IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 13, Issue 5 (Jul. - Aug. 2013), PP 27-34
www.iosrjournals.org

Comparative study of IPv4 & IPv6 Point to Point Architecture on various OS platforms

[1]Dinesh Goyal, [2]Rajendra Singh, [3]Dr. Naveen Hemrajani
[1.2.3]*Suresh Gyan Vihar University, Jaipur*

Abstract: *In this thesis, a comparative study on the performance analysis of IPv4 and IPv6 protocol stacks under Microsoft Windows 2007, MAC and Red Hat Linux Enterprise version 4 in point-to-point and router-to-router architectures have been done in terms of bandwidth utilization (throughput) for different data sizes, round trip time (latency) computation and overhead variation calculation. Real-time experiments have been carried out for the above-mentioned architectures in the laboratory. For point-to-point architecture, three PCs were configured at IPv4 and IPv6 under the Windows, MAC and Linux operating platforms respectively.*

I. Introduction

In 1970, Internet Protocol was designed and introduced to industry in 1981 to objective of the interconnecting of heterogeneous network technologies. IP plays a key role to get popularity of Internet. The huge success of the Internet is pushing IPv4 to its limits [1]. Internet Engineering Task Force (IETF) [2] took initiative to address the limitations of IPv4 in 1990s. IPv4 uses a 32-bit field to identify host interfaces known as Internet Protocol Addresses. When IPv4 was designed 32 bits were enough and the IETF never thought of any limitations of IPv4 for support such a big network like Internet. This 32-bit field is becoming restrictive nowadays; an Internet Address is in short supply. The IETF began to design a successor to IPv4: IPv6 ((Internet Protocol version 6). IPv6 [3] is the new version of the Internet Protocol and it has several improvements. It has extended addressing capabilities; the address field is 128-bits in length. With IPv6, we a have a far greater address space (3.4×2^{38} addresses), we can connect more devices to the Internet without breaking the end-to- end principle, create a complex address hierarchy and benefit from simpler configuration. IPv6 also provides an improved header format and routers are able to process the IPv6 header in a more efficient way. Options (e.g. mobility and security) are a patch in the IPv4 header but, in IPv6, such features are part of the protocol (using the new extension header format).

In summary, the Internet will be even more scalable with IPv6 than with IPv4. The Internet is still using IPv4, but IPv6 is now being widely deployed in research networks, & this deployment is a critical issue. In the future it is possible that the Internet will be IPv6 only. IPv6 deployment must not disrupt the current Internet and, somehow, IPv4 and IPv6 must coexist. It is done by special mechanisms, named transition mechanisms, which allow communication between the IPv4 and the IPv6 world. Transition mechanisms have been designed & implemented but they provide less forwarding speed than a native communication and some of them are difficult to deploy.

The proposed study intend to examine the performance of both the IPv4 and IPv6 protocols in three different platforms, namely Microsoft Windows 2007, MAC and Red Hat Linux Enterprise Version 4 on identical hardware and IPv6 transition mechanism. Our experiments were conducted over an unloaded network using three routers and three workstations.

II. IPV4

The fields in the IPv4 header are:

1. **Version** – Indicates the version of IP and is set to 4. The size of this field is 4 bits.
2. **Internet Header Length** – Indicates the number of 4-byte blocks in the IPv4 header. The size of this field is 4 bits. Because an IPv4 header is a minimum of 20 bytes in size, the smallest value of the Internet Header Length (IHL) field is 5. IPv4 options can extend the minimum IPv4 header size in increments of 4 bytes. If an IPv4 option does not use all 4 bytes of the IPv4 option field, the remaining bytes are padded with 0's, making the entire IPv4 header an integral number of 32-bits (4 bytes).
   With a maximum value of 0xF, the maximum size of the IPv4 header including options is 60 bytes (15×4).
3. **Type of Service** – Indicates the desired service expected by this packet for delivery through routers across the IPv4 internetwork. The size of this field is 8 bits, which contain bits for precedence, delay, throughput, and reliability characteristics.

