Original Article
ISSN (Online): 2456-8651

International Journal of Engineering Science Technologies
November-December 2022 6(6), 1– 8

PERFORMANCE EVALUATION OF BANDWIDTH OPTIMIZATION IN IPV6 HEADER COMPRESSION

Onifade, Isaac [1], Oluwaranti, Adeniran [2], Iwayemi, Ayodeji [3]

[1] Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria
[2] Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Osun-State, Nigeria
[3] Department of Computer Engineering, Federal Polytechnic, Ile-Oluji, Nigeria

Received 02 October 2022
Accepted 01 November 2022
Published 18 November 2022

Corresponding Author

Onifade, Isaac, isafade2003@gmail.com

DOI 10.29121/IJOEST.v6.i6.2022.399

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

ABSTRACT

The development of Internet Protocol version 6 (IPv6) came with a larger header size of 40-Byte compared with the 20-Byte of Internet Protocol version 4 (IPv4) which would have made IPv6 protocol a bottleneck in data transmission over the global network. Practically, given the same user data transmitted over the same physical-layer network medium in both IPv4-enabled and IPv6-enabled networks, the network speed would be slower in the IPv6 setup compared to the IPv4 scenario, but network bandwidth is optimized with the in-built header compression technique in IPv6 which makes it advantageous over IPv4 and being deployed in the recent Operating Systems and networking devices including switches and routers. This performance evaluation is focused on the effect of varying header sizes on a packet transmission by determining both maximum packet throughput and header overhead percentage. These metrics were determined after the simulation of the packet transmission in MatLab vR2021a. The results were then validated with a simple ping command in a Peer-to-Peer (P2P) network connection between two Windows 10 Computer Systems using Ethernet Category 6 cable. The output shows that despite the higher header size in IPv6, data transmission using the protocol is at faster rates, and this is due to the inherent header compression feature in IPv6.

Keywords: Internet Protocol, Data, Network, Transmission, Header

1. INTRODUCTION

Header compression is one of the techniques in data transmission to optimize available network bandwidth which was incorporated into the development of IPv6 protocol. These headers can be compressed, often saving more than 90%, which reduces bandwidth usage and makes efficient use of expensive resources. Other significant advantages of IP header compression include less packet loss and accelerated interactive response. IP header compression, in its simplest form, is the process of reducing the size of extra protocol headers before sending them across a link and restoring them to their original size upon receipt at the other end of the link.

Due to redundant header fields in both subsequent packets of the same packet stream and header fields of the same packet, it is possible to compress the protocol headers. Bormann (2002). In order to meet overall client demand and enhance the performance and quality (QoS) of data across the Internet, new Internet protocols are being deployed. Degermark et al. (1999).

-+hexadecimal is to handle the increase demand for IP addresses and provides better support for mobile IP and network security.

Additionally, it was said that the inclusion of the network-layer encryption technique known as Internet Protocol Security, (Psec), which doubled the size of packet headers from 20 bytes in IPv4 to at least 40 bytes in IPv6, substantially twice the IP operating overhead. EFFNET AB (2004). Historically, compressing protocol headers—like the IPv6 header—has been a desirable method of preserving bandwidth on slow lines, particularly wireless systems. Ercetin and Tassiuslas (2003) The techniques that shrink the 40-byte IPv6 header overhead to fewer bytes (2 or 4 bytes) would boost user throughput and the number of users a network could accommodate. This paper formulated two mathematical models namely maximum packet throughput (***max_pkt***) and header overhead percentage (***hdr_ovh_pcent***) to justify the performance of header size reduction on available IPv6 network bandwidth.

2. WORK

2.1. OSI/ISO REFERENCE MODEL

Open System Interconnection/International Standard Organization (OSI/ISO) model is a communication standard that was established by ISO in 1984 to define and standardize data communications. OSI/ISO reference model still heavily influences network communications, particularly for the telecommunications industry. As indicated in Table 1, the OSI/OSI reference model is composed of seven (7) layers: Application, Presentation, Session, Transport, Network, Data-Link, and Physical.

The concept of OSI/ISO reference model is that Information Technology (IT) systems vendors can select a protocol for each layer while ensuring compatibility with systems from other vendors that may use different protocols Knutson (2004). Meanwhile, OSI model had provided a common language for communications networking community Lilley et al. (2000).

