Performance Analysis of 4to6 and 6to4 Transition Mechanisms over Point to Point and IPSec VPN Protocols

Shaneel Narayan, Salman Ishrar, Avinesh Kumar, Ruchinav Gupta, Ziafil Khan
Unitec Institute of Technology
Auckland, New Zealand
Email: snarayan@unitec.ac.nz

***Abstract*—This paper provide an overview of behaviour of transition mechanisms, with and without VPN Protocols. Performance metrics related to networks have been gathered from test-bed implementations. The two transition mechanisms we will be evaluating are 4to6 and 6to4. Both of these mechanism have certain advantages and disadvantages. VPN protocols, PPTP and IPsec were configured on the transition mechanism and compared with selected networking metrics. The key networking metrics that were captured in this research were throughput,delay,jitter,DNS throughput,DNS delay, DNS jitter, for both TCP and UDP protocols. VoIP throughput was also measured and discussed. The testbed consisted of two (IPv4/IPv6) capable Cisco routers, and two machines which were running Windows 7 and Windows Server 2012.**

Keywords - *Transition mechanism, IPv4, IPv6, 4to6, 6to4, PPTP, IPsec, VPN, performance evaluation.*

I. Introduction

With the exhaustion of IPv4 addressing space rapidly approaching, the Internet Engineering Task Force (IETF) developed the IPv6 protocol so more hosts could be accommodated. When compared to IPv4, IPv6 provides additional capabilities as well as a more simplified header, and other improvements such as, flow labelling (true quality of service) and built in authentication and privacy support. IPv4 is limited to 32 bit (four-byte) addresses, which amounts to 4,294,967,296 IPv4 addresses, and as they have been assigned to users, overtime the overall number of IPv4 addresses have slowly depleted. It is difficult to migrate from IPv4 to IPv6, as IPv4 has been around almost 35 years and transition cannot happen overnight as the transition process is complex. To support the transition between IPv4 and IPv6, IETF developed various transition mechanisms. The purpose of these transition mechanisms is to allow interoperability between IPv4 networks and IPv6 networks. In short, transition mechanisms' sole purpose is to encapsulate IPv4 packets and transport them over IPv6 network infrastructure and the other way around. There are three main transition mechanism methods: dual-stack, tunnelling and translation. We will be discussing 4to6 and 6to4 transition mechanisms which fall under the tunnelling category.

This paper will showcase and discuss key performance metrics such as throughput, delay, jitter,DNS and VoIP metrics for IPv4, IPv6, and 4to6 and 6to4 transition mechanisms, with and without VPN protocols. The two VPN protocols that will be used alongside the transition mechanisms are PPTP (Point to Point Tunneling Protocol) and IPsec (Internet protocol security). The main objective of our study was to evaluate two transition mechanisms working alongside two VPN protocols and assess the overall end to end network performance. We will refer to the term router to router tunnelling throughout the paper which means IPv6 to IPv4, or IPv4 to IPv6 tunnelling encapsulation at the routers.

Section two will discuss the background information in regards to the main differences between IPv4 and IPv6. We will briefly overview the various transition mechanisms, and review router to router tunnelling for 4to6 and 6to4 transition mechanisms. In section three, we will discuss the test-bed setup used for these networks and the tools used to monitor the networks. Section four will explain our experimental results which are showcased by graphed data, and finally in section five we draw conclusions and make our final remarks on our findings.

II. Background

IPv4 is the current version of the Internet Protocol and was developed in the early 1970s. It was the most dominant standard network layer protocol used to exchange data over the Internet. Due to the fact that IPv4 uses 32-bit addresses, it is limited to only roughly 4.3 billion addresses. This address space is insufficient for the future as it is depleting at a rapid rate, and the five regional Internet Registries have urged companies and users to migrate to IPv6 as soon as possible. As we cannot move between IPv4 to IPv6 in a short time-space, it was essential that the development of a mechanism that supports both IPv4 and IPv6 was created and would last at least 15-20 years during the transition period, in which IPv4 will eventually totally deplete as mentioned in [1]. IPv6 was developed as a successor to IPv4 and its primary focus was to provide a larger IP address space to move onto, before IPv4 addresses completely run out. The key advantages IPv6 has over IPv4 are increased performance, enhanced security, better QOS (Quality of service)[2], and enhanced scalability. IPv6 uses 128 bit addresses as opposed

978-1-4673-8975-4/16/$31.00 ©2016 IEEE

to IPv4s 32 bit addresses, meaning IPv6 can provide $3.4X10^{38}$ [4] number of addresses, which easily accommodates more users for the future.

There are numerous transition mechanisms available as discussed in [3], such as NAT64, ISATAP, 6RD, 6to4, 4to6, and 6 over 4. Nat64 can only be configured with IPv6 hosts and it is not the most popular transition mechanism as it has some DNS translation issues. 6RD specifies a protocol to deploy IPv6 sites via an Internet Service Providers', (ISP) IPv4 network and it is built on 6to4 transition mechanism with the difference being it uses ISPs' IP. ISATAP is an automatic tunnelling protocol which transmits IPv6 packets between Duals Stacks hosts over an IPv4 network [10]. 6to4 transition mechanism, allows IPv6 data to be transported across IPv4 network[4]. 6to4 is the most widely used tunnelling technique. In a 4to6 transition mechanism, IPv4 hosts communicate over IPv6 network infrastructure. For this research, we conducted studies on 4to6 and 6to4 transition mechanisms. 6to4 and 4to6 [5] both encapsulate the packets at the router and we will refer to it as router to router encapsulation.

