UDP-IPv6 Performance in Peer-to-Peer Gigabit Ethernet using Modern Windows and Linux Systems

Burjiz K. Soorty
School of Computer and Mathematical Sciences
Auckland University of Technology
Auckland, New Zealand

Nurul I. Sarkar
School of Computer and Mathematical Sciences
Auckland University of Technology
Private Bag 92006, Auckland 1142, New Zealand
Email: nurul.sarkar {at} aut.ac.nz

Abstract— **In this paper the performance of IPv6 is investigated using Windows and Linux operating systems over a Gigabit Ethernet link. IPv6 is set to take over IPv4 and is built into most new Windows and Linux client/server operating systems (OSs). Deploying IPv6 in larger networks requires an evaluation to determine which new OS would be most efficient in terms of bandwidth and latency for both TCP and UDP (User Datagram Protocol) applications. While most previous studies have focused on TCP-IPv6 evaluation, a very few researchers have actually evaluated UDP-IPv6 using modern OSs. In this paper we quantify the performance of UDP for IPv6 against IPv4 using four commonly used Windows and Linux systems. Quality of service (QoS) parameters such as throughput, latency, jitter and CPU usage are measured empirically to determine which OS provides the best system performance over IPv6 networks. The effect of packet length on system performance is also investigated. Our findings reported in this paper provide some insight into UDP-IPv6 performance for those operating systems and the effects of packet length on performance.**

Keywords- Bandwidth; IPv6, operating systems; packet length; user-datagram protocol (UDP)

I. INTRODUCTION

IPv6 is the next generation Internet protocol that will replace the current IPv4 in solving the problems of address exhaustion, security, and mobility. IPv6 will not only increase the address space but will also provide new features, such as multimedia delivery, QoS, auto-configuration, security and mobility enhancements. However, IPv6's larger address space (128-bit) will introduce a larger overhead in the header of IP datagrams (a 40-byte header vs. IPv4's 20-byte header). This increased overhead will have drawbacks of lower bandwidth and higher latency when implementing IPv6 in modern OSs. It is useful to be able to quantify the bandwidth as well as latency degradation as a result of IPV6 implementation in peer-to-peer Gigabit Ethernet. In this paper we measure the performance of UDP for IPv6 empirically by considering modern MS Windows and Linux client/server OSs. In particular we compare the performance of IPv6 against IPv4 using four of the newest and commonly used client/server OSs (e.g. Windows 7, Windows Server 2008, Ubuntu 10.04, and Red Hat Enterprise Server 5.5) in a peer-to-peer Gigabit Ethernet link. These OSs were selected based on their popularity and relevance to our study.

Previous studies on IPv6 have shown that transport protocol and packet length can affect the system performance [1]. Most of the previous studies have evaluated the performance of IPv6 for TCP [2]. However, a very limited research has been carried to quantify UDP-IPv6 performance, a transport-layer protocol used for delay-sensitive traffic.

Rapid advancement in internet technology and growing use of IP-based applications, such as Voice over Internet Protocol (VoIP), Internet Protocol Television (IPTV), Mobile Internet, Video-Conferencing, and Online-gaming all have invariably led to the exhaustion of the IPv4 address pool [3].

As IPv4 addresses near exhaustion and the rapid growth of IPv6 deployment, organizations are left with three choices. Firstly, organizations can continue to use IPv4 in internal networks and use Network Address Translation (NAT) to communicate across the current IPv4 infrastructure and use NAT Protocol Translation (NAT-PT) [4]. This option although foregoes any upgrade costs for an organization to upgrade to IPv6 it does also hinder progress towards establishing a predominant IPv6 internet. Second, to perform a complete internal-network migration to IPv6 and use tunneling mechanisms such as 6 to 4 Generic Routing Encapsulation (GRE) [5] and Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) which allow IPv6 communication across current IPv4 infrastructure [6]. Third, continue to use IPv4 and gradually upgrade to IPv6 whilst using both protocols depending on whichever is more beneficial to the organization's needs. Thus instead of undergoing the cost for revamping an entire IPv4 network architecture firmly in place, the company has an option to invest within a comparatively lower budget and purchase a dual-stack router to service IPv4 and IPv6 clients in their internal network [7]. This deployment strategy thus provides both the performance and cost-effectiveness.