4. **Total Length** – Indicates the total length of the IPv4 packet (IPv4 header + IPv4 payload) and does not include link layer framing. The size of this field is 16 bits, which can indicate an IPv4 packet that is up to 65,535 bytes long.
5. **Identification** – Identifies this specific IPv4 packet. The size of this field is 16 bits. The Identification field is selected by the originating source of the IPv4 packet. If the IPv4 packet is fragmented, all of the fragments retain the Identification field value so that the destination node can group the fragments for reassembly.
6. **Flags** – Identifies flags for the fragmentation process. The size of this field is 3 bits; however, only 2 bits are defined for current use. There are two flags—one to indicate whether the IPv4 packet might be fragmented and another to indicate whether more fragments follow the current fragment.
7. **Fragment Offset** – Indicates the position of the fragment relative to the original IPv4 payload. The size of this field is 13 bits.
8. **Time to Live** – Indicate the maximum number of links on which an IPv4 packet can travel before being discarded. The size of this field is 8 bits. The Time-to-Live field (TTL) was originally used as a time count with which an IPv4 router determined the length of time required (in seconds) to forward the IPv4 packet, decrementing the TTL accordingly. Modern routers almost always forward an IPv4 packet in less than asecond and are required by RFC 791 to decrement the TTL by at least one. Therefore, the TTL becomes a maximum link count with the value set by the sending node. When the TTL equals 0, an ICMP Time Expired-TTL Expired in Transit message is sent to the source IPv4 address and the packet is discarded.
9. **Protocol** – Identifies the upper layer protocol. The size of this field is 8 bits. For example, TCP uses a Protocol of 6, UDP uses a Protocol of 17, and ICMP uses a Protocol of 1. The Protocol field is used to demultiplex an IPv4 packet to the upper layer protocol.
10. **Header Checksum** – Provides a checksum on the IPv4 header only. The size of this field is 16 bits. The IPv4 payload is not included in the checksum calculation as the IPv4 payload and usually contains its own checksum. Each IPv4 node that receives IPv4 packets verifies the IPv4 header checksum and silently discards the IPv4 packet if checksum verification fails. When a router forwards an IPv4 packet, it must decrement the TTL. Therefore, the Header Checksum is recomputed at each hop between source and destination.
11. **Source Address** – Stores the IPv4 address of the originating host. The size of this field is 32 bits.
12. **Destination Address** – Stores the IPv4 address of the destination host. The size of this field is 32 bits.
13. **Options** – Stores one or more IPv4 options. The size of this field is a multiple of 32 bits. If the IPv4 option or options do not use all 32 bits, padding options must be added so that the IPv4 header is an integral number of 4-byte blocks that can be indicated by the Internet Header Length field.

III. IPv6

The fields in the IPv6 header are:

1. Version – 4 bits are used to indicate the version of IP and is set to 6.
2. Traffic Class – Indicates the class or priority of the IPv6 packet. The size of this field is 8 bits. The Traffic Class field provides similar functionality to the IPv4 Type of Service field. The use of the Traffic Class field is defined in RFC 3697.
3. Flow Label – Indicates that this packet belongs to a specific sequence of packets between a source and destination, requiring special handling by intermediate IPv6 routers. The size of this field is 20 bits. The Flow Label is used for non-default quality of service connections, such as those needed by real-time data (voice and video). For default router handling, the Flow Label is set to 0. There can be multiple flows between a source and destination, as distinguished by separate non-zero Flow Labels.
4. Payload Length – Indicates the length of the IPv6 payload. The size of this field is 16 bits. The Payload Length field includes the extension headers and the upper layer PDU. With 16 bits, an IPv6 payload of up to 65,535 bytes can be indicated. For payload lengths greater than 65,535 bytes, the Payload Length field is set to 0 and the Jumbo Payload option is used in the Hop-by-Hop Options extension header.
5. Next Header – Indicates either the first extension header (if present) or the protocol in the upper layer PDU (such as TCP, UDP, or ICMPv6). The size of this field is 8 bits. When indicating an upper layer protocol above the Internet layer, the same values used in the IPv4 Protocol field are used here.
6. Hop Limit – Indicates the maximum number of links over which the IPv6 packet can travel before being discarded. The size of this field is 8 bits. The Hop Limit is similar to the IPv4 TTL field except that there is no historical relation to the amount of time (in seconds) that the packet is queued at the router. When the Hop Limit equals 0, an ICMPv6 Time Exceeded message is sent to the source address and the packet is discarded.