Tunneling 4to6 allows communication from one IPv4 site to another IPv4 site over an IPv6 network without the need for a configured tunnel or IPv6 compatible IPv4 addresses. In a 4to6 transition mechanism, routers are configured to encapsulate IPv4 packets into an IPv6 packet and tunnel it over to the other router, which decapsulates it from IPv6 to IPv4 packets and sends it to the IPv4 host [11]. Fig. 1 outlines

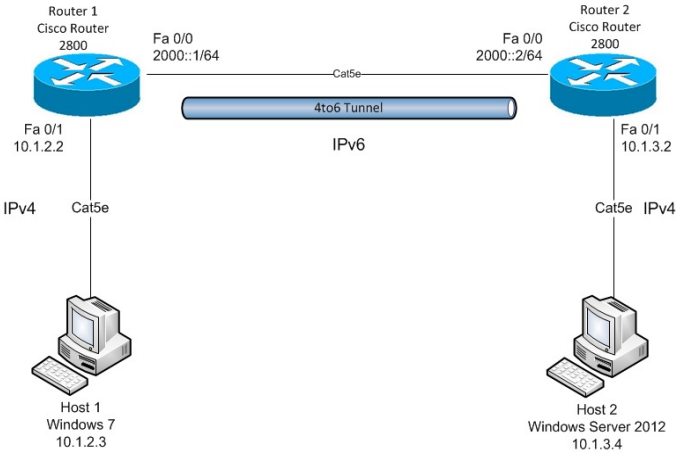

Fig. 1: 4to6 Network Diagram

an IPv4 packet being sent to router one which is encapsulated to an IPv6 packet between Fast Ethernet 0/1 to 0/0 and forwarded to router 2 using the IPv6 tunnel; packets are then decapsulated to a IPv4 packet and forwarded to host two.

Tunneling 6to4 is an automatic transition mechanism for establishing a tunnel that is used to provide connectivity between IPv6 nodes via an IPv4 network. In this tunnel setup there is no end node configuration and router configuration is minimal. The 6to4 transition mechanism is the most used tunneling technique and can be configured to use IPv6 prefix addresses as well as IPv4 addresses. A 6to4 transition mechanism makes the transition from an IPv6 network to an IPv4 network smooth.

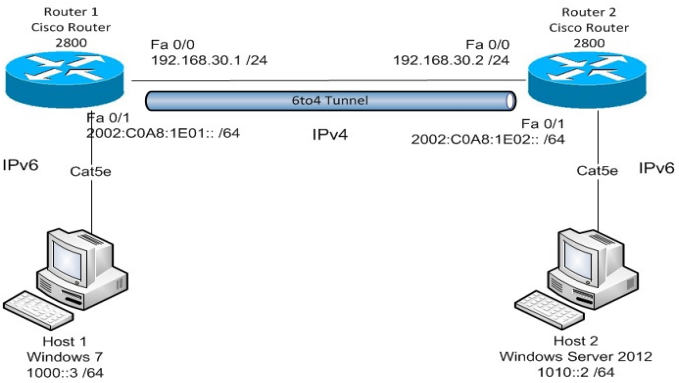

Fig. 2: 6to4 Network Diagram

Fig. 2 above shows an IPv4 tunnel which is configured on interface 0/0 to have IPv6 prefixes 2000::1/64 and 2000::2/64 on the inside and outside, with IPv4 addresses where translation happens from IPv4 to IPv6 before packets are forwarded to the next hop. In this tunnelling setup the IPv6 host sends packets over the IPv4 network to another IPv6 host. The packets are encapsulated at the edge of router1 from IPv6 to IPv4 packet and router two decapsulates IPv4 packets to IPv6.

Virtual Private Network (VPN) [6] is a technology which secures public networks such as the Internet or a private network. A VPN ensures a high level of security by encrypting packets from one end to the other. Also mentioned in [9], VPNs eliminate the need for expensive long distanced leased lines used for private networks and can be configured on cisco routers or software routers which makes it very cost effective. We implement the following VPN protocols among our network:

Point-to-Point Tunneling Protocol (PPTP) is a virtual private network protocol that is used by businesses and companies to extend their own corporate network over the public Internet via private tunnels. PPTP can also be used for site-to-site VPN connections and also for remote access. This protocol allows the encryption of multi-protocol traffic, then encapsulates it in the IP header which is sent across the Internet. PPP frames are encapsulated in the IP datagrams which are then transmitted over the network.

Fig. 3 below shows the PPTP packet which contains the IP datagram.

Using encryption keys which are generated from the MS-CHAP v2 or the EAP-TLS authentication process by the

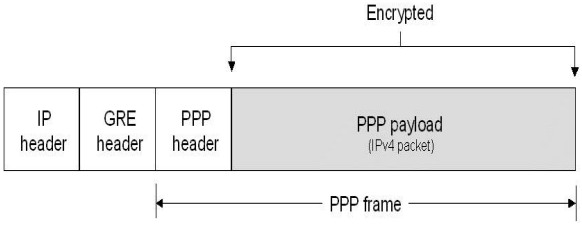

Fig. 3: PPTP Packet containing IP datagram

Microsoft Point-to-Point Encryption (MPPE)[7] to encrypt the PPP frame. All VPN clients must use the MS-CHAP v2 or EAP-TLS authentication protocol to encrypt the payloads of the PPP frames.