Although IPv6 can solve numerous problems associated with IPv4, deploying IPv6 in organizations may have bandwidth issues [8]. This paper provides an in-depth performance evaluation of IPv6 against IPv4 focusing on the effect of client/server OSs and packet length on UDP. These parameters are in-sync with industry standards, for instance with packet-length a range between 128 to 1408 bytes is considered since packet fragmentation occurs at 1500 bytes over Ethernet as per RFC 1191. Furthermore there has been very limited work on investigating the IPv6 performance bottleneck and to study of the effect of various OS stacks on system performance. This paper aims to shed light on that as well as discuss some of the issues and trade-offs between the two IP-stacks.

In this paper we address the following two research questions.

1. *What drawbacks do we have in implementing IPv6 on a Gigabit Ethernet network for UDP?*
2. *Which modern client/server OS provides the best system performance over IPv6 networks?*

To address the above questions we evaluate IPv6 in a peer-to-peer Gigabit Ethernet link for UDP using the four of the commonly used MS Windows and Linux-based client and server OSs. We measure UDP link throughput, round-trip time (RTT), jitter, and CPU utilization empirically to find the best OS over IPv6 networks.

The remainder of this paper is organized as follows. In Section II, we first review literature on IPv6 performance issues. The measurement procedure and experimental set up are discussed in Section III. The packet-generation and traffic-measuring mechanisms along with the evaluation methodology are also discussed. The experimental results and comparative analysis are presented in Section IV and a brief conclusion in Section V ends the paper.

II. A REVIEW OF LITERATURE

Manford et al. [9] evaluated the performance of UDP over IPv4 and IPv6 using two client-server networks running Windows XP-Windows Server 2008 and Windows Vista-Windows Server 2008. Throughput and RTT were measured using IP Traffic over a Fast Ethernet network. Their results have shown that the throughput difference between IPv4 and IPv6 for Windows Vista-Server 2008 was insignificant. Likewise, the throughput difference between IPv4 and IPv6 for Windows XP-Server 2008 was insignificant. They concluded that with the exception for the packet length of 384 bytes, there was negligible difference between the two OSs studied. However, Windows Vista-Server 2008 performed slightly better than Windows XP-Server 2008.

In 2010, Soorty et al. [10] evaluated the performance of IPv4 and IPv6 using Category 5e and Category 6 over a Gigabit Ethernet network. Network throughput and packet delays were measured to evaluate the system performance. Furthermore their study also took into account the additional parameters, such as packet length that can also affect the system performance. A detailed analysis reported considering various packet lengths ranging from 128 to 1408 bytes. They found that UDP achieved higher throughput using Category 5e than Category 6, thereby making Category 5e more suitable for data traffic. Packet delay was found to be lower with Category 6 than Category 5e as Category 6 provides about 12 dB (or 16 times) better Signal-to-Noise Ratio than Category 5e over a wide range of frequencies thus making Category 6 more preferable for delay-sensitive traffic [10].

Manford et al. [11] measured UDP performance over IPv4 and IPv6 on two peer-to-peer networks, namely Windows XP and Windows Vista. They focused on setting up a network commonly found in small-to-medium sized businesses (SMBs). It was found that UDP throughput ranged from 26.5 to 85.6 Mbps for IPv4 and 24.0 to 84.6 Mbps for IPv6 using Windows Vista. UDP throughput for Windows XP was found to be 21.8 to 86.0 Mbps for IPv4, and 20.7 to 81.0 Mbps for IPv6. Overall, Windows Vista performed slightly better than Windows XP.

In 2005, Wu et al. [12] evaluated IPv6 performance over a Gigabit Ethernet network. Unlike other studies, Wu's work focused on evaluating IPv4 and IPv6 over inter-domain routing areas and included a Multi-Protocol Label Switching (MPLS) backbone. Their work emphasized core routing and switching performance rather than the performance of enterprise network OS. However, UDP throughput found to be 349 Mbps for IPv4 and 339 Mbps for IPv6, whereas TCP throughputs found to be 144 Mbps and 141 Mbps for IPv4 and IPv6, respectively.