Table 1

Table 1 OSI Model

OSI Layer	Function provided
7 Application	Network application such as file transfer and terminal emulation.
6 Presentation	Data formatting and encryption.
5 Session	Establishment and maintenance of sessions.
4 Transport	Provision of end-to-end reliable and unreliable delivery.
3 Network	Delivery of packets of information which includes routing
2 Data Link	Transfer of units of information, framing and error checking
1 Physical	Transmission of binary data of a medium

2.2. TCP/IP PROTOCOL SUITE

A group of communication protocols called TCP/IP, or Transmission Control Protocol/Internet Protocol, is used to connect network devices on the internet. TCP/IP streamlined the 7-Layer OSI/ISO model into 4-layer TCP/IP protocol stack as illustrated in Table 2. A private computer network will often implement TCP/IP as its communications protocol of choice (an intranet or extranet) Mary and Shacklett (2021).

Table 2

Table 2 TCP/IP protocol suite	
APPLICATION	HTTP, SMTP, FTP, SNMP
TRANSPORT	TCP, UDP
INTERNET	IP, IPV4, IPV6, ICMP, EIGRP, RIP
LINK	ETHERNET, PPP, ISDN

Layer 4: Application Layer: This consists of protocols that support specific applications or services to end-users. For instance, HTTP is used for locating web servers, FTP is used for file transfer between hosts, SMTP provides e-mail transfer among end-users.

Layer 3: Transport Layer: It is responsible for message delivery between processes with end-to-end communication. TCP provides connection-oriented and reliable packets transmission in a specific order and checks for error in transmission while UDP is a connectionless-oriented and unreliable transport protocol.

Layer 2: Internet Layer: This is also known as network layer which sends the packets from any network and any computer to reach a destination irrespective of route they follow. IP is the most common internet-layer that routes source address to destination address on the network, ICMP is used by a host and gateway to send errors in transmission back to the sender i.e., error notification while the ARP associates the logical (IP) addresses with the physical (MAC) addresses.

Layer 1: Data-link: This layer is responsible for simply transmitting a frame of bits between cooperating processes on different machines.

2.3. IPV6 HEADER COMPRESSION OPERATION

The flexibility of IPv6 header to be compressed from 40 bytes to 2-4 bytes would definitely reduce the header overhead and allows small packets for bulk data with good line efficiency Murdocca and Heuring (2000). The UML diagram for IPv6 header compression is shown in Header compression takes advantage of the hop-by-hop redundancy of the headers in a flow of packets over a link Smythe (1995). Instead of sending the whole header, a compressed header can be sent.

A full header containing a context identifier, or CID, is sent over the network to start the compression of the headers of a packet stream.

Most of the fields in this entire header are stored as context by the compressor and decompressor.

The fields in the headers that make up the context have constant values; therefore, nothing needs to be transmitted over the network. Upon the reception of a packet with an associated context, the compressor removes the IPv6 header, 40Bytes from the packet and appends CID which is 2-4bytes long EFFNET AB (2004). Then the compressed IPv6 header with **CID = 90 (binary 1011010)** is transmitted over the link to the receiving node where there is decompressor which examines the **CID=90** with its own context array, if it is found it inserts IPv6 header back into the packet and transmits the packet to the exact destination. Figure 1 illustrates the initialization and operation of header compression and indicates that the reduction in the number of full headers transmitted can result in overall decreased overhead.