Internet Protocol Security (IPsec) is the main structure for a group of protocols for security purposes, which sits on top of the Internet Protocol (IP) layer. IPsec [8] offers two choices of security levels, which are Authentication Header (AH) and Encapsulating Security Payload (ESP). AH allows authentication of the sender of the data and the second security service, which ESP supports the encryption of the data as well as the authentication of the sender of the data. The information associated with both the security services is put into the packet of the header, which follows the IP packet header. The following Fig. 4 shows the IPsec packet with encryption and authentication.

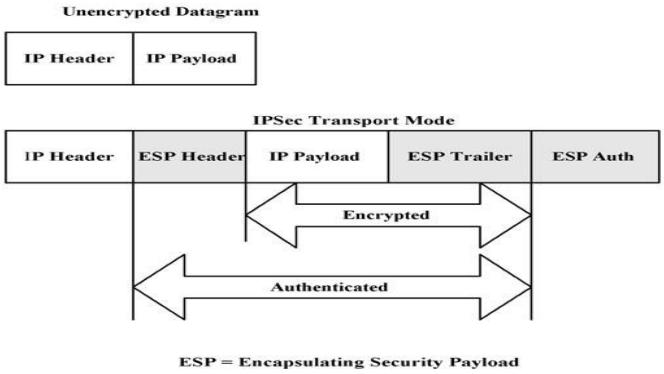

Fig. 4: IPsec packet with encryption and authentication

III. Testbed Setup

Fig. 5 depicts the testbed used in this research. The testbed consisted of two computers (Intel Core i7-4770 CPU 3.40 GHz, 8GB RAM) connected via TP Link Gigabit Switch (1000Mbps) using Cat5e cables (1000Mbps); both computers had one network interface card (Intel 1000 GT Network Adaptor). There were two Cisco 2800 series integrated routers, IOS version 12.4. Host 2 hosted a Windows Server 2012 operating system, which was used to send network traffic between the two routers through the transition mechanism, to host one, which was a receiver machine running Windows 7. For our testing to be concise and consistent and to achieve

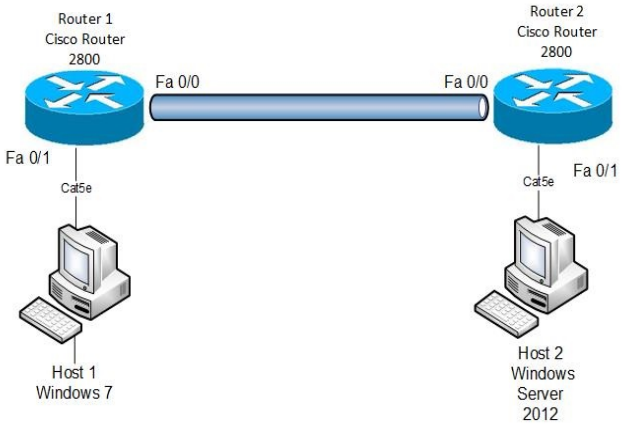

Fig. 5: Testbed setup

optimal results, we kept all the hardware the same. Initially IPv4 and IPv6 were tested, followed by 4to6 and 6to4 without any VPN protocols. The 4to6 transition mechanism was configured with PPTP, followed by 6to4 configured with IPsec.

There are various performance monitoring tools available which are used to generate traffic and analyse networks. Different tools provides different test results and in [7] a study was conducted on four of these tools: Iperf, IP traffic, Netperf, and D-ITG. D-ITG [8] proved to be the most reasonable performance monitoring tool. D-ITG (Distribued Internet Traffic Generator) measures traffic using two components ITGSend and ITGRecv and is capable of sending traffic of various packet sizes. To secure accurate data all tests were executed six times for 20 seconds at a given packet size, from the range of 128bytes upto 1536 bytes. We analysed our network's overall performance using D-ITG. In [10-14], similar test-beds have been used by the authors.

IV. Results and Discussion

The graphs below showcase overall UDP and TCP traffic throughput,delay, jitter, DNS and VoIP for IPv4/IPv6 vs multiple transition mechanisms, with and without VPN.

Fig. 6 showcases the overall TCP throughput,TCP delay and TCP Jitter for IPv4/IPv6 vs multiple transition mechanisms with and without VPN. There is a big throughput difference for IPv4 and IPv6. For packet size 384, IPv4 has a throughput of 74 Mbps, whereas IPv6 peaked of 89 Mbps. There was a slight increase for IPv4 at a bigger packet size, 1536, giving it its maximum throughput of 84 Mbps while IPv6 recorded 90 Mbps. This suggests that IPv6s TCP throughput is faster than IPv4. The 6to4 transition mechanism gave a similar result as IPv6 with hardly any variations. However, when IPsec was configured on the 6to4 transition mechanism the throughput vastly dropped. All the values recorded on 6to4 with IPsec configured gave a throughput of 4 Mbps, which is a 95.506 drop compared to the 6to4 transition mechanism. The

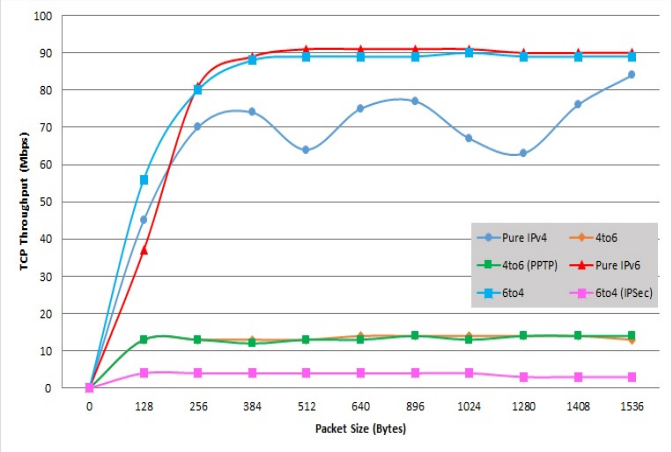

Fig. 6: TCP Throughput

transition mechanism 4to6 gave a consistent throughput for all packet values with a throughput of 14 Mbps. Throughput values for both 4to6 and 4to6 with PPTP produced a similar result with no major difference in throughput.