The above review of literature pertaining to UDP-IPv6 evaluation is summarized in Table I.

TABLE I KEY RESEARCHERS AND THEIR MAIN CONTRIBUTIONS IN EVALUATING UDP-IPV6

Researcher	Year	Performance evaluation
C. Manford et al. [9]	2010	Measured network throughput and packet delays in Fast Ethernet for UDP using Windows XP-2008 and Windows Vista-2008.
B.K. Soorty et al. [10]	2010	Measured network throughput and packet delays in Gigabit Ethernet for TCP and UDP using Windows Vista.
C. Manford et al. [11]	2009	Measured network throughput and packet delays in Fast Ethernet for UDP using Windows XP and Vista.
Tin-Yu Wu et al. [12]	2005	Measured network throughput and packet delays in Gigabit for TCP and UDP using Fedora Core 2.0.

We observe that most of the papers reviewed in this section considered network throughput and packet delays only. Performance metrics such as jitter and CPU utilization were not studied which may affect the performance of two IP stacks according to an earlier study by Zeadally et al. [1].

Our main contribution in this paper is to obtain new results by quantifying UDP-IPv6 performance degradation with respect to QoS parameters such as Throughput, RTT, jitter, and CPU utilization and to investigate which of the newest client/server OSs provide the best system performance.

III. TESTBED AND MEASUREMENT PROCEDURE

In this paper we study UDP-IPv6 performance using the latest Windows and Linux systems. Figure 1 shows network topology adopted which is basically a peer-to-peer Gigabit Ethernet client and server network. We did not use any routers or switches in the experimental setup to ensure that no additional delays experienced in the network due to intermediate devices, such as routers. Furthermore all services (running on default) consuming network bandwidth and/or CPU resources were disabled to get unbiased and more accurate results. No third-party applications were used to optimize or influence network performance in any way.

Each workstation was separated from the other by a distance of approximately one meter. This was done so as to maintain consistency with earlier study and thus produce results indicative for a fair comparison of the same [11]. The client and the server machines were connected using a Category 6 Crossover UTP (Unshielded Twisted Pair) cable maintaining EIA/TIA 568-B wiring configuration (Fig. 1).

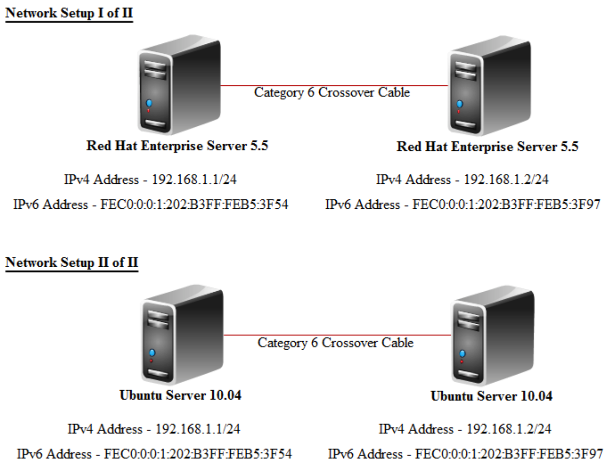


Figure 1: IPv6 evaluation test-bed for user-datagram protocol (UDP)

The hardware benchmark consisted of four workstations, all of which surpassed the minimum and recommended settings for the applicable OSs tested on them. Workstations 1 and 2 were each running under Windows 7 and Windows Server 2008 (Intel® Core™ 2 Duo processors with 4 GB 800 MHz DDR-2 Corsair® RAM modules). Workstation 3 was running for Ubuntu (Intel Core 2 Duo processor with 2 GB 800 MHz DDR-2 RAM modules). Workstation 4 was a Lenovo T40 laptop that was used for Red Hat Server after repeated tests revealed that the native hardware configuration met the satisfied recommended OS settings. Workstations were equipped with the Realtek PCIe GBE Family Controller Network Interface Cards (NICs) to carry out the Gigabit Ethernet evaluations on the network.

