CRC To Support Internet Protocol–IPV6

[1]Shahsad. A. S, [2]Dr. K.V Purushothaman
[1]Travancore Engg. College, Kollam, Kerala, India
[2]Heera College Of Engg. & Technology, Panavoor, Kerala, India

Abstract
IPV6 is a new technology that has been developing as a replacement for existing Internet protocol, IPV4. IPV6 was designed with same additional features to the IPV4. In order to meet the requirement of fast growing internet, we must switch from IPV4 to IPV6. IPV6 is a datagram protocol same as IPV4 having 128 bit addresses instead of 32 bit in IPV4. An IPV6 packet transmitted through the internet has to pass through many routers along the network. Therefore chances of occurrence of error increases. So error detection/ correction method is to be utilized for the data. CRC is one of the most popular error detection mechanism. Here CRC is used to generate a checksum over the modified field at transmission nodes instead of calculating CRC over the whole frame. In the modified it is 1 byte hop limit only, so there is no need to calculate CRC for the whole frame in intermediate nodes. By calculating CRC only for the modified field of a frame helps to increase the processing time at the sender, receiver and also the overall processing time of an IPV6 packet.

Keywords
IPV6, IPV4, CRC, Parallel CRC, IPV6 Datagram

I. Introduction

In last few years, Internet undertook an unexpected growth. It become a primarily technology in the human life today. The explosive growth of internet cause address depletion problem. The current internet protocol is Internet Protocol version 4 (IPV4). Due to the scarcity of address the IPV4 protocol cannot satisfy all the requirements of always expanding Internet. The solution is to switch from IPV4 to IPV6. IPV6, Internet Protocol Version6 is the next generation network layer protocol designed for replacing IPV4, which is presently existing. The main advantage of IPV6 over IPV4 is its large address space, ie. 128 bit addresses in addition to the users.

IPV6 is a 128 bit address scheme, so the possible number of unique address is 3.4x1038. Its a datagram protocol same as IVP4. Here also data transmission in done as packets. An IVP6 packet transmitted through the internet has to pass through many routers along the network. Hence the chance of occurrence of error is more. So, some error detecting mechanism must be used for the integrity of data. Most networking protocols use CRCs to verify data whether any error was occurred. CRC performs a mathematical calculation on a block of data and returns a code or number about the content of the data. The resultant number uniquely identifies that block of data. The unique number is used to check the validity of data.

IVP6 packet transmission uses traditional TCP/IP for CRC calculation and regeneration in data link layer. Most networking protocols use CRCs to verify data whether any error was occurred. CRC performs a mathematical calculation on a block of data and returns a code or number about the content of the data. The simplest way to calculate CRC is done by dividing the data word M(x) by generator polynomial G(x) using modulo-2 division.

II. Internet Protocol Version 6- IPV6

IPV6 in a datagram protocol same as IVP4. The main reason for designing a new internet protocol (IPV6) is to increase the number of addresses. In present scenarios the current annual growth rate is 51%, so 63 new host per minute, 11 domains per minute or 10, 000 ISP worldwide are added to the network. ISO/OSI reference model can also used for IPV6 packets. IPV6 is a 128 bit address scheme instead of 32 bit scheme in IPV4 the possible number of unique address in IPV6 is 3.4x1038. It uses LAN better than IPV4 because no ARP is used, only Neighbor discovery protocol is used. Another feature is mobility that is each user is provided with two IP addresses, one permanent and one dynamic, used in the roaming only. In real time application priority is important. In IPV6, 4 bit priority is provided in IPV6 header and also 16 different types traffic priorities.

A. IPV6 Data Gram

An IPV6 packet comprises of header and upper layer payload as shown in fig 1. The IPV6 header consists of main header having a fixed size of 40 bytes and an extension header with optional size.