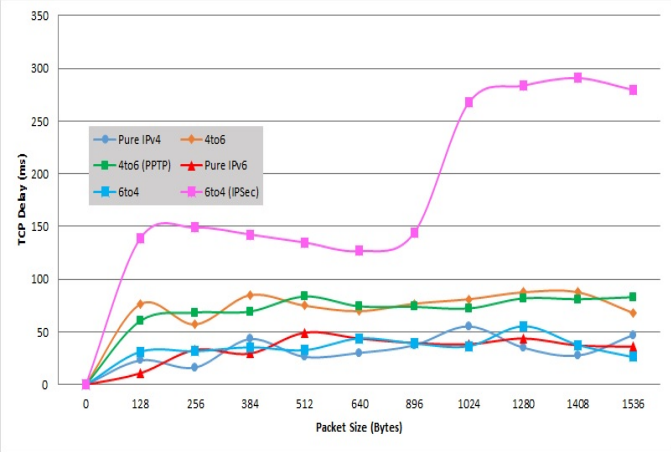

Fig. 7: TCP Delay

Fig. 7, above, displayed the average overall TCP delay between IPv4, IPv6, and 6to4, 4to6 transition mechanisms with and without VPN protocols. Although IPv4 and IPv6 delay displayed similar consistency, by averaging and computing the delay figures can confirmed that IPv6 had 5.8% higher delay. For the 6to4 transition mechanism, there were no significant changes between packet sizes 128 and 1536 bytes. TCP delay fluctuated between 25ms and 45ms. IPsec configured on the 6to4 transition mechanism had 128 bit encryption, hence the delay was significantly higher when compared to the 6to4 transition mechanism without VPN. It is worth noting that delay for 6to4, with IPsec, increased rapidly between packet sizes 896 and 1024 by a total of 85.80%. The transition mechanism 4to6, with and without VPN protocol, showed little variation delay between packets 128 to 1408. The highest delay for 4to6 was at packet 1408, with a delay of 87.6ms, as opposed to 4to6 PPTPs delay at packet size 512 which had a delay of 83.8ms.

Fig. 8 shows the average TCP jitter between IPv4/IPv6, 4to6, 4to6 with PPTP, 6to4, and 6to4 with IPsec transition mechanism.

All the values measured for Pure IPv4 and Pure IPv6 displayed no difference for jitter. The lowest jitter for both was recorded at packet size 128, averaging 0.042ms, and was highest at 1280Bytes with 0.4ms. Transition mechanisms 4to6 and 4to6 with PPTP showed some major increases in jitter compared to IPv4. At the packet size 512 and 896 there was a slight decline of 0.08ms. The 6to4 transition mechanism had a lower jitter than Pure IPv4 from packet sizes 640 to 1408. IPsec on 6to4 gave it a very high jitter, at packet size 1536, and 6to4 had a jitter of 0.24ms; with the IPsec protocol jitter was 5.8ms, which is a 5.5ms difference. As shown in Fig. 8, 6to4, IPv4 and IPv6 had the lowest jitter and 6to4 with IPsec produced the maximum jitter.

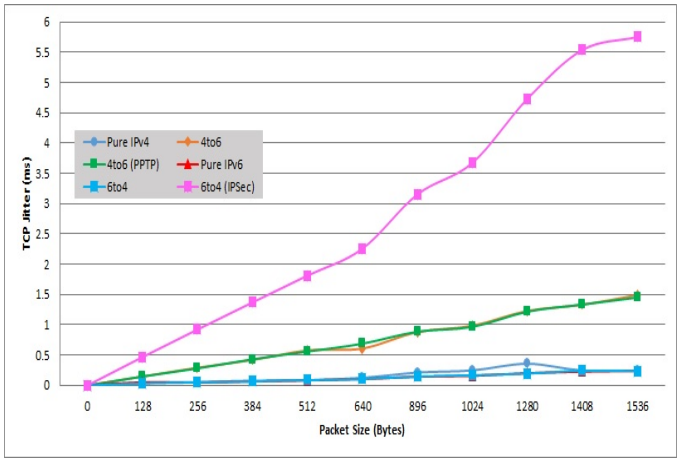

Fig. 8: TCP Jitter

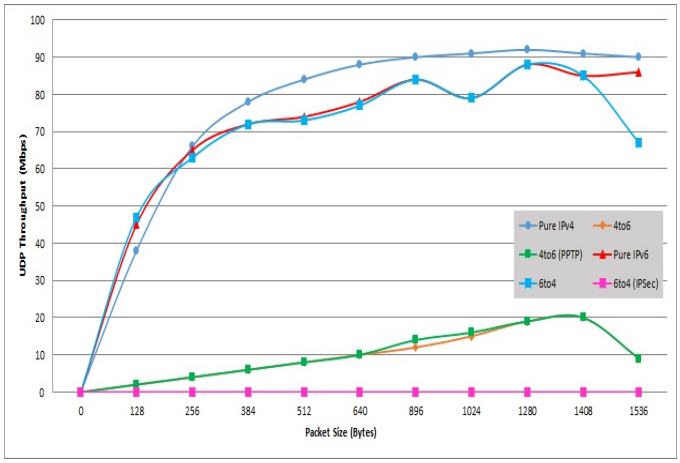

Fig. 9: UDP Throughput

Fig. 9 showcases the overall UDP throughput, UDP delay, and UDP jitter for IPv4/IPv6 vs multiple transition mechanisms, with and without VPN.