A. *Measurement Tools and Metrics*

Several data-generating and traffic-measuring tools were researched for the purpose of evaluating IPv6 on Windows and Linux OSs. For Windows, IP Traffic [13] was selected as the preferred tool due to its extensive history as a widely-used tool and for its overall accuracy in evaluating network performance. IP Traffic is also the only program to work on Windows 7 and Windows Server 2008. Furthermore, several publications such as [11] used IP Traffic to evaluate performance of IPv4 and IPv6. It has also been commonly used on wireless LAN evaluation [14, 15].

For Linux systems, a heavily modified tool of Iperf [16] was used to evaluate the performance of IPv4 and IPv6. Iperf is an open-source network performance measurement tool that can create TCP and UDP data streams and measure the throughput and RTT. Iperf was used by Narayan et al. [17] as the primary evaluation tool for IPv6 performance study. Furthermore, for better comparison the settings in Iperf were modified to match the settings in IP Traffic.

IPv6 traffic was sent, measured and recorded separately over a different session to IPv4. Both tests were done in isolation from each other using the same test-bed. For each evaluation-run performed, IP Traffic sent a total of one million packets. Packets were sent from the source machine and received at the destination machine. Due to the nature of the UDP protocol where no acknowledgement is transmitted back to the source, the timestamp was recorded at the destination to measure overall throughput and RTT. Ten such runs were recorded per protocol for each Windows-based OS. A total of 10 million packets were thus sent before each protocol's throughput and delay was recorded for every packet-size using a particular Windows based OS. A similar approach was used for recording performance of each Linux based OS using Iperf. This was done in-order to maintain accuracy and consistency of results. A standard deviation of less than 10% was implemented to accurately measure overall system performance.

IV. RESULTS AND COMPARATIVE ANALYSIS

We measure throughput, RTT, jitter and CPU usage to evaluate UDP-IPv6 performance using Windows and Linux client/server OSs. We gradually increased the data packet lengths from 128 to 1408 bytes to observe the impact of packet lengths on system performance.

A. *Throughput Performance*

In Fig. 2, we plot packet lengths against UDP throughput for both IPv4 and IPv6 using Windows and Linux client-server OSs for packet lengths of 128, 384, 640, 896, 1152, and 1408 bytes. We observe a steady increase in throughput with packet length. This is because the larger packets can carry more payloads and they require less transfer to move the data from the source to the destination. We observe that the highest throughput is achieved using Ubuntu -Red Hat Server client-server network than Windows Server 2008. For example, for IPv4 using Ubuntu-Red Hat Server the throughput is 774.46 Mbps (6.5% higher than Windows Server 2008.) at packet length of 1408 bytes. For IPv6, the highest throughput is achieved (725 Mbps) for Ubuntu-Red Hat Server. This is about 7.5% higher than the throughput obtained under Windows Server 2008 at packet-length of 1408 bytes. We now quantify the throughput degradation of IPv6 (see Table II).

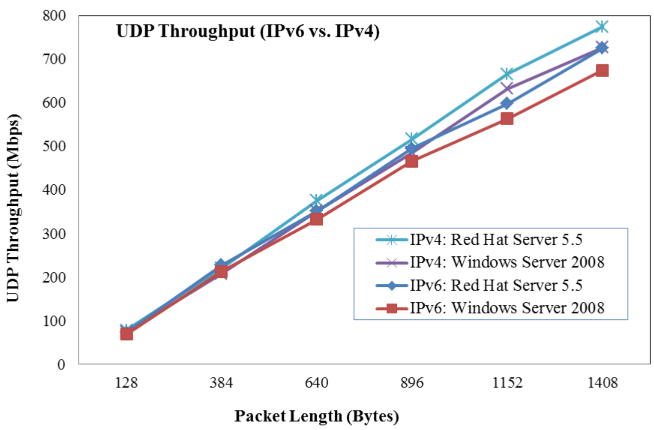


Figure 2: UDP Throughput comparison of IPv6 and IPv4 on Windows and Linux client-server OSs

Table II compares mean UDP throughput for IPv6 and IPv4 using both Windows and Red Hat Servers. The mean throughputs were computed by taking average of all the throughputs obtained at packet lengths of 128, 384, 640, 896, 1152, and 1408 bytes. We found that IPv4 offers about 5.9% higher throughput than IPv6 for both Windows and Red Hat Servers. By comparing both OSs, one can observe that Red Hat Server is achieved about 5.7% higher throughput than Windows Server for both IPv6 and IPv4. UDP throughput is higher on IPv4 than IPv6 using both Windows and Linux systems. This is mainly due to IPv6's larger header which is about six times larger than the IPv4 header.