7. Source Address – Stores the IPv6 address of the originating host. The size of this field is 128 bits.
8. Destination Address – Stores the IPv6 address of the current destination host. The size of this field is 128 bits. In most cases the Destination Address is set to the final destination address. However, if a Routing extension header is present, the Destination Address might be set to the next router interface in the source route list.

IV. IPv4 Vs. IPv6

Following table 2.2 shows the key differences between IPv4 and IPv6 protocol. "Introduction to IP Version 6" published by Microsoft Corporation dated February 2006 [12] where a detail description presented on IPv6 and its features and address format etc. All this key issues are defined in the various Requests for Comments - RFC lead by Internet Engineering Task Force – IETF. The left side of the table represents IPv4's features and the right side represents IPv6's features.

Table 1 Differences between IPv4 and IPv6 [12]

IPv4	IPv6
Source and destination addresses are 32 bits (4	Source and destination addresses are 128 bits (16
IPsec support is optional.	IPsec support is required
No identification of packet flow for QoS handling by routers is present within the IPv4 header.	Packet flow identification for QoS handling by routers is included in the IPv6 header using the Flow Label field.
Fragmentation is done by both routers and the sending host.	Fragmentation is not done by routers, only by the sending host.
Header includes a checksum.	Header does not include a checksum.
Header includes options.	All optional data is moved to IPv6 extension headers.
Address Resolution Protocol (ARP) uses broadcast ARP Request frames to resolve an IPv4 address to a link layer address.	ARP Request frames are replaced with multicast Neighbor Solicitation messages.
Internet Group Management Protocol (IGMP) is used to manage local subnet group membership.	IGMP is replaced with Multicast Listener Discovery (MLD) messages.
ICMP Router Discovery is used to determine the IPv4 address of the best default gateway and is optional.	ICMP Router Discovery is replaced with ICMPv6 Router Solicitation and Router Advertisement messages and is required.
Broadcast addresses are used to send traffic to all nodes on a subnet.	There are no IPv6 broadcast addresses. Instead, a link-local scope all-nodes multicast address is used.
Must be configured either manually or through DHCP.	Does not require manual configuration or DHCP.
Uses host address (A) resource records in the Domain Name System (DNS) to map host names to IPv4 addresses.	Uses host address (AAAA) resource records in the Domain Name System (DNS) to map host names to IPv6 addresses.
Uses pointer (PTR) resource records in the IN-ADDR.ARPA DNS domain to map IPv4 addresses to host names.	Uses pointer (PTR) resource records in the IP6.ARPA DNS domain to map IPv6 addresses to host names.
Must support a 576-byte packet size.	Must support a 1280-byte packet size

From the above table we understood the difference of both IPv4 and IPv6 protocol and now we look in to the IPv4 addresses and IPv6 equivalents as under:

Table 2 IPv4 Addresses and IPv6 Equivalents [12]

IPv4 Address	IPv6 Address
Internet address classes	Not applicable in IPv6
Multicast addresses (224.0.0.0/4)	IPv6 multicast addresses (FF00::/8)
Broadcast addresses	Not applicable in IPv6
Unspecified address is 0.0.0.0	Unspecified address is ::
Loopback address is 127.0.0.1	Loopback address is ::1
Public IP addresses	Global unicast addresses
Private IP addresses (10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16)	Site-local addresses (FEC0::/10)
Autoconfigured addresses (169.254.0.0/16)	Link-local addresses (FE80::/64)
Text representation: Dotted decimal notation	Text representation: Colon hexadecimal format with suppression of leading zeros and zero compression. IPv4-compatible addresses are expressed in dotted decimal notation.