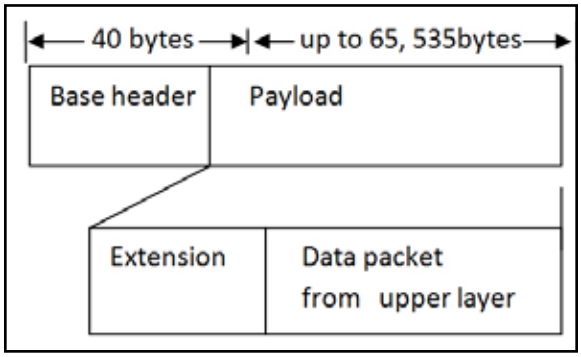

Fig. 1: IPV6 Packet

In IPV6 header, 32 bytes for source and destination address and remaining 8 bytes are used by 6 additional fields as shown in fig. 2. In IPV4 its 16 bytes, used by 12 additional fields. So in IPV6, header fields are less, hence more efficient processing and speed can be achieved.

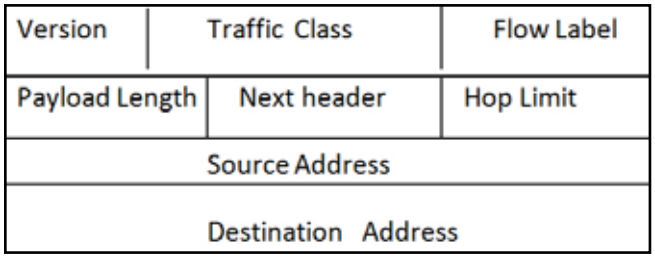

Fig. 2: IPV6 Header

III. Cyclic Redundancy Check -CRC

CRC is the most popular method used to detect communication errors caused by noise channels. The CRC performs a mathematical calculation on a block of data and return a code or number about the contents of that data. The resultant number uniquely identities

that block of data. The unique number is known as checksum which is used to check the validity of data.

Using CRC, we can find all single and double bit errors, also for almost all bits errors. All CRCs employ arithmetic over the finite field. CRC arithmetic is referred to as polynomial arithmetic modulo-2. The bitwise operator XOR is equivalent to adding or substracting. To compute CRC of a message, another polynomial called the generator Polynomial G(x) is chosen, G (x) should have a degree greater than zero and less than that of the message M(x). The generator polynomial used have is CRC-32 standard, ie

$G(x)=x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^{8}+x^{7}+x^{5}+x^{4}+x^{2}+x+1$

CRC 32 is a 32 bit polynomial that will detect all errors that spam less than 32 contiguous bits and all 2 bit errors less than 2048 bits apart.

In general, an n-bit CRC is calculated by representing the data stream as a polynomial M(x), multiplying M(x) by x^n (where, n is the degree of the polynomial G(x). The resulting remainder is appended to the polynomial M(x) and transmitted. The complete transmitted polynomial is then divided by the same generator polynomial at the receiver end. If the result of this division has no remainder, there are no transmission errors.

A. CRC Implementation

1. Serial Implementation

It takes more number of clock cycles to give the output and almost cost is very high. Hence we are going for parallel implementation of CRC, which gives the output with few clocks cycles.

2. Parallel Implementation

A parallel implementation operates on multiple bits of the data stream per clock cycle. So the number of clock cycles reduces and hence the overall speed increases.

B. CRC Verification and Recalculation

When a data transmission occurs over internet, First the CRC verification is performed in the data link layer to detect errors, If the verification fails, the frame is discarded and frame will be retransmitted by the source otherwise frame will be passed to network layer. After the verification, the frame is passed to the upper layers for further process. Then CRC code is calculated and is appended to the frame and the frame is forwarded through the physical layer to its next hop.

IV. Existing and Proposed System

In computer network, routers have the important role in the packet forwarding process in order to ensure that each packet reaches its correct destination. In general a router needs to decide to which node a packet has to be forwarded. This decision is made in the network layer of the router. However before the packet reaches the network layer, it must pass through the data link layer, where error detection process taken place. The data link layer always verifies the CRC code attached with the received frame to ensure that the frame from the previous hop is error free. Only the correct frame will be delivered up to the Network layer for forwarding.