As shown in this graph, the UDP throughput for IPv4 increased constantly at higher packet values from packet size 128 to 1280 bytes, with a minimal drop from packet size 1280 onwards. Maximum throughput measured for UDP on IPv4 was 92 Mbps with packet size 1280. Throughout all the values IPv6 and 6to4 transition mechanism were comparable as there was not much variation in throughput until packet size 1408, where the throughput for the 6to4 transition mechanism dropped vastly from 88Mbps to 67Mbps. IPsec traffic on the 6to4 transition mechanism could not be measured. Transition mechanisms 4to6 and 4to6 with PPTP had got the lowest throughput overall in comparison with 6to4 and IPv4. IPv4 at 1408 had a throughput of 85Mbps and 4to6, with and without PPTP, was 20Mbps. The average UDP delay for IPv4 and

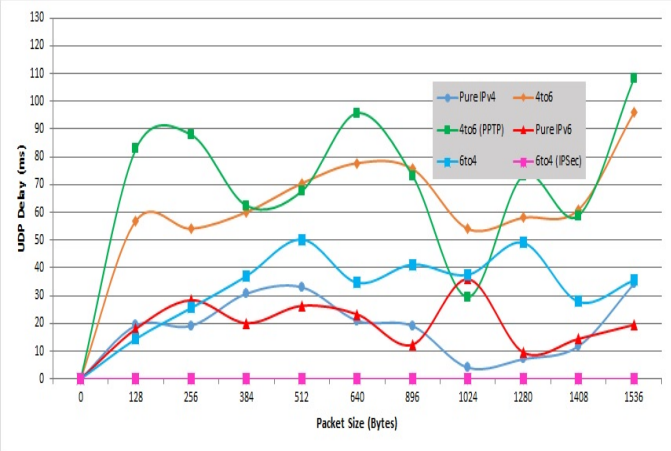

Fig. 10: UDP Delay

IPv6 was similar, as plotted in Fig. 10. There was a gradual increase at packet size 1024 for IPv6 otherwise delay remained the same until packet size 1048. At 1408Bytes, IPv4s delay increased by 13ms in comparison with IPv6. For transition mechanism 6to4, the delay increased as the packet sizes got larger when compared with IPv4 and IPv6. The highest delay was recorded for 4to6 transition mechanism and the 4to6 with PPTPs delay was slightly higher than 4to6.

Fig. 11 shows the UDP jitter for all the networking protocols measured using D-ITG. IPv4, IPv6 and 6to4 jitter results illustrates the same pattern. There was a slight increase in jitter for 6to4 between packet sizes 1408 and 1536 bytes. 6to4 with IPsec could not be monitored for UDP performance, so it has been plotted as 0. Transition mechanism 4to6, with and without PPTP gave consistent results in jitter.

Whilst analyzing Fig. 12, the DNS throughput we discovered that the highest throughput shown was 1.08Kbps on Pure IPv4 at packet size 1408. When looking at pure IPv6, we saw the highest throughput at packet 1408 was lower than IPv4, being 1.06Kbps. 4to6 showcased higher DNS throughput when compared with 4to6 PPTP, the difference in overall throughput almost reaching 0.089Kbps extra. Worth noting was that the

