IMPROVING IPv6 PACKETS TRANSMISSION OVER HIGH SPEED NETWORKS BY INTRODUCING CRC EXTENSION HEADER

SUPRIYANTO

UNIVERSITI SAINS MALAYSIA

2010

IMPROVING IPv6 PACKETS TRANSMISSION OVER HIGH SPEED NETWORKS BY INTRODUCING CRC EXTENSION HEADER

by

SUPRIYANTO

Thesis submitted in fulfillment of the requirements
for the degree of
Master of Science

April 2010

ACKNOWLEDGEMENTS

In the name of Allah, the Most-Merciful, the All-Compassionate, Praise be to Allah, we seek His help and His forgiveness. May He send peace and blessings on Prophet Muhammad, his family and his companions.

I owe my deepest gratitude to my main supervisor Assoc. Prof. Dr. Rahmat Budiarto for giving me the opportunity to work with him in this very exciting area. I thank for his guidance, support and encouragement, without which this thesis would not have been possible. I am grateful to him for all the time and energy he spent in helping me improve my research.

Likewise, I am also grateful to my co. supervisor, En. Azlan bin Osman for his helps and comments during some events, Professor Dr. Rosni Abdullah as the dean of School of Computer Sciences USM for providing space to write this thesis. Professor Dr. Sureswaran Ramadass as the director of National Advanced IPv6 Centre for Excellence (NAv6) for his professional advice and supports.

I would like to thank to Directorate General of Higher Education of the Ministry of National Education of the Republic of Indonesia for supporting my study. Prof. Dr. Rahman Abdullah, M.Sc as Rector of Sultan Ageng Tirtayasa University (Untirta) for his support and allowing me to pursue my master at USM. Ir. Rimunarto, MT, Head of Electrical Engineering Department of Untirta for providing time to complete my master. May Allah make our department better in the future.

I also acknowledge and appreciate the love and support of my family (Ana, Maryam, Asiyah and Khadijah) for sharing every moment of living in Penang with me. Thank you for their understanding what I have been doing to leave them every day and night. Thanks also to my parents for their blessings and prayers to my success.

Last but not least, a special thank to my friends and colleagues, both inside and outside USM. The small groups of SABR (supri, abidah, bahareh and rafiza), that has been really patience to conduct our weekly discussion in order to enhance our knowledge. The group of Forkommi, that has given moral supporting to reach success in my live. May Allah blessing us to be always continue in His way.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATION

IP	Internet Protocol
RIRs	Regional Internet Registries
NAT	Network Address Translation
IPSec	IP Security
VoIP	Voice over IP
IETF	Internet Engineering Task Force
IPv6	Internet Protocol version 6
TCP/IP	Transmission Control Protocol/Internet Protocol
IPv4	Internet Protocol version 4
CRC	Cyclic Redundancy Check
FCS	Frame Check Sequence
BER	Bit Error Rate
CRC-32	Cyclic Redundancy Check 32 bits
CEH	CRC Extension Header
ISO/OSI	International Standard Organization / Open Systems Interconnection
TTL	Time to Live
RFC	Request for Comments
QoS	Quality of Service
PDUs	Protocol Data Unit
DMA	Direct Memory Access
MAC	Medium Access Control
GIEH	Generic IPv6 Extension Header

Hdr	Header
TLV	type-length-value
AH	Authentication Header
ESP	Encapsulating Security Payload
TCP	Transmission Control Protocol
UDP	Unit Datagram Protocol
ICMPv6	Internet Control Message Protocol version 6
EH	Extension Header
CPU	Central Processing Unit
HW	Hardware
FDDI	Fiber Distributed Data Interface
ATM	Asynchronous Transfer Mode
LAN	Local Area Network
WAN	Wide Area Network
RA	Router Advertisement
ARQ	Automatic Repeat reQuest
FEC	Forward Error Correction
ACK	Acknowledgment
d_{min}	Minimum Hamming Distance
IEEE	International Electric and Electronic Engineering
CRC-32E	CRC-32 bits used in Ethernet
CRC-32C	CRC-32 bits proposed by Castagnoli
CRC-32K	CRC-32 bits proposed by Koopman
iSCSI	Internet Protocol Small Computer System Interface
IANA	Internet Assigned Numbers Authority

DLL	Data Link layer
HSN	High Speed Networks
OWD	One Way Delay
IPDV	Inter Packet Delay Variation
RTT	Round Trip Delay
EDC	Error Detection Capability
HD	Hamming Distance
PT	Processing Time
ms	milliseconds

PENAMBAHBAIKAN PENGHANTARAN PAKET IPv6 MELALUI RANGKAIAN KELAJUAN TINGGI DENGAN MENCADANGKAN KEPALA TAMBAHAN CRC

ABSTRAK

Pemanfaatan kelebihan ciri-ciri IPv6 dan rangkaian kelajuan tinggi pada penghantaran paket IPv6 dipercayai akan membuat penghantaran semakin pantas. Seterusnya verifikasi dan penghasilan semula kod *Cyclic Redundancy Check* (CRC) di setiap router akan menjadi halangan. Tesis ini berusaha untuk mengurangkan pengiraan CRC di penghala dengan memanfaatkan kelebihan ciri-ciri IPv6. Kepala tambahan CRC *(CRC Extension Header)* dicadangkan sebagai kepala tambahan baru untuk melakukan semakan ralat di lapisan Rangkaian. Penjanaan kod CRC memerlukan satu polinomial penjana. CRC-32C dipilih sebagai polinomial penjana bagi CEH yang dicadangkan kerana ianya mempunyai tempoh pemprosesan berkurangan. Analisa telahpun dijalankan dengan membandingkan antara pengesanan ralat di lapisan Rangkaian menggunakan CEH dan pengesanan ralat di lapisan Pautan menggunakan *Frame Check Sequence* (FCS). Keputusan menunjukkan bahawa penghantaran paket IPv6 dengan CEH sebagai pengesan ralat mampu mengurangkan kelewatan rangkaian. Kelewatan rangkaian berkurangan 72% bagi paket-paket bersaiz kecil dan 66% bagi paket-paket bersaiz besar. Pengurangan kelewatan rangkaian pada penghantaran paket IPv6 disebabkan oleh ketiadaan pengiraan dan penjanaan semula kod CRC di setiap penghala. Tempoh pemprosesan paket IPv6 dengan CEH di pihak pengirim mahupun di pihak penerima lebih tinggi. Ini kerana penjanaan CEH lebih kompleks daripada penjanaan FCS. Walaubagaimanapun, tempoh pemprosesan hanya meningkat 15% secara purata. Peratusan ini amat kecil dibandingkan dengan peratusan pengurangan kelewatan rangkaian secara purata 68%. Analisa lain dilakukan untuk mengetahui kecekapan