An IPV6 packet transmitted through the internet has to pass through many routers along the network. When we look into the IP packet processing in an intermediate system, we can see that only few bits of the packets transmitted will change. Most of the field remains same. In IPV4, only 3 bytes (TTL- time to live one byte and header checksum field two bytes) are changing while forwarded to an intermediate mode.

In IPV6 packet, there is only a byte which is changing that is hop limit field which is decreasing by one after forwarding. As only a small change in IP header is occurring there is no need to check the overall packet. In the existing system, CRC is calculated over the whole frame. Thus the CRC processing time will be very high for IPV6 packet. In order to reduce the processing time, we proposed a system for CRC calculation where CRC computation is done only in the modified field of the frame. Thus CRC computations become faster and thus overall CRC computation of an IPV6 frame is only 15 bytes. This technique is effective to reduce CRC time processing in a router.

V. Experiment

To verify this proposed concept an experiment was done. Here error detection was simulated using three methods. First method calculates the CRC for all the frame. Second method calculates the CRC for only the modified bytes in the frame. Third method CRC code is appended as extension header.

Here, we consider five nodes as shown in fig. 3, Sender, router 1, internet (also a router), router 2 and receiver. All the nodes are PCs which were configured as IPV6 network. Sender is a PC that in installed with a program that is capable of generating IPV6 of the above three features

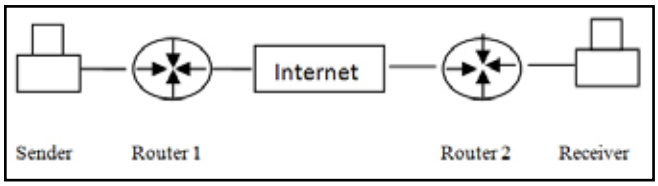

Fig. 3: Packet Transmission from Sender to Receiver

In the first two methods, the sender generates the IPV6 packets. The packet is then encapsulated by data link layer by adding header and trailer. Thus IPV6 frame is formed. Each intermediate node will receive the packet, process it and computes for CRC code to detect errors. In the second method, the CRC calculation is done only for the modified fields. Thus CRC computations become faster and the CRC processing time is also reduced.In third method, hop limit is to be separated from the IPV6 packet first before calculating the CRC code and this CRC code has to be inserted into IPV6 packet. So the processing time of this method is higher than the other two methods.

VI. Experimental Results

In the experiment, we measure processing time packet error rate and packet loss. Processing time is the time required to process an IPV6 packet. Packet error occurs when the receiver detects an error. If the packet does not reach the receiver in time, then it is considered as loss.

Here the processing time at the sender side, receiver side and overall processing time for an IPV6 packet is compared for the three methods. The sender's processing time is the time required to generate an IPV6 packet, including CRC code generation. While the receivers processing time is the time required to verify the IPV6 packet received including CRC code verification. An IPV6 packet require a certain amount of time to process at the sender as well as at the receiver. The processing time at the sender obtained uses different methods is shown in Table 1.

Table 1: Senders Processing Time

Packet size (bytes)	64	128	256	512	1024
(i)CRC done for whole frame	0.0683	0.692	0.761	0.801	0.832
(ii)CRC done for modified field	0.663	0.678	0.721	0.791	0.812
(iii)CRC in extension header	0.698	0.701	0.787	0.821	0.881

Table 1, shows that the processing time of the packet at the sender increases with the increase in packet size. This is because CRC code generation was done byte per byte of packet. Similarly, the receiver processing time is shown in Table 2. Here also the processing time is higher for larger packets.