Network bits representation: Subnet mask in dotted decimal notation or prefix length	Network bits representation: Prefix length notation only
DNS name resolution: IPv4 host address (A) resource record	DNS name resolution: IPv6 host address (AAAA) resource record
DNS reverse resolution: IN-ADDR.ARPA domain	DNS reverse resolution: IP6.ARPA domain

From the above tables we understood the difference of both IPv4 and IPv6 protocol and IP addresses and now we look in to the differences of header fields of both protocols as under:

Table 3 Comparing the IPv4 and IPv6 Headers [13]

IPv4 Header Field	IPv6 Header Field
Version	Same field but with different version numbers.
Internet Header Length	Removed in IPv6. IPv6 does not include a Header Length field
Type of Service	Replaced by the IPv6 Traffic Class field.
Total Length	Replaced by the IPv6 Payload Length field, which only indicates the size of the payload.
Identification Fragmentation Flags Fragment Offset	Removed in IPv6. Fragmentation information is not included in the IPv6 header. Header.
Time to Live	Replaced by the IPv6 Hop Limit field.

The one new field in the IPv6 header that is not included in the IPv4 header is the Flow Label field.

Ipv4 To Ipv6 Transition Mechanisms And Scenario

The designers of IPv6 recognize that the transition from IPv4 to IPv6 will take years and that there might be organizations or hosts within organizations that will continue to use IPv4 indefinitely. Therefore, while migration is the long-term goal, equal consideration must be given to the interim coexistence of IPv4 and IPv6 nodes. There are different types of node exist in the network such as [14] IPv4-only, IPv6-only, IPv6/IPv4 node, IPv4 node and IPv6 node. There are different types of compatibility addresses such as IPv4-compatible addresses, IPv4-mapped addresses, 6over4 addresses, 6to4 addresses, ISATAP addresses, Teredo addresses. To coexist with an IPv4 infrastructure and to provide an eventual transition to an IPv6-only infrastructure, the following mechanisms are used.

V. Laboratory Setup For The Experiment

In our test-lab, we arranged a set of hardware and software. Our test setup consists three dual stack (IPv4/IPv6) routers: three Cisco routers model 2811. Dual stack implementation specification can be found in [16]. We have three identical workstations that were connected directly to the routers and were configured to be separate networks. Each router supported two separate networks each. All workstations were equipped with Intel Pentium IV 910 MHz processors, 512 megabytes of SDRAM, and 3COM 10/100 PCI network adapters. The workstations were loaded with Windows 2007, Macintosh and Red Hat Linux Enterprise version 4. Windows had the IPv4 stack as a standard protocol; however in order to get IPv6 support, an add-on package was installed, but in Macintosh & Linux was IPv6 loaded automatically. A number of testing tools have been used for the experiment such as IPerf 1.7.0 [17] and PING.

Performnace Metrics

We use bandwidth utilization (throughput) and round trip time (latency) performance metrics for measuring performance of IPv4 and IPv6 protocols. IPerf 1.7.0 and PING tools are used to measure performance. The measurement interval was selected to be 60 seconds and the data sizes were about 128 KB to 61.44 MB. Each test was repeated several times to obtain consistent results. Metrics parameters are in the subsequent sections.

Bandwidth Utilization

Bandwidth Utilization (throughput) [18] is the net carrying capacity of an element corrected for overhead.

Throughput is a theoretical value, calculated based on the operating characteristics of a particular network. It represents the effective capacity of a connection or service once all the things are considered. Following formula illustrates this concept and how it relates to estimate the throughput of a device or network link.