Figure 1

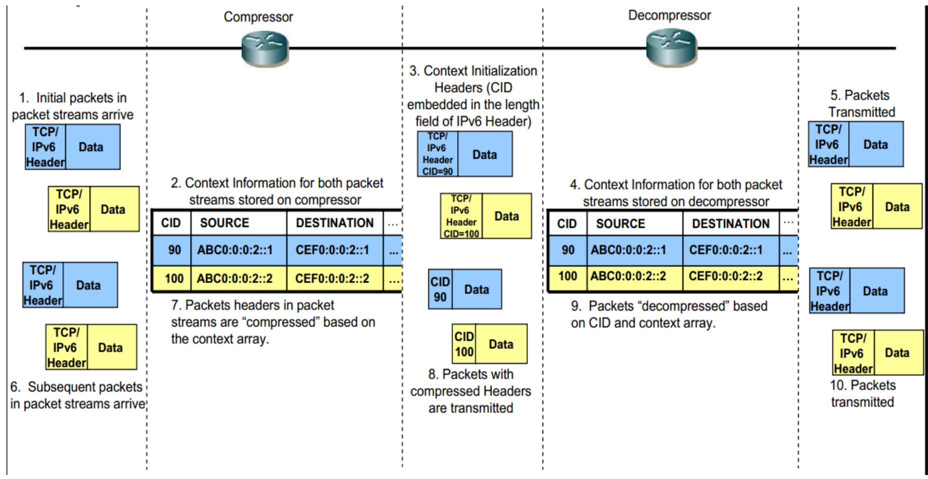

Figure 1 Initialization and Operation of Header Compression Protocols Ertekin et al (2004)

3. METHODOLOGY

Two equations were deduced to evaluate the performance of the header compression in IPv4 and IPv6 scenarios. The first model is ***maximum number of packets*** which is based on the effect of reducing header size on maximum number of packets transmitted over a data network and it is defined by Equation 1

$$max\_pkt = \frac{bw}{pload\_z + hdr\_z}$$

Equation 1

where ***max_pkt*** is the maximum number of packets transmitted per second,

bw is the bandwidth of the network link,

pload_z is the payload size and ***hdr_z*** is the header size.

The second approach defines how the packet header reduction minimizes the header overhead and it is called header overhead percentage ***hdr_ovh_pcent*** which is described by Equation 2

$$hdr\_ovh\_pcent = \frac{hdr\_z}{hdr\_z + pload\_z} \ X \ 100\,\%$$

Equation 2

where ***hdr_ovh_pcent*** is the header overhead percentage

hdr_z is the header size

pload_z is the payload size

Meanwhile, the performance metrics were simulated using MatLab Simulink from which graphical results were generated to evaluate the header compression effect on network bandwidth. A peer-to-peer network shown in Figure 2 was setup using Windows 10 computers and Ethernet cable to validate the performance of IPv6 connection over IPv4.

Figure 2

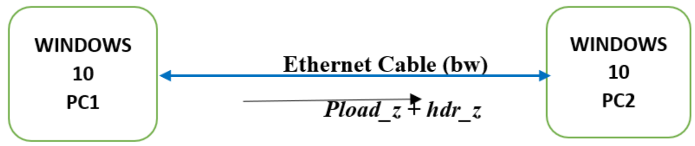

Figure 2 Simple Peer-To-Peer Network

4. SIMULATIONS

MatLab Simulink was employed to analyze the two empirical models in (1) and (2). The following were assumed in order to achieve ideal simulation results.

1) Constant bandwidth of 512 bytes-per-second was used for all the simulations.
2) Transmission medium effect was negligible.
3) Extension headers were not taken into consideration.
4) Error bits generated in transmission were negligible.
5) Compression and decompression timings were negligible.
6) Packet reordering was negligible.

Two simulations were run for a constant bandwidth value of 512bps and payload size ranging from 50 to 150 bytes for three different header sizes: **20-Byte** IPv4 header, **40-Byte** IPv6 header and **2-Byte** Compressed IPv6 header. The first simulation in Figure 3 was a measure of maximum number of packets, ***max_pkt*** generated from different payload sizes with a specified header size while the Figure 4 was the simulation for header overhead percentage ***hdr_ovh_pcent*** on different payload sizes.