TABLE II : UDP THROUGHPUT OF IPV6 AND IPV4 FOR WINDOWS SERVER 2008 AND RED HAT SERVER 5.5

Operating System	**Mean UDP Throughput (Mbps)**		**IPv4 achieves higher throughput than IPv6 (%)**
	IPv4	***IPv6***	
Ubuntu - Red Hat Server	438.68	411.83	6.1
Windows 7 - Server 2008	412.92	386.33	6.4
Red Hat Server achieves higher throughput (%)	5.9	6.2	

B. RTT Performance

The second metric measured and evaluated in this paper was packet delay. UDP packets were sent across the IP networks and round trip time was measured in milliseconds.

Fig. 3 compares RTT for both IPv4 and IPv6 over each client-server network. On Windows 7 Server 2008, the lowest latency is 1.27 ms for both IPv4 and IPv6. The highest delays are 4.67 ms and 4.09 ms, for IPv4 and IPv6 respectively. For example, on Ubuntu-RHES, the lowest RTT is 1.37 ms for IPv4 and 1.35 ms for IPv6. The highest latency is 4.92 ms for IPv4 and 4.33 ms for IPv6 at packet length of 1408 bytes on Ubuntu-RHES compared to 4.67 ms for Windows Server 2008 also recording at the largest packet-size of 128 bytes.

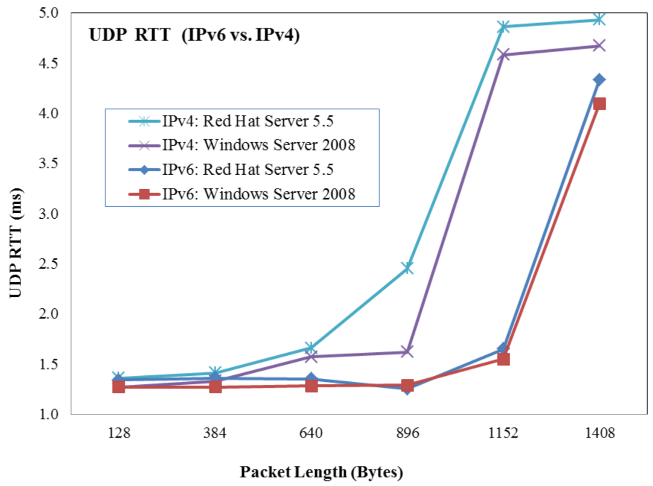


Figure 3: UDP RTT comparison of IPv4 and IPv6 on Windows and Linux client-server OSs

Despite Ubuntu-RHES producing a significantly higher throughput for UDP than using Windows 7-Server 2008, the difference in UDP delay between the two networks is considerably low. This is probably due to Ubuntu having a more efficient socket layer that responds faster to kernel switches during system calls. We also observe that the lowest delay is achieved for shorter packet length. This is mainly due to throughput being comparatively lower on smaller packets. With UDP having no error-correction mechanism there is no significant overhead in relation to the payload and therefore no restriction in throughput from reaching its maximum. This therefore results in relatively lower delay compared to larger packets. As shown in Fig. 3 and Table III, the RTT is significantly smaller for IPv6 than IPv4 for all three networks.

TABLE III: UDP RTT FOR IPV6 AND IPV4 ON WINDOWS AND LINUX SYSTEMS

Operating System	**RTT (ms)**		**IPv6 achieves lower RTT (%)**
	IPv4	***IPv6***	
Ubuntu - Red Hat Server	2.78	1.88	32.4
Windows 7 - Server 2008	2.51	1.79	28.7
Windows achieves slightly lower RTT (%)	9.7	4.8	

C. Jitter and CPU usage Performance

The third metric measured and evaluated in this paper was jitter. UDP datagrams were sent across the IP networks and jitter was measured at the receiving node (server) for each network scenario.

Fig. 4 compares UDP jitter for IPv4 and IPv6 over the two client-server networks. UDP jitter is lower on Windows 7-Server 2008 for both, IPv4 and IPv6.

