Effect of WPA2 Security on IEEE 802.11n Bandwith and Round Trip Time in Peer-Peer Wireless Local Area Networks

Peng Li, Samad S. Kolahi, Mustafa Safdari, and Mulugeta Argawe
Department of Computing, Unitec New Zealand
penglinz@gmail.com
skolahi@unitec.ac.nz

Abstract—**In this paper 802.11 wireless peer-peer network is evaluated for both IPv4 and IPv6 in Windows 7 and Fedora 12 operating systems. IPv4 has higher throughput than IPv6 for all packet sizes for both Windows 7 and Fedora 12 operating systems. Results further indicate that implementing WPA2 wireless security reduces bandwidth and increase delay in wireless networks.**

Keywords-WPA2, IPv4 and IPv6, TCP, 802.11n Peer-Peer Wirless LAN.

I. INTRODUCTION

In wireless networking, the IEEE 802.11n is the most recent wireless standard, and defines how to design interoperable Wireless Local Area Network (WLAN) equipment that provides a range of capabilities including effective data rates, quality of service, range optimization, reliability, network management and security. Due to the addition of the MIMO (Multiple-Input and Multiple-Output) technology, the 802.11n theoretically supports maximum data rate of up to 600 Mbps and maximum coverage area of up to 250 meters [1]. Therefore, it has been proven that the IEEE 802.11n wireless standard has increased bandwidth and wireless coverage area than its predecessors, the IEEE 802.11a/b/g.

With the growth of the Internet and its increasing globalization, the current Internet Protocol version 4 (IPv4) will run out of addresses in near future [2]. Internet Engineering Task Force (IETF) developed a new version of Internet Protocol, IPv6 that not only substantially expands the address space to 2^{128}, but also has a raft of additional benefits that were lacking in the predecessor, such as auto-configuration, more granular control of QoS (Quality of Service), better secure features, and compatible with latest 3G mobile technology. New versions of popular end-user operating systems have capability for IPv6, and therefore hardware vendors, software developers and Internet Service Providers (ISP) are moving towards offering support for IPv6 [3].

Wireless access is still in its infancy and not as secure as wired network. The main security protocol, Wi-Fi Protection Access, Version 2 (WPA2/802.11i) came with the purpose of solving several serious weaknesses in the Wired Equivalent Privacy (WEP) cryptography method. It is able to provide wireless access authentication due to its encryption algorithm, which provides key enabler for secure wireless networks, allowing for client and wireless network authentication [4].

In this paper, we produce new results for the impact of WPA2 security on the performance of IPv4 and IPv6 over Peer-to-Peer 802.11n wireless LAN. At the time of this research, Windows operating systems had approximately 90% market share [5] while Linux based operating systems are getting more and more popular. We established test-bed to compare the performance of wireless 802.11n with and without WPA2 security for IPv4 and IPv6 using TCP (Transmission Control Protocol) on Windows 7 and Fedora 12 operating systems.

The organization of this paper is as follows. In the next section the related research is discussed. Section III outlines the experimental setup used in this research. Section IV covers generating and traffic measurement tool. In Section V, we present results and discuss the findings. Section VI covers conclusion. Future work is described in Section VII followed by references.

II. RELATED RESEARCH

Previous researchers have studied performance comparison of IPv4 and IPv6 on different operating systems, and looked at some aspects of the influence of encryption on network performance.

In 2004, Zeadally et al. [6] compared performance of IPv4 and IPv6 protocols on different operating systems including Windows 2000, Solaris and Linux. Their results demonstrated that IPv4 and IPv6 on Linux outperformed Windows 2000 and Solaris 8 for all the metrics used. In addition, they found out there was a slightly degradation in throughput and round-trip latency performances for IPv6 compared to IPv4 on Windows 2000 and Solaris.

In 2007, Filho et al. [7] evaluated the impact of security mechanisms WEP and WPA on the performance of 802.11 g network. Their results showed when the security protocols including WEP 64, WEP 128 and WPA were used in the IEEE 802.11g wireless network, the demand time of processing traffic was increased and the throughput was decreased. In

addition, the UDP throughput dropped by 4% for WEP 64, 7% for WEP 128, and 5% for WPA on Windows XP when the security protocols were applied.

In 2008, S.S. Kolahi et al. [8] investigated the influence of wireless 802.11g LAN encryption methods on throughput and Round Trip Time (RTT) for Various Windows operating systems over Peer-to-Peer network. Their results showed that the performance in terms of throughput and response time suffered when encryption technique was implemented. Degree of degradation depended on the operating system and encryption method used. TCP traffic suffered a degradation of approximately 4-6% with WPA for different Windows operating systems.