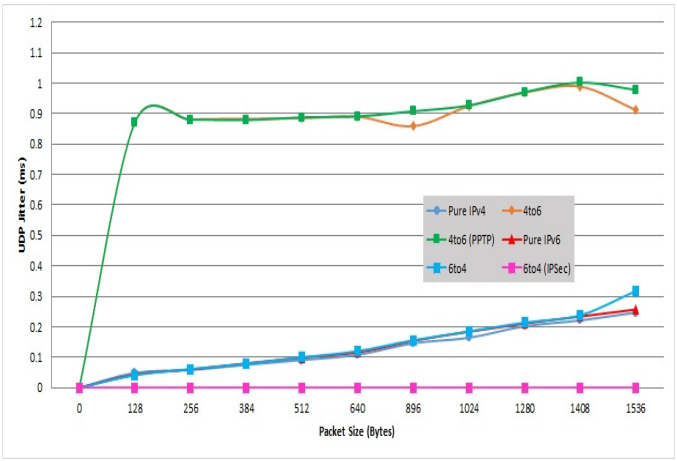

Fig. 11: UDP Jitter

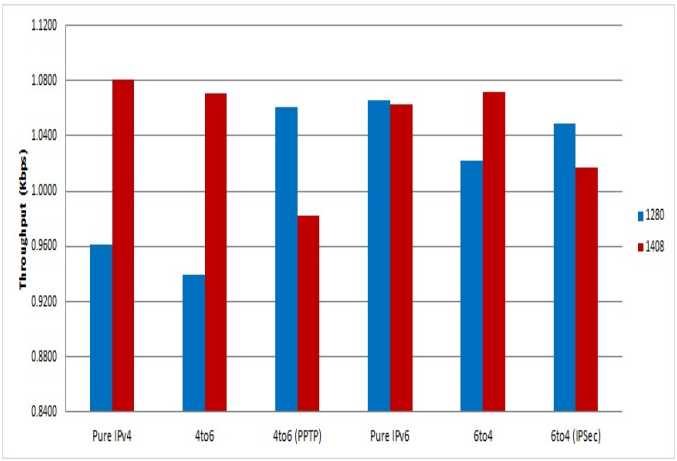

Fig. 12: DNS TCP Throughput

packet value 1208, had higher DNS throughput with 4to6 PPTP than 4to6 on its own. The same behavior was shown when comparing 6to4 and 6to4 configured with IPsec. 6to4 on its own displayed high DNS throughput, being 1.07Kbps almost matching Pure IPv4, whilst 6to4 (IPsec) had higher DNS throughput at packet 1280, as opposed to packet 1408 which displayed the highest overall throughput.

Fig. 13 depicts that the highest delays experienced were on Pure IPv4 and 4to6 networks, with the delays being 38ms for IPV4 and 40ms for 4to6. Pure IPv6 did encounter less latency when compared to Pure IPv4 at both packet sizes. 6to4 displayed the least amount of latency (6.22ms) for packet size 128, whilst for packet size 1536 Pure IPv6 had the lowest latency being 12.6ms. Between both transition mechanisms without any VPN configured 4to6 encountered a lot more latency at both packet sizes as opposed to 6to4. However, when VPNs were configured on both transition mechanisms, a key finding was 6to4 with IPsec did experience more latency then 4to6 with PPTP at packet size 128 bytes.

Fig. 14 illustrates and highlights the TCP DNS jitter for the Pure IPv4, 4to6, 4to6 (PPTP), Pure IPv6, 6to4 and 6to4

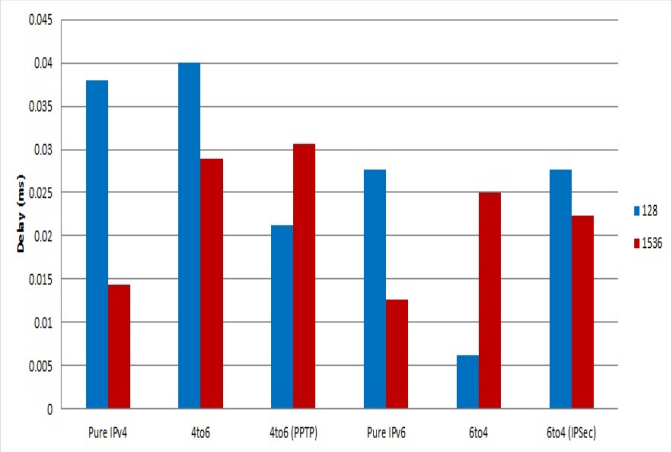

Fig. 13: DNS TCP Delay

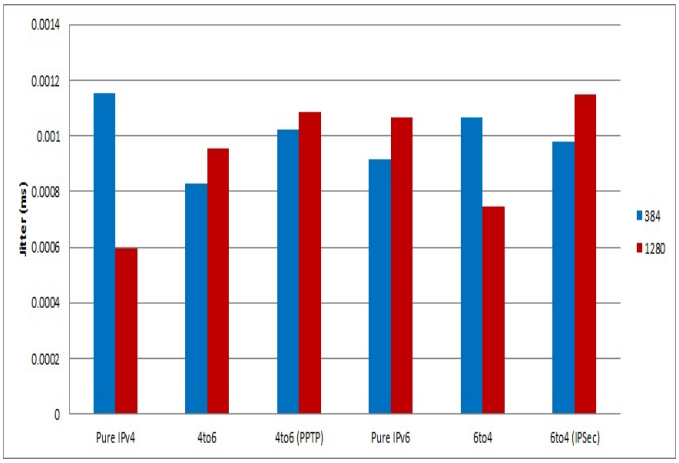

Fig. 14: DNS TCP Jitter

(IPsec). Pure IPv4 had the highest TCP DNS Jitter (1.15ms) at the third packet size, 384Bytes, compared to IPv6, which showed (0.915ms. At the packet size 1280, IPV4 displayed the lowest DNS jitter value (0.60ms) compared to Pure IPv6, which was 1.06ms. 4to6 with the point to point tunneling protocol showcased higher TCP DNS jitter compared to 4to6 without any VPN protocol. Finally 6to4 displayed higher TCP DNS Jitter at packet size 384, being roughly 0.085ms higher than 6to4 with IPsec. At packet size 1280, 6to4 IPsec had a much greater delay (1.15ms) as opposed to 6to4 (0.75ms).