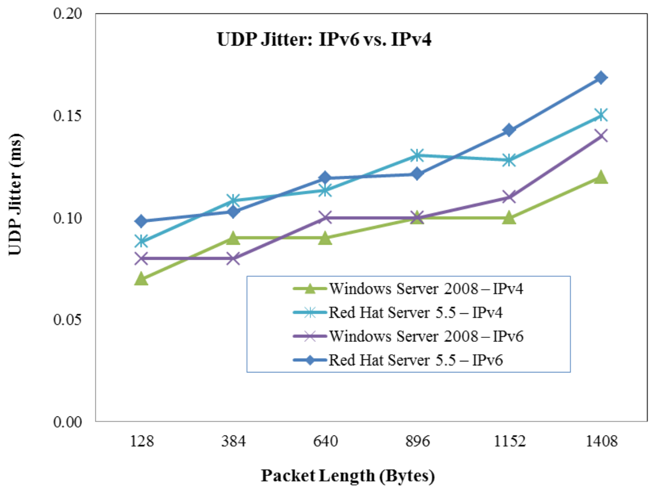


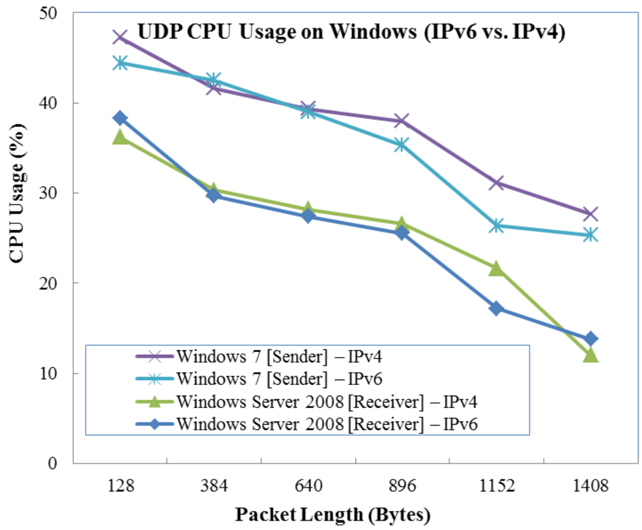
Figure 4: Jitter comparison of Windows and Linux client-server operating systems for IPv4 and IPv6

The fourth metric measured and evaluated in this paper was CPU utilization for transferring data across the network (client host). Table IV shows CPU usage for transferring UDP packets using Windows and Linux OSs.

TABLE IV: UDP CPU USAGE FOR IPV6 AND IPV4 USING UBUNTU AND WINDOWS 7

Operating System	UDP CPU Usage (%)		IPv6 uses less CPU processing power (%)
	IPv4	*IPv6*	
Ubuntu - Red Hat Server	23.8	20	16
Windows 7 - Server 2008	27.7	25.3	8.7
Ubuntu uses less CPU processing power (%)	14.1	20.9	

As shown in Fig. 5, we observe that CPU utilization is slightly lower with Ubuntu than it is on Windows 7 for IPv4 and IPv6. For example, minimum utilization for Ubuntu is 23.80% for IPv4 and 19.97% for IPv6 compared to 27.66% for IPv4 and 25.34% for IPv6 on Windows 7.



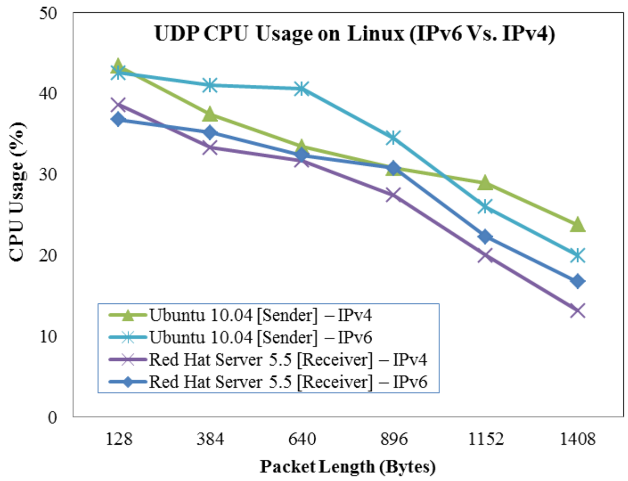


Figure 5: Comparison of UDP CPU Usage for IPv6 and IPv4 using Windows and Linux client/server networking

D. Summary of findings and practical implications

The results presented in the previous section provide some insight into the impact of IPv6 deployment on Gigabit Ethernet and the choice of modern client/server OS that provides the best system performance.