penghantaran paket IPv6 tanpa pengawal ralat lapisan bawah. Ini dilakukan dengan cara meniadakan CEH dan FCS sebagai pengawal ralat di lapisan bawah. Oleh itu, pengawal ralat hanya dilakukan di lapisan atas iaitu pengawal ralat lapisan Pengangkutan. Keputusan menunjukkan bahawa tempoh pemprosesan paket IPv6 di pengirim mahupun penerima turun sehingga 85%. Walaubagaimanapun, pengawal ralat jenis ini hanya menggunakan *checksum* 16 bit yang tidak merangkumi seluruh kepala paket IPv6 sehingga menyebabkan keupayaan pengesanan ralat sangat rendah dibandingkan dengan CEH dan FCS. Ralat penghantaran yang terjadi pada kepala IPv6 yang tidak diliputi oleh *checksum* tidak akan dapat dikesan.

IMPROVING IPv6 PACKETS TRANSMISSION OVER HIGH SPEED NETWORKS BY INTRODUCING CRC EXTENSION HEADER

ABSTRACT

Utilizing the advantages of IPv6 features and high speed networks technologies on IPv6 packets transmission is believed will force the transmission to be faster. Thus, verification and regeneration of cyclic redundancy check (CRC) code in every router results in high network latency. This thesis attempts to decrease the network latency by eliminating CRC calculation in router using the advantage of IPv6 features itself. The CRC Extension Header (CEH) is introduced as a new IPv6 extension header to perform error detection in the Network layer. Generation of CRC code requires a generator polynomial. Thus, it is important to get a suitable generator polynomial for the CEH. The CRC-32C is chosen as a generator polynomial for the proposed CEH due to its less processing time. Analysis was done by comparing error control at the Network layer using CEH and error control at the Data Link layer using FCS (Frame Check Sequence). The result demonstrated that transmitting IPv6 packets with CEH as error control provide lower network latency. The network latency decreases by 72% for small packets and 66% for large packets. The decrease in network latency of IPv6 packets transmission is due to the elimination the CRC calculation and regeneration in every router. Processing time of IPv6 packet with CEH both in the sender and the receiver is higher than FCS because CEH generation uses more complex algorithm than FCS generation. However, the increase of average processing time at the sender and the receiver is only 15%. This percentage is very small compared to 68% reduction of average network latency. Another analysis was done to investigate performance of IPv6 packets transmission without lower layer error control. This is done by eliminating both the CEH and the FCS as lower layer

error control. Thus, error control is only conducted in upper layer which is Transport layer error control. The result obtained shows processing time of IPv6 packets both in sender and receiver decreases significantly (85%). However, the type of error control only uses 16 bits checksum that does not cover entire IPv6 main header which resulted in lower error detection capability compared to CEH and FCS method. Any transmission error occurring within the IPv6 header not covered by the checksum will be undetected.

CHAPTER 1

INTRODUCTION

1.1 Background

The explosive growth of Internet has brought extremely large number of IP (Internet Protocol) address consumption. The biggest explosion occurred in 2003 – 2005, where the growth reached 34 percent per year (Beijnum, 2007). The Internet Protocol Journal reported in September 2007 that 68 percent of IPv4 address has been allocated to RIRs (Regional Internet Registries), 14 percent is reserved for private use, multicast and special purposes and 18 percent of the address are unallocated. If the growth remains as stated, the remainder of unallocated address predicted will be fully depleted in 2010 (Huston, 2007). This scarcity of IP address becomes the main problem of the future Internet.

To overcome the IP address depletion problem, people deployed Network Address Translation (NAT). NAT is a technology that allows network with local address to communicate with global address (Internet). However, NAT solves IP depletion problem to some extent but has many disadvantages. It does not give much room to run security applications such as IPSec and renumbering. It also blocks several Internet applications especially those which requires two ways communication such as videoconferencing, online gaming and VoIP. Consequently, for long term Internet deployment, integrating IPv6 is the only option to overcome the address depletion problem. IPv6 is successor of the current widely used Internet Protocol, IPv4. It was developed by IETF (Internet Engineering Task Force) in the year 1995 (Deering and Hinden, 1995). The new version of Internet protocol was

designed to cover larger address space with 128 bits address field. However, the protocol not only overcome the address depletion but also brings a big package of benefits. The benefits include scalability, simple header format, mobility support, auto configuration, integrated quality of service, and support for real time application and more efficient on packet forwarding.

The advantages of IPv6 features may drive faster IPv6 packets transmission theoretically. IPv6 packets processing in a router must be more efficient due to simpler and fixed size of IPv6 header. In addition, transmission speeds always increases every time and advance technologies on high speed network including gigabit Ethernet and fiber optic has also been discovered. Unfortunately, there is an issue on the Internet protocol stacks (TCP/IP) that is used to transmit IPv6 packets currently. Infrastructure of the protocol suite does not support utilization of IPv6 features for IPv6 packets transmission optimally. Hence, IPv6 packet transmission is treated just like other packet transmission even though it brought many advantages. Furthermore, the transition process from IPv4 to IPv6 is moving at slow pace and people are lackluster to do it.

1.2 Issues on Error Detection Mechanism

A principle of IP network system is to transmit IP packet from one end point (source) to another point (destination). In order to transmit IPv6 packets from source to destination the packets may go through intermediate system that consists of routers. A router represents three of five layers of TCP/IP protocol suite: Physical layer, Data Link layer and Network layer. The process that a transmitted packet

needs to undergo at each router includes error detection code computation at Data Link layer and packet forwarding operation at Network layer.