Figure 3

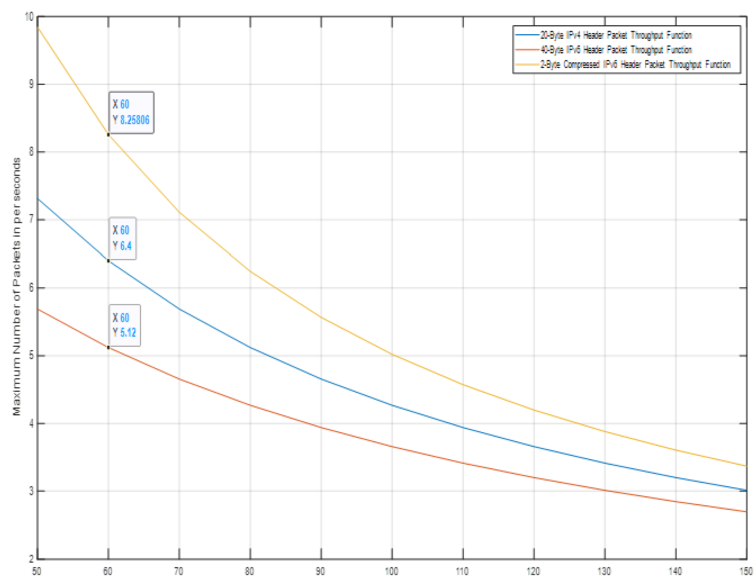

Figure 3 Maximum Packet Throughput

Figure 4

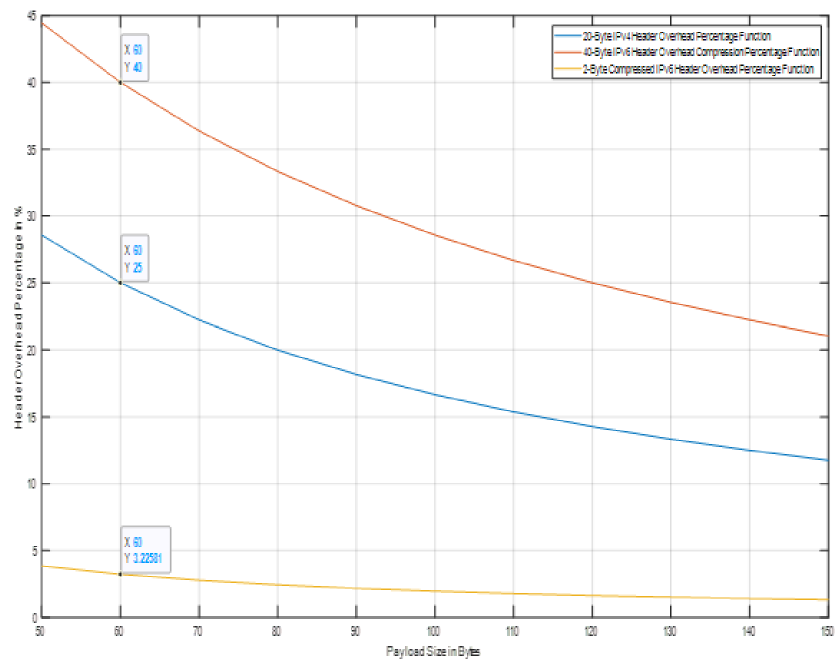

Figure 4 Header Overhead Percentage

5. RESULTS AND DISCUSSIONS

The simulation's graphical results shown in Figure 3 and Figure 4 were generated based on the corresponding mathematical equations (1) and (2) for maximum packet throughput and header overhead percentage respectively.

Figure 3 is a non-linear function with a plot of payload size ***pload_z*** against maximum number of packets ***max_pkt***. It shows three simulation outputs for 20-byte IPv4, 40-byte IPv6 and 2-byte compressed IPv6 headers which are indicated by colors blue, red and yellow respectively. Given a specific payload size at **X=60**, 2-byte compressed IPv6 header gave the highest number of packets per second at **Y= 8 (approx.),** 20-byte IPv4 header payload delivered number of packets per seconds at **Y = 6 (approx.)** while the uncompressed 40-byte IPv6 header output the least number of packets per seconds at **Y= 5 (approx.).**

This simulation result had demonstrated that more packets are transmitted with reducing the header size attached to a payload.

The second non-linear function is a plot of payload size ***pload_z*** against header overhead percentage ***hdr_ovh_pcent.*** The curves demonstrate that the smaller the header size, the lower the overhead percentage which makes it easy for packets transmission. Given a specific payload size of **X=60byes,** the 2-byte compressed IPv6 header in color yellow has the lowest header overhead percentage at **Y=3 (approx.)** while the red-colored 40-byte IPv6 header has the highest header overhead at **Y=40** which would have made network bandwidth constrained.

5.1. RESULTS VALIDATIONS

A simple peer-to-peer connection was setup between two Windows 10 computers using Ethernet cable to establish that IPv6 network is bandwidth-optimized compared to IPv4 network. The two computers were configured with IPv4 addresses **192.168.0.2/24** and **192.168.0.3/24** while the IPv6 addresses used are **fe::3/64** and **fe::2/64** respectively .