Table V summarizes the performance of IPv6 and IPv4 for UDP over Windows and Linux client/server networks.

TABLE V : SUMMARY OF UDP-IPV6 EVALUATION RESULTS

OS	Throughput		RTT	
	IPv6	*IPv4*	*IPv6*	*IPv4*
Linux		best	better	
Windows		better	best	

OS	Jitter		CPU Usage	
	IPv6	*IPv4*	*IPv6*	*IPv4*
Linux		better	best	
Windows	←No Difference→		better	

Note:

1. **IPv6 vs. IPv4:** IPv6 achieved about 6% lower throughput than IPv4. However, IPv6 is better in achieving about 30% lower RTT than IPv4.

2. **Linux vs. Windows:** Red Hat Server achieved about 6% higher throughput than Windows Server 2008. However, Windows Server 2008 achieved approximately 5% lower RTT than Red Hat Server for IPv6.

We observe the following system performance characteristics:

1. **Throughput (IPv6 vs. IPv4):** IPv4 achieved approximately 6% higher throughput than IPv6 for both Windows and Red Hat Servers (IPv4 is better). Red Hat Server achieved about 6% higher throughput than Windows Server for both protocols (Red Hat Server is better).

2. **RTT (IPv6 vs. IPv4):** IPv6 achieved about 30% lower RTT than IPv4 for both OSs (IPv6 is better). Windows Server achieved about 5% lower RTTs than Red Hat Server for IPv6 (Windows Server is better).

3. **Jitter (IPv6 vs. IPv4):** IPv4 obtained slightly lower Jitter than IPv6, especially for larger packet length for both OSs (IPv4 is better). Windows Server achieved about 20% lower Jitter than RHES for both protocols (Windows Server is better).

4. **CPU Usage (IPv6 vs. IPv4):** IPv6 consumed about 3.5% less CPU power than IPv4 for both Windows 7 and Ubuntu (IPv6 is better). Ubuntu consumed about 10% less CPU power than Windows 7 (Ubuntu is better).

V. CONCLUSION

In this paper we studied UDP-IPv6 performance in a peer-to-peer Gigabit Ethernet link using modern MS Windows and Linux operating systems. Our findings based on the empirical study are summarized below.

- UDP throughput for IPv6 was highest on the Linux network running Ubuntu with RHES. Certainly, Ubuntu-RHES offers an average throughput of 411.83 Mbps compared to 386.33 Mbps on Windows 7-Server 2008 network.
- UDP throughput for IPv4 was also highest on the Linux network running Ubuntu with RHES (Ubuntu-RHES offers an average throughput of 438.68 Mbps compared to 412.92 Mbps on the Windows 7 Server 2008).
- UDP average RTT was lowest on Windows 7-Server 2008 for IPv4 (2.51 ms compared to 2.78 ms for Ubuntu-RHES). For IPv6, RTT was also lowest on Windows 7-Server 2008 (1.79 ms compared to 1.88 ms for Ubuntu-RHES).
- UDP jitter was lowest on Windows 7-Server 2008 than Ubuntu-RHES.
- CPU utilization to transfer UDP packets across the network was more efficient over Ubuntu compared to Windows 7-Server 2008.

We observed that the Linux network running Ubuntu with RHES performed significantly better on IPv6 than the Windows network running Windows 7 with Windows Server 2008. Even though RTT and jitter were comparatively higher on Ubuntu-RHES, but the overall performance was better on Linux due to its significant throughput gain.

REFERENCES

[1] S. Zeadally and L. Raicu, "Evaluating IPv6 on Windows and Solaris," IEEE Internet Computing, vol. 7, no. 3, pp. 51-57, 2003.