TCP/IP protocol suite employs cyclic redundancy check (CRC) in the form of Frame Check Sequence (FCS) field to perform error detection in Data Link layer. Every router has to verify the CRC code brought by the frame received and regenerates a new CRC code to the frame that will be transmitted to the next node. This process is done to make sure the frame transmitted is free from transmission error along a certain hop. With high speed network availability and very low bit error rate (BER) medium, verification and regeneration of CRC in each router is time consuming task and increase the network latency. In fact, transmission error is almost zero in very low BER medium such as fiber optic (Tanenbaum, 2006). In addition, due to linearity of CRC code, bigger packet size requires more time to do the computation. Thus, error detection in Data Link layer (link by link error control) of each router is likely to become the source of bottleneck in the near future (Braun and Waldvogel, 2001).

1.3 Problem Statements

IPv6 is a new Internet Protocol which offers set of unique benefits as mentioned in Section 1.1. Unfortunately, IPv6 packets transmission still follows the traditional protocol stack within the TCP/IP suite. It fails to take full advantages of IPv6 features due to redundancy of error control mechanism at Data Link layer. Even though header checksum field was removed from IPv6 main header, error detection mechanism in Data Link layer is still an issue (Kay and Pasquale, 1996, Walma,

2007). The challenge is to find an error detection mechanism with low network latency for IPv6 packets transmission over high speed networks.

Based on the problem statement, the following are two main questions addressed by this research as follows:

1. What is the structure of IPv6 extension header to lower network latency of IPv6 packets transmission over high speed networks?
2. How to do error control at Network layer by utilizing feature of IPv6 packets on IPv6 packet transmission over high speed networks?

1.4 Research Objectives

In order to address the two research questions, the following are two objectives of this research:

1. To propose a new structure of IPv6 extension header to lower network latency of IPv6 packets transmission over high speed networks.
2. To propose a method to do error control at Network layer by utilizing feature of IPv6 packet on IPv6 packets transmission over high speed networks.

1.5 Scope and Limitations

IPv6 packets transmission over high speed networks is a wide area that involves many aspects of networking. This thesis focuses on decreasing network latency due to duplicate CRC verification and regeneration in every router on IPv6 packets transmission over high speed networks. Transmission error is low level error caused by the medium used. A medium could be anything that has very low bit error

rate includes copper and fiber optic. However, copper is the one that is available in the laboratory. This thesis uses 32 bits generator polynomial with minimum Hamming Distance 4 to generate CRC code for error control. It can detect all single bit error and 3 bits burst error.

The use CRC Extension Header to perform error control in Network layer to decrease network latency of IPv6 packets transmission is not common. It still has limitations on the size of generator polynomial and processing time of the first packet. In the future, it will not perform well if the size of IPv6 packet larger than the covered data length of the generator polynomial. Processing of the first packet introduces overhead at the sender and receiver. On system with lower speed processor, there is a high chance of packet drop to occur.

1.6 Research Methodology

This research has been conducted using a combination of theoretical analysis and experiment to study the performance of a new method to decrease network latency of IPv6 packets transmission over high speed networks. **The first step** is to investigate the reasons for high network latency on IPv6 packets transmission over high speed networks related to error control operation. General problems of error control schemes are duplicate CRC verification and regeneration that consume large amount of time to transmit IPv6 packets on error free medium. Result of the investigation can provide some ideas that could be proposed to reduce the network latency.

Second step is to formulate a new solution based on the ideas that can reduce the duplication of CRC computation and decrease network latency based on

theoretical result. The proposed solution is to utilize an IPv6 feature which is extension header to do error control in Network layer. This thesis proposed a new structure of extension header called CRC Extension Header (CEH). **The third step** is to develop a prototype of the proposed extension header and a method to do error control in Network layer utilizing the CEH. **The final step** of this research is to validate the prototype using experimental test-bed of IPv6 packets transmission over high speed networks (to be discussed in detail in Section 4.2). Results obtained from the experiments will be analyzed to justify performance of the proposed solution.

1.7 Thesis Contributions

In turn, this thesis contributes on IPv6 packets transmission over high speed networks as follows:

1. New structure of IPv6 extension header called CRC Extension Header (CEH) to decrease network latency of IPv6 packet transmission over high speed network.
2. A new error control method at Network layer instead of Data Link layer for IPv6 packets transmission over high speed networks.

1.8 Outline of the Thesis

This thesis is organized into six chapters. The organization of each chapter is outlined as follows:

Chapter 1 briefly outlined the significance of IPv6, its advantages and issues on error detection mechanism of the new Internet Protocol packet transmission. It is followed by problem statement, research objectives, scope and limitations and research methodology. Thesis contributions are also stated in this chapter.

Chapter 2 provides theoretical background of the thesis includes advantages of IPv6 network system including IPv6 packets format, larger address space and extension header. It is followed by investigating error control mechanism in the existing IPv6 packets transmission. Detail explanation about cyclic redundancy check (CRC) is also presented including the types of generator polynomial that usually used to generate CRC code. The rest of this chapter contains discussion of related works on duplicate CRC verification and regeneration.

Chapter 3 describes design of the proposed CRC Extension Header (CEH) as a new error control method at Network layer on IPv6 packets transmission over high speed networks. The design includes format, generation and computation of CEH. Advantages of the proposed extension header are also listed in this chapter.

Chapter 4 provides experiment of CEH to do error control in Network layer. It includes scenario of the experiment of IPv6 packets transmission with CEH, explains the parameters as well as the metrics used.

Chapter 5 presents result of the experiments of IPv6 packets transmission with CEH. This chapter also analyzes the result to justify the feasibility to use CEH as error control at Network layer. The performance of CEH and FCS methods will be compared.

Chapter 6 concludes of all chapters of this thesis. It also covers this thesis achievement. Future works of this area are also included in this chapter.

CHAPTER 2

THEORITICAL BACKGROUND AND RELATED WORKS

This chapter provides advantages of Internet Protocol version six (IPv6) as successor of existing Internet protocol (IPv4). It also discusses limitation of the current network infrastructure to transmit IPv6 packets over high speed networks that caused unoptimal usage of the new Internet Protocol features. Furthermore, the concept of IPv6 extension header is also presented as one of the important feature of IPv6. Error control on IPv6 network system is investigated to find out the root cause of inefficiency of the current error control mechanism. Subsequently, the fundamental concept of cyclic redundancy check (CRC) including algorithm and generator polynomial is presented in detail. This chapter concludes with the discussion of some related works on reducing duplicate CRC verification and regeneration in router and closed by chapter summary.