L

$$qL = [(Q/K) - \sum_{i=1} \theta_i]d$$

where qL is the realized channel throughput at protocol layer, L, Q is the gross data rate based on the transmission technology, K is the number of channels or traffic flows, θi is the channel protocol overhead at layer i, and d is the duplex factor. θi is the accumulated protocol loss over layer L and subtending layers.

Round Trip Time (Latency)

Round trip tine [18] is sometimes used interchangeably with *response time,* which is the time taken between sending and reception of the data. Response time can be thought of as round trip time from the perspective of the user, or the sending device. For this reason, the same caveats that applied to round trip time also apply to response.

The PING program is often used to measure network response time. This is an Internet control message protocol (ICMP), message that sends packets to a specific host at an IP address and times the response. Although this program can be indicative of network-based processing such as connection setup, routing, and transmission delay, it may not be truly reflective of overall response from a service perspective.

VI. Results Of Bandwidth Utilization

Figure 1 shows that both IPv4 and IPv6 protocols under Windows perform quite closely. IPv6 incurs 1 to 2% more overhead in this type of data size.

Fig.1 Performance of TCP in IPv4 and IPv6 under Windows

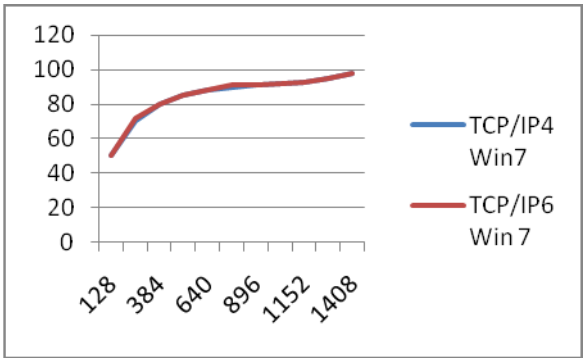


Fig.2 Performance of TCP in IPv4 and IPv6 under Linux

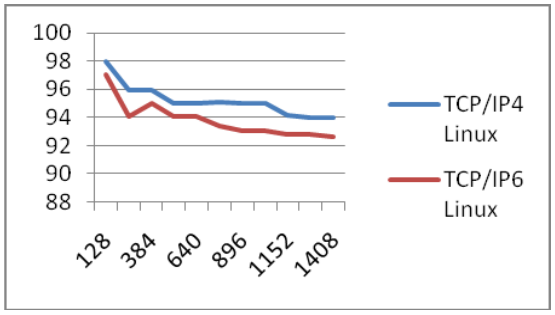


Fig.3 Performance of TCP in IPV4 and IPv6 under MAC

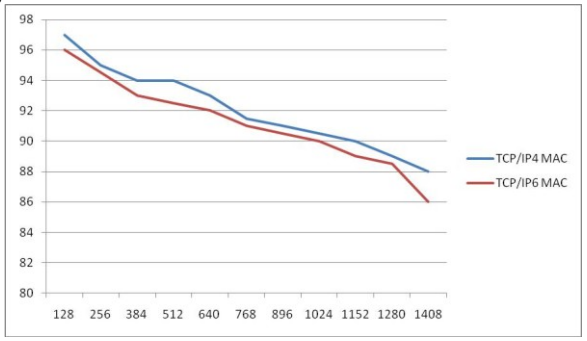


Fig. 4 Performance of TCP in IPv4 under Windows, Mac and Linux

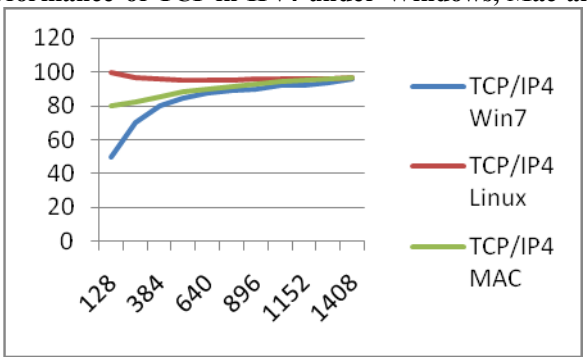


Fig. 5 Performance of TCP in IPv6 under Windows, Mac and Linux

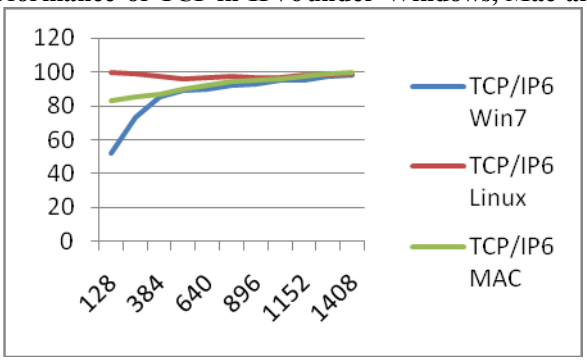
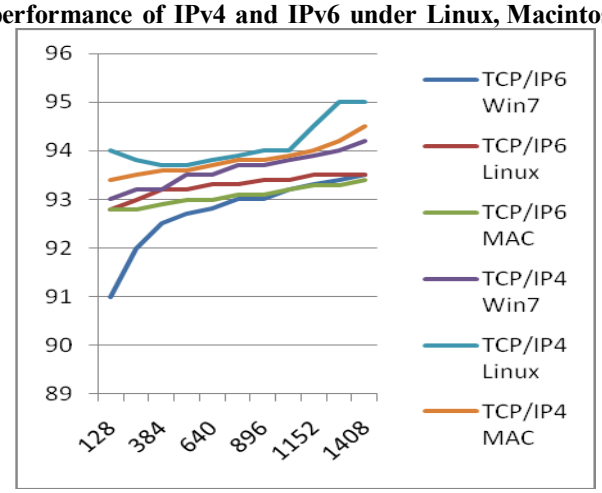