Table 2: Recievers Processing Time

Packet size (bytes)	64	128	256	512	1024
(i)CRC done for whole frame	0.501	0.524	0.567	0.598	0.601
(ii)CRC done for modified field	0.471	0.492	0.521	0.538	0.551
(iii)CRC in extension header	0.531	0.581	0.601	0.653	0.688

The comparison of overall processing time for different packet size is listed in Table 3. The comparison shows the processing time of packets with CRC in extension header is higher than the other two. The lowest processing time is obtained in the second method, where the CRC code is calculated only on the modified field (hop limit) which is the proposed method.

Table 3: Overall Processing Time

Packet size (bytes)	64	128	256	512	1024
Processing time (in milli second)					
(i)CRC done for whole frame	1.281	1.321	1.381	1.421	1.531
(ii)CRC done for modified field	1.121	1.167	1.250	1.361	1.412
(iii)CRC in extension header	1.321	1.413	1.452	1.561	1.621

Table 3, shows that the processing time is lower in the second case, where CRC is calculated only to the modified field of the frame. Also table shows that the processing time increases as the packet size increases. Since IPV6 packets are to be processed very fast, the reduction in the processing time by the proposed method will be very much beneficial for IPV6 packet transmission through internet.

Packet error rate comparison of IPV6 packet under these three methods shows that there is no erroneous IPV6 packets on all transmission methods.

A packet considered as lost when it does not reach the destination in time. Here also processing time plays a key role. If processing time is high the packets will not reach the destination in time. So that packet may be considered as a lost packet. So high speed processors is needed in the third method where CRC code is in the extension header. Packet loss of IPV6 packet on the proposed method using Intel Pentium processor is shown in Table 4.

Table 4: Packet Loss During IPV6 Transmission

Packet size (bytes)	64	128	256	512	1024
Packet Sent	162318	93464	55412	28408	16905
Packet received	162314	43463	55411	28408	16904
Packet Loss	4	1	1	0	1

VII. Conclusion

Here we proposed a new approach to fasten the CRC calculation of an IPV6 packet by calculating the CRC code only for the modifying field of the frame not for the whole frame. The processing time of an IPV6 is very small comparing to the existing methods. Based on the results, we can increase the speed of packet processing at every node thereby providing high-speed IPV6 packet transmission through Internet

References

[1] Jiang, Zhangzhen Shenzhen,"Methods and system and devices for IPV6 datagram transmission in the Ethernet", Europeans Patent Application, 153 EPC, June 2009.

[2] D. Liu U. Nordquist, T. Henriksson,"CRC Generation for protocol Processing", Norchip 2000, Turki, Finland, pp. 288-293.

[3] S. Hagen,"IPV6 essentials", Oriently, July 2002.

[4] Andrew. S Tanenbaum,"Computer networks", Further edition, prentice hall of India, New Delhi, 2006.

[5] Weidong Lu , Wong S,"A fast CRC update implementation", pp. 113-120, [Online] Available: http://www.ce.et.tude LFT nl

[6] F. Braun, M. Waldvogel,"Font Incremental CRC updates for IP over AIM networks", IEEE workshop on high performance switching and Routing 2001.

Shahsad. A.S received his B.E degree in Electronics and Communication Engineering from Madurai Kumaraj University, Madurai, India, in 2003, the M.E Degree in Communication systems from Anna University, Chennai, India in 2005 and presently doing research work in networking under Kerala University, Kerala, India. He was lecturer in Electronics in 2005, Asst. Professor in 2009 and Assoc. Prof in 2011 with Kerala University. His area of interest include networking, communication techniques and television engineering.

Purushothaman K V received his BSc (Engg.) degree in Electrical Engg., from Kerala Univ., India in 1967, the M.Sc. (Engg) degree in Applied Electronics from Madras University, India in 1973, and the Ph.D degree in Microprocessor applications in Power Electronics from Madras University, India in 1984. He was a lecturer in Electrical Engineering in 1967, Asst. Professor in 1978 and Professor in ECE in 1982 with Kerala University. His areas of research include Power electronics, Microprocessor applications and Electronic Circuits.