2.1 Advantages of Internet Protocol Version 6 (IPv6) Features

As explained in Chapter 1, IPv6 is an enormous Internet technology that has been developed as an evolutionary of the existing Internet Protocol, IPv4. As the future Internet technology, it has to meet the need of high speed data communication such as higher transfer rate, error free and real time application. To achieve the requirements, IPv6 has been designed with some improvements over the former Internet Protocol, IPv4. This section introduces enhancements of the Internet Protocol technology including new IP header format, large address space as well as extensibility of its extension header.

2.1.1 The IPv6 Header Format

In the ISO/OSI *(International Standard Organization/Open System Interconnection)* reference model, message that is created in Application layer moves down through many layers and encapsulated in each layer. In the encapsulation process, message is added by header or header and trailer. Header is information added to the unit of data to ensure it can reach the destination correctly and safely. In the case of IPv6, there is an IPv6 header which is added by Network layer in IPv6 network system that is standardized in RFC 2460 (Hinden and Deering, 1998) as shown in Figure 2.1. It has many improvements from the format of the current Internet Protocol which is IPv4 header. Some fields of IPv4 header that are infrequently used were dropped from IPv6 header, some of them that generally work were kept and some new features were added where the functionality was necessary such as flow label (Bradner & Mankin, 1995).

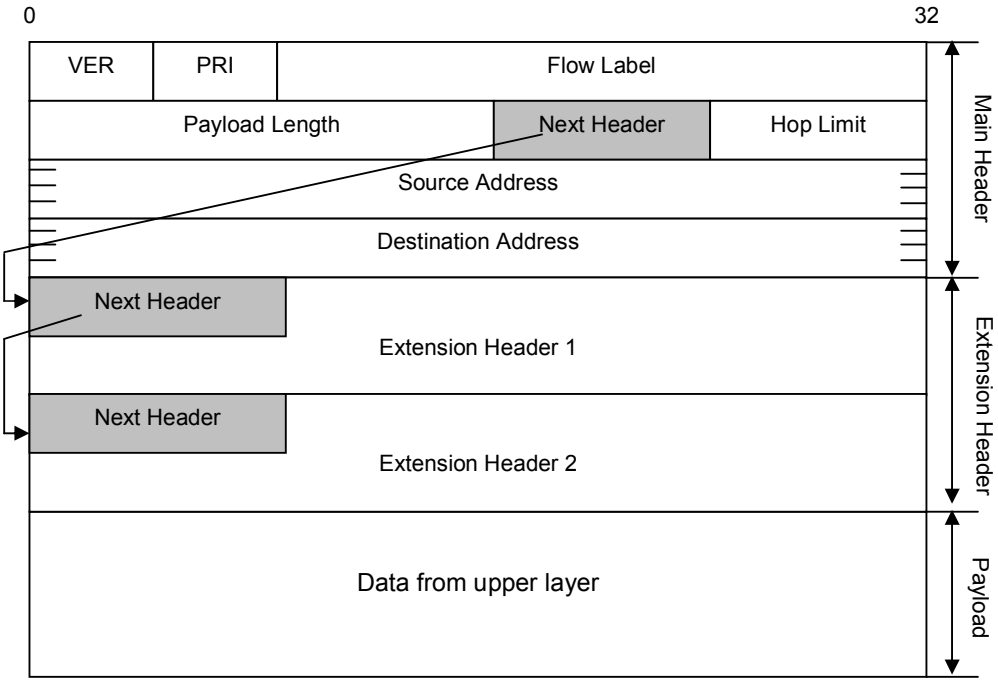

Figure 2.1 IPv6 Header Format with Extension Header

The first field of IPv6 main header is the 4-bits *version* field, this field indicates the protocol version. It will be used by operating system of receiver machine to forward the packet into the right stacks. In the case of IPv6, this field has a value of 06. The second field is an 8 bits *traffic class* field describing packet priority or its enlistment into a certain traffic class. This field is similar to the type of service field in IPv4. The following 20 bits is *flow label* that contains information that helps a router to determine the handling of each packet in the flow quickly. This flow label is the only new field introduced in the IPv6 header (Blanchet, 2005).

The 16 bits *payload length* carries the information on packet size including extension header. With 16 bits, it can identify the maximum length of packet of 2^{16} or 65,535 bytes. If the payload is bigger than 65,535 bytes, this field is set to zero and a special Jumbo Payload option is set up as extension header. The next 8 bits field is *next header* that defines either header or data type that follows the IPv6 main header. The 8 bits *hop limit* has the same meaning as the Time to Live (TTL) in IPv4. It is defined in units of second before the packet will be discarded. The value of this field decrease by one each time a router forwards the packet. The last two fields of Figure 2.1 before extension header field are *IPv6 source and destination address* field. These fields are the largest field in the IPv6 header, each field address is 128 bits long. Source address is the address of the source of the IPv6 packet. While destination address is the address of the recipient of the IPv6 packet. More detail of the address space will be discussed in Section 2.1.2.

There are advantages of the IPv6 header format over the former, IPv4 header. The IPv4 header format was introduced in RFC 791 (Postel, 1981) that has 12 fields

including options field. Its size varies depending on the options field required. The minimum size is 20 bytes, without options and the maximum header size with options is 60 bytes. Options field is variable length field that may be zero or more options in an IPv4 packet. Address fields size of IPv4 is 32 bits as depicted in Figure 2.2.

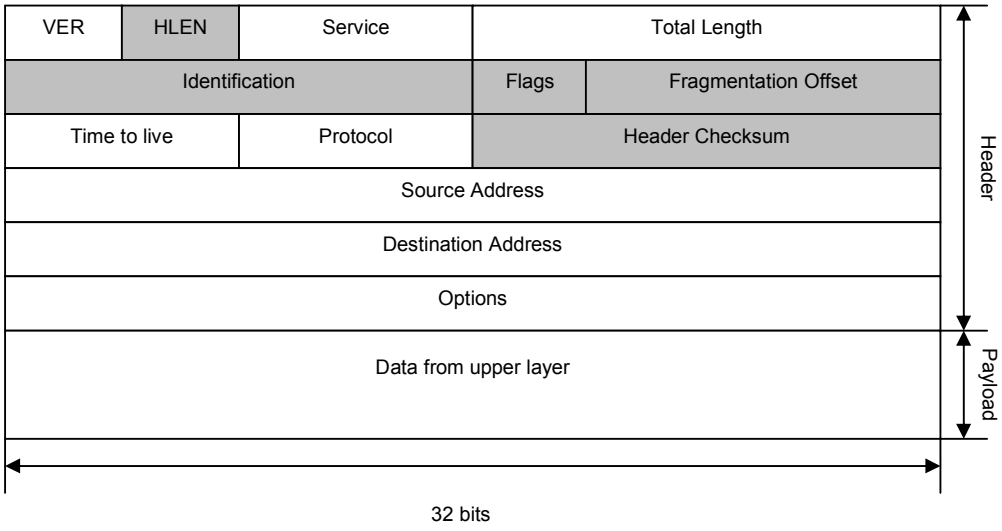

Figure 2.2 Format of IPv4 Header

Based on RFC 1752 (Bradner & Mankin, 1995), there are some important features of IPv6 header over IPv4 header include:

1. **Simplified header format**

   IPv6 is a major improvement of IPv4. Some infrequently fields used in IPv4 were removed which are Header Length (HLEN) field, Identification field, Flags field, Fragmentation field, and Header Checksum field (shaded fields in Figure 2.2). There are several new fields added such as flow label and traffic class. Total number of fields in IPv6 header is 8 fields instead of 12 fields in IPv4 and total size of the header is fixed at 40 bytes, while IPv4 header's

length varies from 20 to 60 bytes. Simpler header format and fixed size should result in faster packet processing in intermediate node.

2. **Expanded addressing**

   Address field is increased from 32 bits in IPv4 to 128 bits in IPv6 header. This allows all nodes in the world to be addressable and reachable by Internet connection. Larger address space also support more levels of addressing hierarchy and removing the need of Network Address Translation (NAT). It also provides easier allocation of addresses to downstream and improves end to end capabilities.

3. **Support of extension header**

   Extension header is similar with options header field in IPv4. However, options are part of IPv4 header and have restriction on size, while IPv6 extension header is part of IPv6 payload. Moving options fields from header allows for more efficient packet forwarding and greater flexibility for introducing new options in the future. The concept of extension header will be discussed in Section 2.1.3.

4. **Quality of service capabilities**

   A new capability is added to enhance the quality of service (QoS) by enabling labeling of IP packets belonging to a particular traffic flow. Sender could request special handling to the nodes by setting the flow label field value accordingly.

2.1.2 IPv6 Address Space

As mentioned in Section 2.1.1, the last two fields of IPv6 main header are source address and destination address. Each of these fields is 128 bits length. This

means IPv6 address field is four times bigger than IPv4 address field size. This size could support 2^{128} or 3.4 x 10^{38} addresses. Assuming the populations of human in the world are 7 billion, each person may obtain 4.8 x 10^{28} addresses. Until now, just 13 % of the addresses were allocated for global unicast address. Nevertheless, this number does not mean there are already used. Table 2.1 shows the current assignment of IPv6 address space (Blanchet, 2005)

Table 2.1 Current Assignment of IPv6 Address Space

Prefix (binary)	Start (hex)	End (hex)	Usage	Space used (%)
0000 0000	0000::	00ff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Uspecified, localhost	0.3
0000 0001	0100::	01ff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	0.3
0000 001	0200::	03ff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	0.6
0000 010	0400::	05ff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	0.6
0000 011	0600::	07ff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	0.6
0000 1	0800::	0fff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	3
0001	1000::	1fff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	6
001	2000::	3fff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unicast global, 6to4, Anycast	13
010 110	4000::	dfff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	60
1110	e000::	efff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	6
1111 0	f000::	f7ff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	3
1111 10	f800::	fbff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	1
1111 110	fc00::	fdff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unique-local	0.6
1111 1110 0	fe00::	fe7f:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Unassigned	0.2
1111 1110 10	fe80::	febf:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Link-local	0.1
1111 1111	ff00::	ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff	Multicast	0.3

The large address space introduces some advantages of the protocol as follows:

1. All nodes in the world not only computers but also other equipments may be addressable and reachable by IPv6 network. It provides more flexibility in assigning IPv6 addresses by tying them to an interface rather than a node (Fineberg, 2005).

2. IPv6 allows a single interface to have multiple IPv6 addresses of any type (unicast, anycast, and multicast) or scope (local and global). It makes the task such as renumbering and multi homing easier (Hinden and Deering, 2006).
3. Network Address Translation (NAT) is no longer needed because all nodes in the world may get more than one IPv6 address.
4. More levels in addressing hierarchy can be provided, easier allocation of addresses to downstream, and global routing table is more scalable.
5. It enables easier addressing plans and decreases network management cost. It will save time and budget.

2.1.3 IPv6 Extension Header

An IPv6 packet consists of IPv6 main header, extension header and upper layer data. The format of the packet including extension header was shown in Figure 2.1. The extension header is placed after destination address field before data from upper layer field. This section discusses the advantages of IPv6 from extension header point of view and the concept on it will be investigated in Section 2.3. This field is similar with the option field in IPv4 as depicted in Figure 2.2.

The existence of the extension header in IPv6 features gives many advantages in terms of IPv6 packets transmission.

1. All extension headers except hop by hop extension header will not be processed by the routers. This will increase routing performance.
2. The IPv6 packet may add many extension headers and these are not limited to 40 bytes as in IPv4 options.

3. New extension header can be added incrementally without any impact on current implementation.
4. An IPv6 packet may bring more than one type of extension header.

2.2 Limitation of IPv6 Packets Transmission

The layering concept in computer communication aims to decrease networking complexity. In the concept, every layer is separated from other layers and not dependent each other. Hence, the improvement of Network layer protocol has only small impact to other layers. Transmission of IPv6 packets still follows the traditional protocol stack which is TCP/IP *(Transmission Control Protocol/Internet Protocol).* A typical communication of two computers in TCP/IP protocol stacks is illustrated in Figure 2.3.