Fig. 15 displays the results for the following: UDP DNS throughput for Pure IPv4, 4to6, 4to6 (PPTP), Pure IPv6, 6to4 and 6to4 (IPsec). Pure IPv4 showed a rapid increase at packet size 512 when compared to Pure IPv6 which displayed almost half the DNS throughput. 4to6 displayed lower DNS throughput when compared to 4to6 with PPTP VPN at packet size 384, but 4to6 displayed slightly higher DNS throughput at packet size 512. 6to4 in both packet sizes showcased better DNS throughput results compared to 6to4 configured with IPsec VPN. In summary, IPv4 displayed

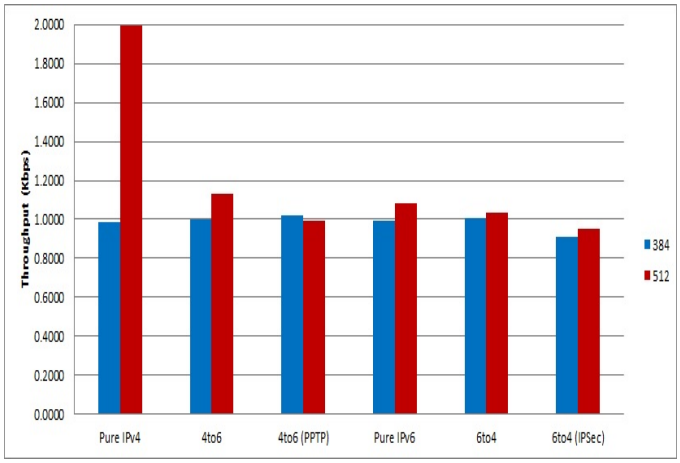

Fig. 15: DNS UDP Throughput

the overall highest DNS throughput, and 6to4 configured with IPsec VPN displayed the lowest overall DNS throughput.

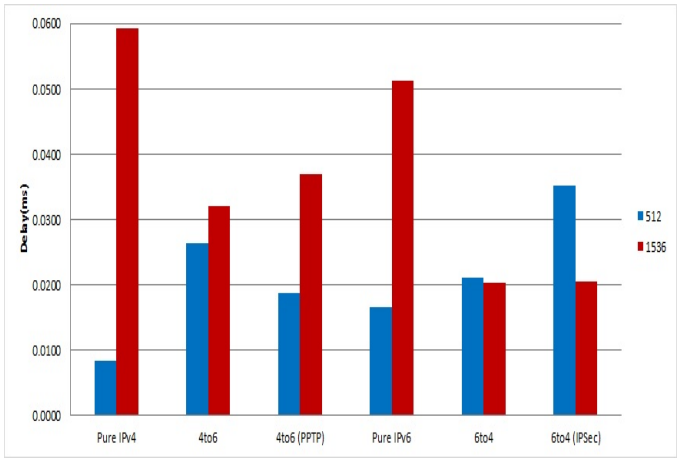

Fig. 16: DNS UDP Delay

Fig. 16, above, presents the overall DNS delay between Pure IPv4, 4to6, 4to6 PPTP, Pure IPv6, 6to4, and 6to4 IPsec. Pure IPv4 displayed the highest figure of 0.06ms delay, compared to IPv6s maximum delay of 0.051ms. 4to6 with PPTP had higher overall UDP DNS delay compared to 4to6 without any VPN. At packet size 512, 4to6 PPTP had a slightly higher UDP DNS delay of 0.02ms. 6to4 and 6to4 IPsec displayed similar UDP DNS delay at packet size 1536, but 6to4 with IPsec experienced larger delay at packet size 512, having an overall delay of 0.035ms.

Fig. 17 shows the overall UDP DNS jitter comparing Pure IPv4, 4to6, 4to6 PPTP, Pure IPv6, 6to4, and 6to4 IPsec. Pure IPv4 displayed the highest UDP DNS Jitter at packet size 1280, which was 0.00015ms. IPv6 showed constant UDP DNS jitter at both packet sizes. 4to6 had a similar value to 4to6 PPTP at the 1280 packet size, but had a value of 0.001 delay at packet size 512. 4to6 (PPTP) and Pure IPv6

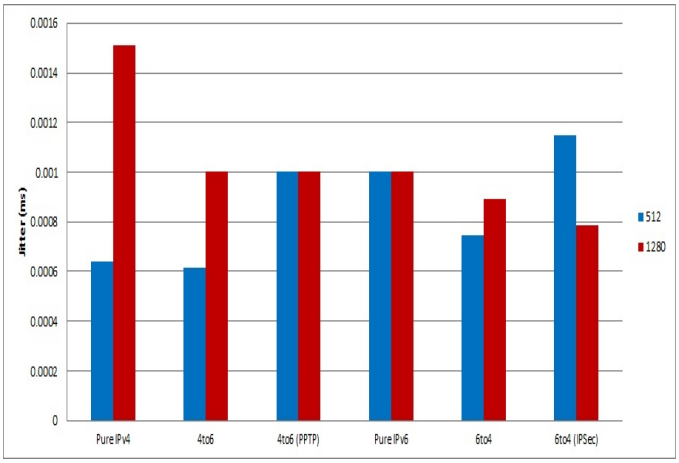

Fig. 17: DNS UDP Jitter

displayed the same jitter for packet sizes 512 and 1280. 6to4, at packet size 1280, had higher UDP DNS jitter when compared to 6to4 IPSec.

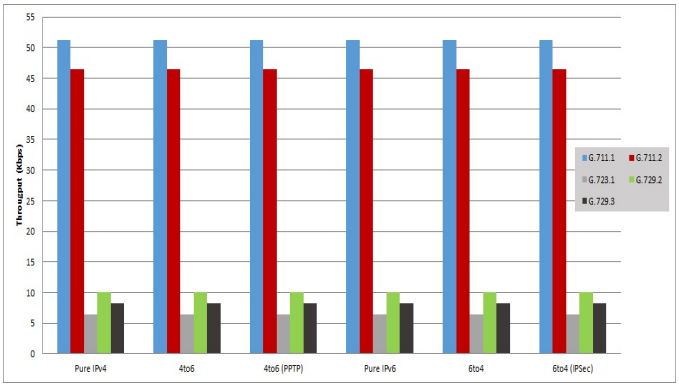

Fig. 18: VoIP Throughput for VoIP Codecs

In Fig. 18, CODEC G.723.1 and G.711.2 both represents the highest throughput overall. The throughput was consistent for the rest of the CODECs across all the networks.