The packets ping statistics on the command prompt is shown in Figure 5

Figure 5

```
Command Prompt
Microsoft Windows [Version 10.0.19043.1706]
(c) Microsoft Corporation. All rights reserved.

C:\Users\UPLAND>ping 192.168.0.3 -t

Pinging 192.168.0.3 with 32 bytes of data:
Reply from 192.168.0.3: bytes=32 time=1213ms TTL=128
Reply from 192.168.0.3: bytes=32 time=699ms TTL=128
Reply from 192.168.0.3: bytes=32 time=391ms TTL=128
Reply from 192.168.0.3: bytes=32 time=78ms TTL=128
Reply from 192.168.0.3: bytes=32 time=246ms TTL=128
Reply from 192.168.0.3: bytes=32 time=170ms TTL=128
Reply from 192.168.0.3: bytes=32 time=562ms TTL=128
Reply from 192.168.0.3: bytes=32 time=266ms TTL=128
Reply from 192.168.0.3: bytes=32 time=656ms TTL=128
Reply from 192.168.0.3: bytes=32 time=344ms TTL=128
Reply from 192.168.0.3: bytes=32 time=32ms TTL=128
Reply from 192.168.0.3: bytes=32 time=423ms TTL=128

Ping statistics for 192.168.0.3:
    Packets: Sent = 12, Received = 12, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 32ms, Maximum = 1213ms, Average = 423ms
Control-C
^C
C:\Users\UPLAND>ping fe::2 -t

Pinging fe::2 with 32 bytes of data:
Reply from fe::2: time=1210ms
Reply from fe::2: time=233ms
Reply from fe::2: time=391ms
Reply from fe::2: time=78ms
Reply from fe::2: time=485ms
Reply from fe::2: time=2ms
Reply from fe::2: time=579ms
Reply from fe::2: time=266ms
Reply from fe::2: time=657ms
Reply from fe::2: time=344ms
Reply from fe::2: time=31ms
Reply from fe::2: time=438ms
Reply from fe::2: time=141ms

Ping statistics for fe::2:
    Packets: Sent = 13, Received = 13, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 1210ms, Average = 373ms
Control-C
^C
C:\Users\UPLAND>
```

Figure 5 IPv4 and IPv6 Ping Statistics

It could be deduced that the average round trip times (RTT) of **373m** in packets transmission in the IPv6 connection is smaller compared to the IPv4 metrics which is **423ms.**

Also, a careful check of the series of the IPv6 packets in the ping test shows a smaller transmission time compared to that of IPv4.

6. CONCLUSION

The numerical evaluations of the effect of reducing or compressing packet header had significantly shown that the embedded compression features in IPv6 protocol stack provide higher speed than IPv4 protocol. The ping test results also established the fact that the IPv6 network has a higher performance in bandwidth optimization with lower latency.

The deployment of IPv6 had started in full of newer systems including operating systems and network devices come with the provision of configuring IPv6 addresses to address the shortage in IPv4 address space and improve the routing core.

CONFLICT OF INTERESTS

None.

ACKNOWLEDGMENTS

None.

REFERENCES

Bormann, C. (2002). Robust Header Compression (ROHC) over PPP, RFC 3241. https://doi.org/10.17487/RFC3241

Degermark et al, (1999). Standards Track IP Header Compression, RFC2507.

EFFNET AB (2004). An Introduction to IPv6 Header Compression.

Ercetin, O. and Tassiuslas, L. (2003). Market-Based Resource Allocation for Content Delivery in the Internet in IEEE Transactions on Computers.52(12), 1573-1584. https://doi.org/10.1109/TC.2003.1252853

Ertekin et al (2004). IPv6 Header Compression paper at North American IPv6 Summit.

Knutson, C. (2004). Evaluation and Implementation of Header Compression Algorithm ECRTP. Unpublished MSc Thesis, Lulea University of Technology, Lulea.

Lilley et al (2000). A Unified Header Compression Framework for Low-Bandwidth Links, 6th International Conference on Mobile Computing and Networking.

Mary E., and Shacklett et al (2021). TCP/IP.

Murdocca, M.J., and Heuring, V.P. (2000). Principles of Computer Architecture. Tom Robbins, United State of America.

Smythe, C. (1995). Internetworking, Designing the Right Architecture. Addison-Wesley Publishers Ltd., Cambridge.