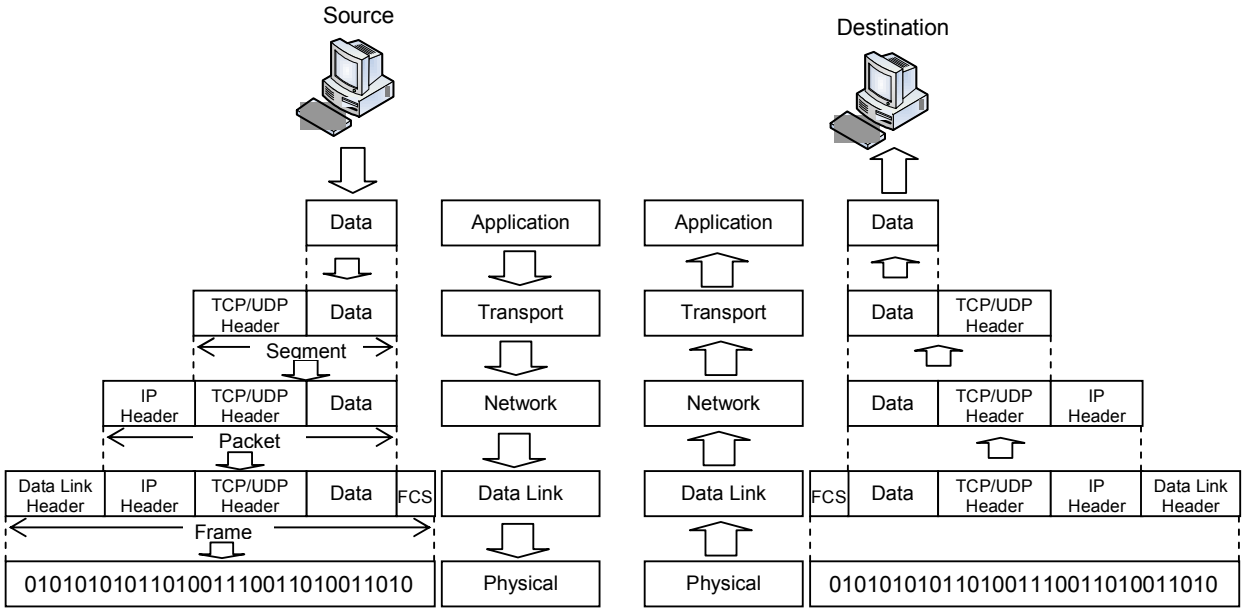

Figure 2.3 General Communications of Two Computers in TCP/IP Protocol Stacks (Cisco, 1999)

Based on Figure 2.3, at the source computer, IPv6 data is generated by Application layer. The data is transmitted in the form of PDUs *(Protocol Data Unit)* down to the Physical layer and then to the transmission medium. Data pass through the medium in the form of bits series. PDU contains data and header from upper layer, it has own name in each layer, segment in Transport layer, packet in Network layer and frame in Data Link layer. At Data Link layer, data is encapsulated with not only header but also a trailer in the form of Frame Check Sequence (FCS) field. FCS contains cyclic redundancy check (CRC) code to ensure the data is free from transmission error.

At the destination part, the IPv6 data is captured by Data Link layer and reverse operations compare to the source part will be performed. Data is de-capsulated by releasing data link header and verification of the FCS to detect transmission error is conducted. If the data contains any error, it will be discarded and it will wait for retransmission, otherwise it will forward the data to upper layer until it reach the Application layer of destination computer. All the process described above occurred only when the two computers are located within the same network.

If the destination computer is not located in the same network as the source computer (see Figure 2.4) the frame will be firstly transmitted to the interconnecting devices (routers). In a router, there are two lower layer processes which are in incoming port and outgoing port of the router. At the incoming port of router, the frame passed through the Data Link layer and it will be checked for transmission error. If there is an error, the frame is discarded, otherwise it will be passed to

Network layer in order to determine which network the packet is going to be forwarded.

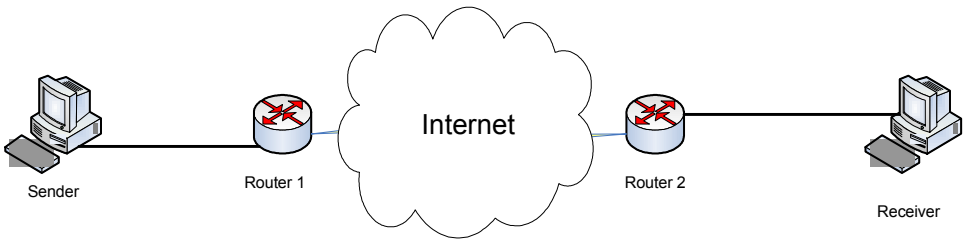

Figure 2.4 Communications through Interconnecting Devices

After Network layer processing is done, the IPv6 packet will be encapsulated in Data Link layer of the outgoing port. In terms of link to link communication, router behaves as receiver on incoming port and sender on outgoing port. The Data Link layer header and trailer field will be modified in every forwarding node. Figure 2.5 demonstrates the communication process traversing through sender, interconnecting devices and receiver.

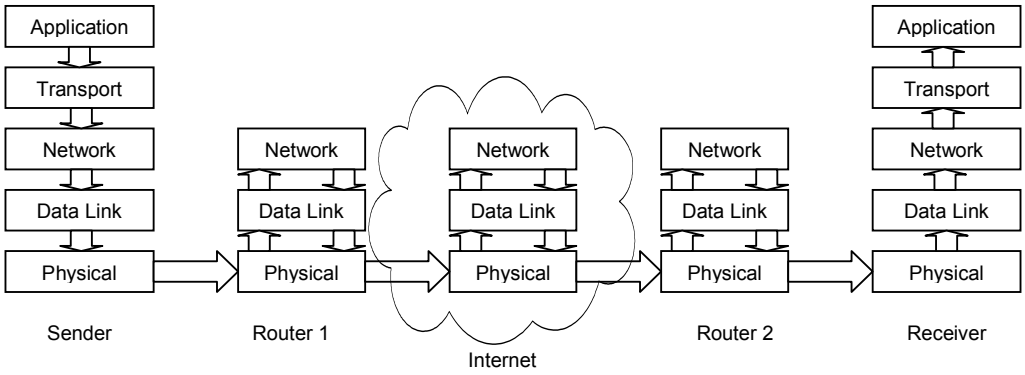

Figure 2.5 Packet Processing Traverse Interconnecting Devices

Based on the IPv6 packets transmission process, there are two major problems that limits network performance, they are overhead and duplicate CRC calculation. Overhead is caused by many encapsulation processes which do not

involve non data processing. Duplicate CRC calculation occurred in every intermediate node. The two limitations will be discussed in the following section.