V. Conclusion

The results obtained show that the 6to4 transition mechanism is reasonably faster and a more reliable protocol. Throughput for the 6to4 transition mechanism was par with IPv6/IPv4 for both TCP and UDP. The 6to4 transition mechanism had significantly lower delay when compared to the 4to6 transition mechanism. When 6to4 was configured with the IPsec VPN protocol throughput was drastically lower than 4to6 and 4to6 with PPTP VPN protocol. Transition mechanism 6to4 displayed very low delay, but when configured with IPsec delay increased to almost three times its original value. TCP/UDP performance results obtained for 4to6 transition mechanism with and without PPTP VPN were consistent when analysing throughput and jitter. 4to6 DNS throughput was higher without PPTP configured.6to4 with IPsec had the highest overall TCP DNS Throughput. 4to6 UDP DNS throughput was lower when compared to 4to6 configured with PPTP VPN, but at packet size 512 it displayed slightly higher throughput. 6to4 configured on its own was similar to 4to6, but once configured with IPsec it recorded the lowest UDP DNS throughput. As far as VoIP traffic goes, there is little to no variation in network performance.

References

[1] WIDE project. *SHISA, 2006. http://www.mobileip.jp/.*

[2] E. Y. Park, J. H. Lee, B. G. Choe,*An IPv4-to-IPv6 Dual Stack Transition Mechanism Supporting Transparent Connections between IPv6 Hosts and IPv4 Hosts in Integrated IPv6/IPv4 Network*, IEEE Conference Publications, 2004. 2, 1024-1027

[3] [3] Maula, A. (2010). *A Review and Qualitative Analysis of IPv6 and IPv4 Interoperability Technologies. Helsinki University of Technology, 1-5.*

[4] Bahaman, E. Hamid, and A. Prabuwono, *Network Performance evaluation of 6to4 tunneling,* in Innovation Management and Technology Research (ICIMTR), 2012 International Conference on, 2012, pp. 263-268.

[5] J.L. Chen, Y.C. Chang, C.H. Lin, *Performance Investigation of IPv4/IPv6 Transition Mechanisms*, IEEE Conference Publications, 2004. 2, 545-550. *doi: 10.1109/ICACT.2004.1292930*

[6] Rouse, M. (2007). *Virtual Private Network (VPN). Retrieved from ¡http://searchenterprisewan.techtarget.com/definition/virtual-private-network*

[7] B. Schneier, Mudge, *Cryptanalysis of Microsoft's Point-to-Point Tunneling Protocol (PPTP).* 1998, pp. 132-141, *doi: 10.1145/288090.288119.*

[8] Avallone, S.; Guadagno, S.; Emma, D.; Pescape, A.; Ventre, G. Ventre, *D-ITG Distributed Internet Traffic Generator, 2004 Quantitative Evaluation of Systems, 2004. QEST 2004.* Proceedings. First International Conference on the, 27-30 Sept 2004, pp 316-317. DOI: *10.1109/QEST.2004.1348045*

[9] Enguo, Z , Guoliang, W. (2009). *Network Security Protection Solutions of Electric Power Enterprise Based on VPN Technology.* 2009 International Conference on Computational Intelligence and Security. 402-405.

[10] S. Narayan, S. Kolahi, S. Sunarto, D. Nguyen,P. Mani, *"The Influence of Wireless 802.11 g LAN Encryption Methods on Throughput and Round Trip Time for Various Windows Operating Systems," In Proceeding of the IEEE Communication Networks and Services Research Conference*, pp. 171-175, April 2008.

[11] S. Narayan,P. Shang, N.Fan, *"Network performance evaluation of internet protocols ipv4 and ipv6 on operating systems," In Proceedings of the IEEE International Conference Wireless and Optical Communications Networks (WOCN'09)*, pp. 1-5, April 2009.

[12] S. Narayan, S. Tauch, *"IPv4-v6 configured tunnel and 6to4 transition mechanisms network performance evaluation on Linux operating systems," In Proceeding of the IEEE 2nd International Conference on In Signal Processing Systems (ICSPS)*, pp. V2-113, April 2010.

[13] S. Narayan, S. Kolahi, K.Brooking, S. de Vere, *"Performance evaluation of virtual private network protocols in Windows 2003 environment," In Proceedings of the IEEE International Conference on Advanced Computer Theory and Engineering (ICACTE'08)*, pp. 69-73, December 2008.

[14] S. Narayan, K. Brooking, S. de Vere, *"Network performance analysis of vpn protocols: An empirical comparison on different operating systems," In Proceedings of the IEEE International Conference on Networks Security, Wireless Communications and Trusted Computing (NSWCTC'09)*, pp. 645-648, 2009.

[15] Govil, 1. Govil, N. Kaur and H. Kaur, *"An examination of IPv4 and IPv6 networks: Constraints and various transition mechanisms", Proceedings of lEEE Sounteastcon*, pp. 178-185, April 2008.

[16] R. Gilligan, E. Nordmark, *Transition Mechanisms for IPv6 Hosts and Routers,* Request for Comments 1933, Internet Engineering Task Force, April 1996.