255.255.255.0
clock rate 64000
!
routereigrp 1
network 10.0.0.0
network 172.16.0.0
no auto-summary
!
routerospf 1
log-adjacency-changes
redistribute static
!
ip forward-protocol nd
!
noip http server
noip http secure-server
!
ipv6 route 2002::/16 Tunnel1
ipv6 route FEC0:5::/64 2002:AC10:3805:1::5
ipv6 router ospf 1
log-adjacency-changes
redistribute static
!
control-plane
!
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous
line aux 0
exec-timeout 0 0
privilege level 15
logging synchronous
linevty 0 4
login
!
end
```

Appendix 5: Configuration of R5

```
R5(config)#do show run
Building configuration...

Current configuration : 1534 bytes
!
version 12.4
service timestamps debug datetimemsec
service timestamps log datetimemsec
no service password-encryption
!
hostname R5
!
boot-start-marker
boot-end-marker
!
noaaa new-model
memory-sizeiomem 5
ipcef
!
noip domain lookup
ip domain name lab.local
ipauth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
ipv6 unicast-routing
ipv6cef
multilink bundle-name authenticated
!
archive
logconfig
hidekeys
!
interface Tunnel1
noip address
noip redirects
ipv6 address 2002:AC10:3805:1::5/64
tunnel source Serial0/0
tunnel mode ipv6ip 6to4
!
interface FastEthernet0/0
ip address 10.1.5.1 255.255.255.0
duplex auto
speed auto
ipv6 address FEC0:5::/64 eui-64
ipv6nd prefix FEC0:5::/64
!
interface Serial0/0
ip address 172.16.56.5 255.255.255.0
no fair-queue
clock rate 64000
!
interface FastEthernet0/1
noip address
shutdown
duplex auto
```

```
speed auto
!
interface Serial0/1
noip address
shutdown
clock rate 2000000
!
routereigrp 1
network 10.0.0.0
network 172.16.0.0
no auto-summary
!
routerospf 1
log-adjacency-changes
!
ip forward-protocol nd
!
noip http server
noip http secure-server
!
ipv6 route 2002::/16 Tunnel1
ipv6 route FEC0:3::/64 2002:AC10:2E04:1::4
ipv6 router ospf 1
log-adjacency-changes
!
control-plane
!
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous
line aux 0
exec-timeout 0 0
privilege level 15
logging synchronous
linevty 0 4
login
!
end
```

Appendix 6: Configuration of R6

```
R6(config-router)#do show run
Building configuration...

Current configuration : 1116 bytes
!
version 12.4
service timestamps debug datetimemsec
service timestamps log datetimemsec
no service password-encryption
!
hostname R6
!
boot-start-marker
boot-end-marker
!
noaaa new-model
memory-sizeiomem 5
ipcef
!
noip domain lookup
ip domain name lab.local
ipauth-proxy max-nodata-conns 3
ip admission max-nodata-conns 3
!
multilink bundle-name authenticated
!
archive
logconfig
hidekeys
!
interface FastEthernet0/0
noip address
shutdown
duplex auto
speed auto
!
interface Serial0/0
ip address 172.16.46.6 255.255.255.0
clock rate 2000000
!
interface FastEthernet0/1
noip address
shutdown
duplex auto
speed auto
!
interface Serial0/1
ip address 172.16.56.6 255.255.255.0
clock rate 64000
!
routereigrp 1
network 172.16.0.0
auto-summary
!
ip forward-protocol nd
!
```

```
!
noip http server
noip http secure-server
!
control-plane
!
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous
line aux 0
exec-timeout 0 0
privilege level 15
logging synchronous
linevty 0 4
login
!
end
```