2.2.1 IPv6 Network System Overhead

Overhead is the time required to perform processing something that is not original data. The IPv6 packets are transmitted through layers with encapsulation and de-capsulation process. Wook (2007) classified overhead at TCP/IP protocol stacks into three classifications: host overhead, NIC (network interface card) overhead, and link overhead. Host overhead is time spent to perform protocol stacks in the operating system's kernel and driver device. Kernel usually implements two layers of TCP/IP which are Transport layer and Network layer. NIC overhead is generated by NIC and corresponds to Data Link layer. Link overhead is the time spent to transfer a packet to and from the transmission medium. The author did experiment with Marynet system and showed NIC overhead of Data Link layer was the dominant overhead for most data size. The host overhead is constant while link overhead depended on the transmission medium used.

NIC overhead consists of per-DMA (Direct Memory Access), per-packet and per-byte overheads. DMA is the way to move data between host and NIC memories. Overhead on DMA is caused by DMA initialization and the value is linear depending on linearity of network buffer. Per-packet overhead is time spent to generate a frame in Data Link layer. A frame consists of data from upper layer, data link header and trailer. The trailer (FCS) must be generated by NIC because the CRC code is calculated from the frame itself. CRC calculation is believed to be the most

computing intensive and time-critical functions that may impede the processing speeds (Lu, 2003).

Kay and Pasquale (1996) profiled processing overhead in TCP/IP protocol stacks. They categorized the major processing overhead in network software as checksum computation, data movement, data structure manipulation, error checking, network buffer management, operating system functions and protocol specifics processing. They concluded that the largest bottleneck to achieve high throughput is computing checksums. This is because checksum computations touch each byte of the message. The time consuming will increase with message size.

2.2.2 Duplicate CRC Calculation

CRC is cyclic redundancy check that is used to detect transmission error. It is usually implemented in lower layer which is the Data Link layer. CRC is a code that is generated from the whole frame including the MAC address, Ethernet type and upper layer data. In a router, it needs to calculate the CRC code twice. First, calculation is done in incoming port to verify whether there is transmission error on the received frame. It decides the erroneous frame is discarded and the correct one is forwarded into Network layer to do Network layer operation. Second, CRC calculation is also done at outgoing port for each frame received from Network layer. This new CRC code is different than the previous frame. This is because the frame's Data Link header and few fields of IP packet have changed. The calculation process can be seen in Figure 2.6 (Weidong Lu, 2004).

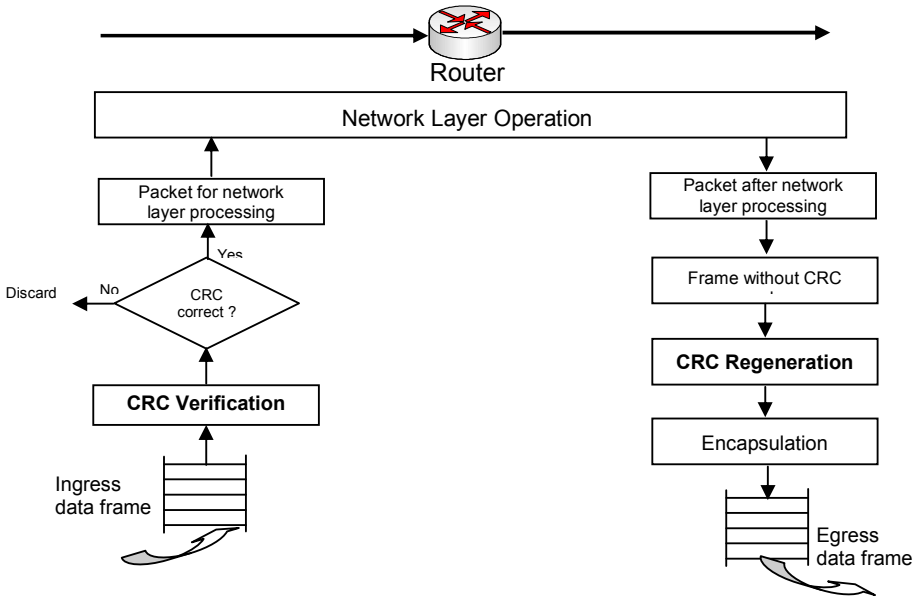

Figure 2.6 CRC Calculations in an Interconnecting Device (Router)

Communication between two computers through Internet most likely will involve more than one router in the journey. Each router needs to calculate and regenerate the CRC code. For a simple network in Figure 2.5, there are four CRC verifications and four CRC generations as shown in Figure 2.7. Unfortunately, the object of the frame's CRC calculation is similar with only minor changes on the frame. Those processes introduced processing overhead in IPv6 network system.

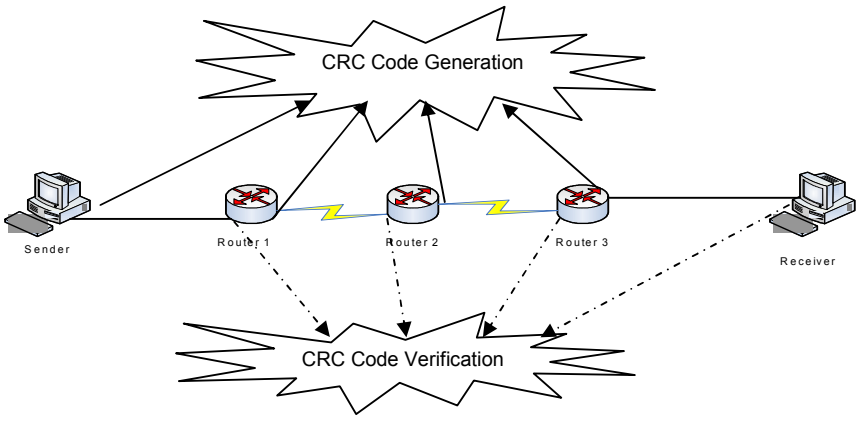

Figure 2.7 Number of CRC Calculation for Figure 2.5

The number of CRC calculation will increase with increasing quantity of interconnecting devices in the network. The number can be formulated as follow

$$n(x) = 2 + 2\sum R(x) = 2(1 + \sum R(x)) \quad (1)$$

Where *n(x)* is number of CRC calculation for both verification and regeneration, *R(x)* is number of router in the network. The number 2 on the first of Equation (1) is CRC calculation both in sender and receiver. Thus, total of CRC calculation in a network with 3 routers in Figure 2.7 is 8.