Fig. 6. Overall performance of IPv4 and IPv6 under Linux, Macintosh and Windows



VII. Results Round Trip Time (Rtt)

Fig. 7 Performance of Latency in IPv4 and IPv6 under Windows

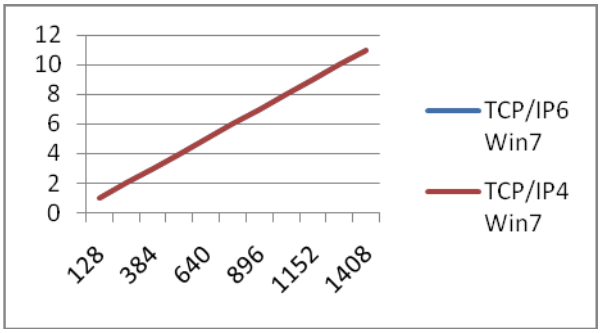


Fig. 8 Performance of Latency in IPv4 and IPv6 under Linux

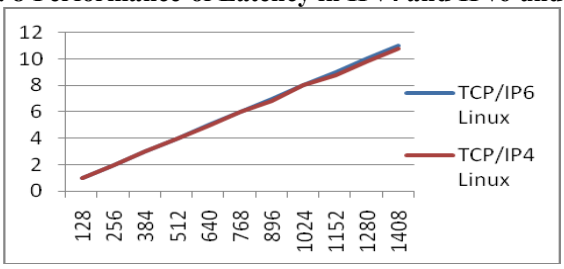


Fig. 9 Round Trip Time Results for IPv4/IPv6 under MAC

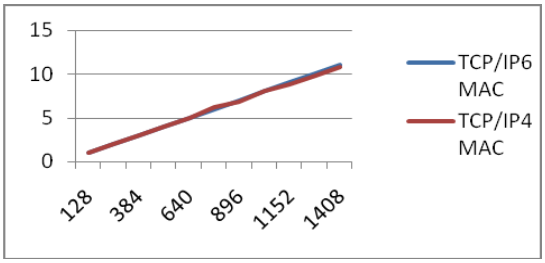
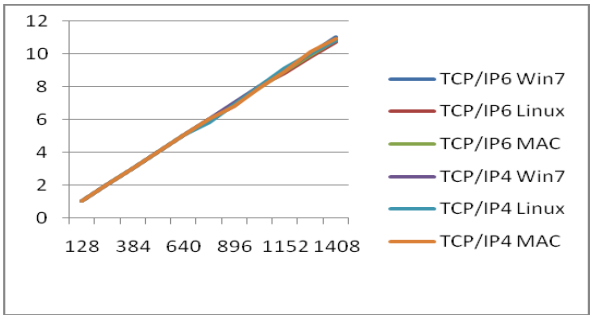


Fig. 10 Performance of Latency in IPv4 and IPv6 under Windows, Mac and Linux



VIII. Analysis

1. Bandwidth Utilization for Point-to-Point Architecture

Under Windows, bandwidth utilization results for IPv4 and IPv6 with data size ranging from 128 KB to 1.408 MB as shown earlier shows that the performance indicators are quite close. In comparison to IPv4, the IPv6 incurs 1 to 2% more overhead in this type of data sizes.

As the header size of IPv6 is bigger than that of IPv4, probably IPv6 incurs more overhead than IPv4. More overhead results for bigger message of bigger data size happens due to bigger number of data packets and its corresponding acknowledgement time used up by the protocol in comparison to smaller message of smaller data sizes.

Under Macintosh, bandwidth utilization results for IPv4 and IPv6 with data size ranging from 128 KB to 1.408 MB as shown earlier shows that the performance indicators are quite close but better than windows. In comparison to IPv4, the IPv6 incurs 2 to 3% more overhead in this type of data sizes.

Under Linux, bandwidth utilization results of IPv6 incurs around 2% more overhead in the smaller data sizes ranging from 128 KB to 1.408 MB as shown earlier.

As IPv6 has bigger header than IPv4 header, in Linux also, IPv6 incurs more overhead than IPv4.

We see that IPv6 under Linux performs better than under, Macintosh, which in turn performs better than Windows for all kinds of data sizes, but at smaller data size level, performance of Windows is poorer. As the data size grows bigger and bigger, the difference becomes lesser and lesser. The reason may be perhaps due to the use of different algorithms and time acknowledgement differences in Windows, Macintosh and Linux platforms.

2. Round Trip Time Computation for Point-to-Point Architecture

As seen earlier, both IPv4 and IPv6 protocols perform at the same level of efficiency under Windows. Actually, Windows permits millisecond level time resolution only. So, it is difficult to capture time in microsecond level directly for smaller sizes data.

We see that IPv4 and IPv6 perform quite closely under Windows. IPv6 incurs 1.8 to 2.9% more overhead for all ranges of data sizes, which matches with theoretical speculations also. IPv6 header is 20 bytes bigger than that of IPv4 and the difference happens to be bigger for bigger overhead.

IX. Conclusion

In the present work, we carried out a series of experiments to compare the performance analysis of IPv4 and IPv6 stack protocols under Windows 2007, Macintosh and Red Hat Linux Enterprise Version 4 platforms. We measured the performance parameters for the protocols in terms of bandwidth utilization and RTT (latency) computation for host-to-host architectures.

Performance analysis for point-to-point architecture was carried out to see only the normal operational characteristics of both the protocols. But our experiments are mostly focused on the router-to-router bandwidth utilization and RTT (latency) performance measurements only.

Another observation is that under Linux platform, bandwidth utilization is better than, Macintosh, which is better that under Windows.