[2] N. I. Sarkar and B. Soorty, "Evaluating IPv6 in Peer-to-Peer Gigabit Ethernet for TCP using Modern Windows and Linux Systems: An Empirical Study," International Journal of Business Data Communications and Networking, vol. 9, no. 1, pp. 50-63, 2013.

[3] A. A. Mahimkar, H. H. Song, Z. Ge, A. Shaikh, J. Wang, J. Yates, Y. Zhang, and J. Emmons, "Detecting the performance impact of upgrades in large operational networks," SIGCOMM Computer Communication Review, vol. 40, no. 4, pp. 303-314, 2010.

[4] M. Mackay and C. Edwards, "A Managed IPv6 Transitioning Architecture for Large Network Deployments," IEEE Internet Computing, vol. 13, no. 4, pp. 42-51, 2009.

[5] P. Savola, "Observations of IPv6 traffic on a 6to4 relay," SIGCOMM Computer Communication Review, vol. 35, no. 1, pp. 23-28, 2005.

[6] L. Sang-Do, S. Myung-Ki, and K. Hyoung-Jun, "The implementation of ISATAP router," presented at the 8th International Conference Advanced Communication Technology (ICACT'06) Phoenix Park, Gangwon-Do, Korea, 20-22 February, 2006, pp. 1163-1165.

[7] U. Narayanan and X. Jiang, "Signaling Cost Analysis of Handoffs in a Mixed IPv4/IPv6 Mobile Environment," presented at IEEE Global Telecommunications Conference (GLOBECOM '07), Washington, DC, November 26-30, 2007, pp. 1792-1796.

[8] M. Kuhne, "The Seven Stages of IPv6 Adoption," IETF Journal, vol. 5, no. 1, pp. 14-17, 2009.

[9] C. Manford, S. S. Kolahi, B. K. Soorty, Z. Qu, and N. Chand, "UDP Performance Comparison of IPv4 and IPv6 using Windows Vista-Server2008 and Windows XP-Server2008 LANs," presented at Advances in Computer Science and Engineering (ACSE 2010), Sharm El Sheikh, Egypt, March 15-17, 2010, pp.

[10] B. K. Soorty, S. S. Kolahi, N. Chand, and Q. Zhang, "Performance Comparison of Category 5e vs. Category 6 Cabling Systems for both IPv4 and IPv6 in Gigabit Ethernet," presented at the IEEE 10th International Conference on Computer and Information Technology, Bradford, UK, June 29-July 1, 2010, pp. 1525-1529.

[11] C. Manford, S. S. Kolahi, B. K. Soorty, Q. Zhang, and N. Chand, "UDP Performance Analysis of IPv4 and IPv6 on Windows Vista and Windows XP over Fast Ethernet Peer-Peer LAN," presented at the 4th International Conference onUbiquitous Information Technologies & Applications, Fukuoka, Japan, December 20-22, 2009, pp. 1-4.

[12] T.-Y. Wu, H.-C. Chao, T.-G. Tsuei, and Y.-F. Li, "A measurement study of network efficiency for TWAREN IPv6 backbone," International Journal Of Network Management, vol. 15, no. 6, pp. 411-419, 2005.

[13] Z. Telecom. (2014). IP Traffic - test & measure. Retrieved March 20, 2014, from http://www.zti-telecom.com

[14] N. Baghaei and R. Hunt, "IEEE 802.11 wireless LAN security performance using multiple clients," presented at the 12th IEEE International Conference on Networks Singapore, 16-19 November, 2004, pp. 299-303.

[15] S. S. Kolahi, S. Narayan, D. D. T. Nguyen, Y. Sunarto, and P. Mani, "The impact of wireless LAN security on performance of different Windows operating systems," presented at IEEE Symposium on Computers and Communications, Marrakech, Morocco, July 6-9, 2008, pp. 260-264.

[16] Jdugan and Mitchkutzko. Iperf. Retrieved March 14, 2014, from http://sourceforge.net/projects/iperf/

[17] S. Narayan, S. S. Kolahi, Y. Sunarto, D. Nguyen, and P. Mani, "Performance comparison of IPv4 and IPv6 on various windows operating systems," presented at the 11th International Conference on Computer and Information Technology Khulna, Bangladesh, December 24-27, 2008, pp. 663-668.