Assuming ability of CRC code generation and verification of all devices are identical, the total time for CRC calculation can be formulated as Equation (2). Verification and generation of CRC code are two different tasks which resulted in two different amount of time to perform. In Equation (2), the two amount of time are separated as sender time and receiver time.

$$t(x) = \frac{n}{2}\{t_s(x) + t_r(x)\} \quad (2)$$

Whereby *t(x)* is total times spend to verify and regenerate CRC code in the entire network, $t_s(x)$ is time needed to generate CRC code at sender and outgoing port of router, $t_r(x)$ is calculation time at receiver side and incoming port of router. *n* in Equation (2) is total number of CRC calculation based on Equation (1).

2.3 The Concept of IPv6 Extension Header

IPv6 header consists of main header and extension header. The concept of IPv6 extension header follows the concept of option field in IPv4. According to RFC 2460 (Hinden and Deering, 1995), concepts of IPv6 extension header are as follow:

1. IPv6 extension header is optional and it is placed between main header and upper layer header in an IPv6 packet (see Figure 2.1).

2. There are numbers of extension header in IPv6, each identified by a distinct next header value (see its list in Table 2.2).
3. An IPv6 packet may carry zero, one, or more extension headers. Each is identified by the next header field of the preceding header or extension header.
4. With an exception of hop by hop options header, extension headers are not processed by any node along a packet's delivery path, until the packet reaches destination node.
5. Extension headers must be processed strictly in the order they appear in the packet.

2.3.1 Format of IPv6 Extension Header

RFC 2460 does not specify the standard of IPv6 extension header format. Each extension header has a specific format depending on their services required. The only similarity in all extension headers is next header field that identify the next following extension header. Krisnan, et al. (2008) proposed a uniform format of IPv6 extension header. They introduced Generic IPv6 Extension Header (GIEH) with generic format as Figure 2.8 that contains four fields as follows:

Next header: 8 bits selector to identify the type of extension header immediately following this extension header. This field has the values as listed in Table 2.2.

Hdr. Ext. Length: 8 bits unsigned integer that indicates the length of the extension header in 32 bits units.

Specific Type: 8 bits unsigned integer that is the actual IPv6 extension header type. This is allocated from IANA.

Header Specific Data: this is the core of extension header that contains specific data as the requirement of the extension header. The length is variable and must be padded as needed in order to ensure that the whole extension header is a multiple of 8 bytes long.

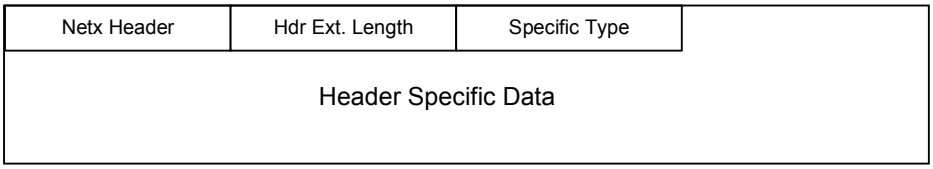

Figure 2.8 the Format of Generic IPv6 Extension Header

2.3.2 Current Types of IPv6 Extension Header

There are numbers of IPv6 extension header that have been specified by IETF (Internet Engineering Task Force) on some RFCs which are:

a. *Hop-By-Hop Options* is a variable size extension header which has options field that need to be examined by all devices on the path. It is specified in RFC 2460 and identified by a next header value of 0 in the IPv6 main header. The options field contains one or more TLV-encoded (type-length-value) options including option type, option data length and option data. This extension header should be placed in the first order of extension header chain because it needs to be processed in every node.

b. *Destination Options* is extension header that is used to carry information that need be processed only by destination node. It is specified in RFC 2460 and recognized with 60 by the preceding header. There are two types processing of destination node for this extension header depends on the option field. First, the options that need to be processed by the first destination that appear

in the IPv6 destination address field and subsequent destinations listed in the routing header. This type of option must be placed before routing header. Second, the one that contains options to be processed only by final destination of the IPv6 packet. It is placed in the last order of extension header chain.

c. *Routing Header* is a method to specify the route for an IPv6 packet, which node to be visited along the way. It is identified using value of 43 by next header field of the preceding header. This type of extension header is used with mobile IPv6 as specified in RFC 3775. In the case of mobility header, it is recognized with the next header value of 135.

d. *Fragmentation Header* is used to send IPv6 packet larger than link layer MTU by the sender. The packet is fragmented by the sender into smaller packet and sends them separately until they reach the destination node identified by destination address field. Fragment packets are reassembled into original un-fragmented form in the destination node. This extension header is identified by next header value of 44 of the immediately preceding extension header.

e. *Authentication Header (AH)* contains information used to verify the authenticity of most parts of the packet that is used to provide connectionless integrity and data origin authentication for IPv6 packets and to provide protection against replays (RFC 4302). It may be applied alone or together with Encapsulating Security Payload (ESP) extension header. Its existence is identified by the value of 51 of next header preceding extension header.

f. *Encapsulating Security Payload (ESP)* carries encrypted data for secure communication that is bale to provide confidentiality, data origin

authentication, connectionless integrity, an anti-replay service, and (limited) traffic flow confidentiality (RFC 4303). It is recognized by next header field of previous extension header with the value of 50.

2.3.3 Processing of IPv6 Extension Header

When an IPv6 packet has more than one extension header, they appear as a chain with the order recommended as listed in the Table 2.2. Based on Table 2.2, each extension header mostly appears once except for Destination option header which may emerge twice (number 3 and 8) at an IPv6 packet. As explained in the previous section, it has two types of destination which are final destination and the one listed in routing header. In case of IPv6 tunneling over IPv4, each packet has own extension header and use the same order listed.

Table 2.2 Recommended of Extension Header Order in IPv6 Packet

Order	Header Type	Next header value
1	Basic IPv6 header	-
2	Hop by hop options	0
3	Destination options (with routing options)	60
4	Routing header	43
5	Fragment header	44
6	Authentication header	51
7	Encapsulation security payload header	50
8	Destination options	60
9	Mobility header	135
	No next header	59
Upper layer	TCP	06
Upper layer	UDP	17
Upper layer	ICMPv6	58

Hop by hop options extension header is the only extension header which should be processed in every node along with packet's delivery path. This is because,