Interestingly, we find from our experimental results that the bandwidth utilization and RTT (latency) parameters of IPv4 are superior to those of IPv6 protocols. For thiscase, we infer that IPv6 results are poorer in comparison to IPv4 due to the bigger overhead constraints of IPv6.

It is an overall observation that router-to-router RTT (latency) performance figures are always less than those of the host-to-host values.

X. Future Work

More research on the following aspects will be useful for further study in this area:

1. Study can be extended to comparative evaluation with IPv6 implementation on other platforms, such as Sun Solaris 10 operating platform;
2. Study can be extended to different router platforms, such as Nortel, Juniper etc.
3. Study can also be extended to using IPSec in IPv6 implementation to observe the overhead enhancement due to encryption and decryption processes;
4. Quality of Service (QoS) testing in IPv6 implementation;
5. Study can be extended t o application test in IPv6-enabled applications services, such as email, web, ftp, video conferencing etc.

References

[1]. Postal, J. “Internet Protocol”. RFC 791, September 1981
[2]. The Internet Engineering Task Force (IETF) http:// www.ietf.org.
[3]. Deering, S., and Hinder,”Internet Protocol, Version 6 (IPv6) Specification.”RFC 2460, December 1998
[4]. “The Design and Implementation of an IPv6/IPv4 Network Address and Protocol Translator” ” Department of Computer Science and Engineering, University of Washington, Seattle, Washington 98195 http://www.cs.princeton.edu/~mef/research/napt/reports/usenix98/
[5]. Ioan Raicu “IPv6 Performance Results”, cs.wayne.edu
[6]. Yi Wang 1, Shaozhi Ye 2, Xing Li, “Understanding Current IPv6 Performance: A Measurement Study”, 3 Department of Electronic Engineering, Tsinghua University, Beijing 100084, P. R. China http://doi.ieeecomputersociety.org/10.1109/ISCC.2005.151
[7]. Sharif Ghazzawi and Chongenun Lee, “Application Response Times for Internet Protocol Version 4 (IPv4) versus Internet Protocol Version 6 (IPv6), The MITRE Corporation, 7525 Colshire Dr. McLean, VA 22102 http://www.mitre.org/work/tech_papers/tech_papers_05/05_0231/05_0231.pdf
[8]. IPv6 and the Next Generation Internet Protocol Overview”, Sprint Communications, Inc., 26750 Agoura Road ,Calabasas, CA, 91302 USA http://www.spirentcom.com/documents/3191.pdf
[9]. Peter Ping Xie, “Network Protocol Performance Evaluation of IPv6 for Windows NT”, California Polytechnic State University, June 1999
[10]. Ioan Raicu “An Empirical Analysis Of Internet Protocol Version 6 (IPv6)” , Wayne State University, Detroit, Michigan, year 2002
[11]. Behrouz A. Forouzan, “Business Data Communications”, DeAnza College published year 2003
[12]. Introduction to IP Version 6”, published by Microsoft Corporation, published September 2003 and updated February 2006
[13]. Marcus, A. Goncalves, Kitty Niles “IPv6 Networks”, McGraw-Hill, 1998
[14]. R. Gillign, E. Nordmark, “Transition Mechanisms for IPv6 Hosts and Routers”, RFC-2893, August 2000
[15]. “IPv6 Transition Technologies”, Microsoft Corporation, Published October 2003 and updated October 2005
[16]. R. Gillign, E. Nordmark, “Transition Mechanisms for IPv6 Hosts and Routers”, RFC-1933, Internet Engineering Task Force, April 1996
[17]. IPerf WWW pages http://dast.nlanr.net/Projects/Iperf
[18]. Matthew Liotine, “Mission-Critical Network Planning” Publisher: Artech House Publishers (October 2003)
[19]. S. Zeadally, R. Wasseem, I. Raicu “Comparison of end-system IPv6 protocol stacks”IEEE Proceedings - Communication, Vol. 151, No. 3, June 2004