In 2009, S.S. Kolahi et al. [9] conducted a study on the impact of overheads of security techniques for 802.11n on Windows XP, Windows Vista and Windows Server 2008. The main contribution of their paper was to investigate the impact of security on throughput and round trip time on those operating systems. Their results indicated that for XP enabling WPA2 results in an average of approximately 8 Mbps less throughput than open systems for IPv4 and 5 Mbps less throughput for IPv6. With WPA2 security enabled on Vista, the results showed that in average, there were approximately 11 Mbps less throughput than open system for IPv4, and 19 Mbps less throughput for IPv6. With WPA2 security, IPv4 provided higher bandwidth than IPv6.

To the authors' knowledge, there is no work done to date in literature. The novelty of this research is to investigate the impact of WPA2 security on the performance of IPv4 and IPv6 using TCP on Windows 7 and Fedora 12 over 802.11n Peer-to-Peer WLAN. The experimental setup used in this research is described next.

III. EXPERIEMNTAL SETUP

To measure performance of IPv4 and IPv6 on Windows 7 and Fedora 12, two client machines with identical hardware (CPU: Intel® Core™ 2 Duo 6300 1.87 GHz, RAM: 2.00 GB, Network card: Air Live Wn-5000 wireless PCI NIC, Hard drive: Western Digital Caviar 7200 [160 GB]) were connected wirelessly via Cisco Linksys WAP4410N 802.11n Access Point (AP). The distance between the access point and the workstations was well within two meters in-order to maintain the optimum signal strength.

The test-bed setup remained constant for all experiments conducted, and the test-bed diagram is displayed as figure 1:

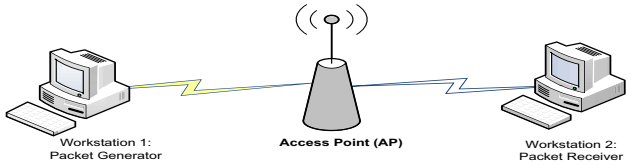

Figure 1: Network test-bed for Windows 7 and Fedora 12

The two different Operating Systems setup and configuration are explained as follows:

- In test-bed I, Microsoft Windows 7 Professional operating system is installed on both client machines. Because both IPv4 and IPv6 support in Windows 7, they can be enabled and configured on both computers simultaneously by using graphic interface.
- In test-bed II, Fedora 12 operating system is installed on both client machines. By default, Fedora 12 contains complete support for both IPv4 and IPv6, thus they can be enabled and configured on both computers simultaneously by using graphic interface or command lines.

According to Killelea [10], throughput (the number of bits transmitted per unit time) depends on several factors in a network, such as process limitations and hardware designs. In-order to eliminate the effect of such conditions, the hardware is benchmarked and a similar setup is used for all the tests to negate the effect of the processor limitations and hardware design. In addition, the connection between the access point and the client is wireless. The distance between the access point and the workstations is well within two meters in-order to maintain the optimum signal strength.

Parameters used for the access point configuration are:

(a) Channel bandwidth – In addition to the direction of the transmission, a channel is characterized by its bandwidth. In general, the greater the bandwidth of the assigned channels, the higher the possible speed of transmission. The access point provided two options here, 20 MHz and 40 MHz, and the latter was selected to utilize the full bandwidth.

(b) Guard Interval – Guard intervals are used to ensure that distinct transmissions do not interfere with one another. The purpose of the guard interval is to introduce immunity to propagation delays, echoes and reflections, to which digital data is normally very sensitive. This function was left appropriately to its default setting on the access point.

(c) CTS (Carpal Tunnel Syndrome) Protection Mode – This function boosts the access point's ability to detect all wireless connections but severely degrades performance, hence this setting was disabled to maximize performance.

(d) Beacon Interval – This function indicates the variable times in which clients meet the access point, this includes send and receive packets, and synchronism [7, 8]. This setting was best left at the default interval of 100ms.

(e) DTIM (Delivery Traffic Indication Message) Interval – This setting specifies how often the access point broadcasts a Delivery Traffic Indication Message. According to the manual of the specific Linksys access point used in this project, lower settings ensure efficient networking. The default setting of 1ms therefore was left for achieving the best results.

(f) RTS Threshold – RTS (Request-to-Send) is a signal sent from the transmitting station to the receiving station requesting permission to transmit data. This setting is used to decrease the problem of the hidden stations due to distance or signal blockage [11]. The manual for the Linksys access-point recommended that this be left at the default setting of 2347 for optimum performance.

(g) Fragmentation Threshold – This specifies the number of bytes used to fragment the frames with a purpose to increase

transfer reliability. If the frame size is very big, it can cause heavy interference and elevate the retransmissions rate. On the other hand, if the frame is too small, it will create overhead during the transmission and reduce the throughput rate [7, 8]. The parameter value for this was left at the default setting of 2346.

IV. Data Generation and Traffic Measurement Tool

Netperf [12] was selected as the primary tool for this particular test-bed to analyze the performance of IPv4 and IPv6 on the different operating systems over 802.11n WLAN. In addition, Netperf can be used to measure the performance of many different types of networking. It creates and sends TCP packets in either IPv4 or IPv6 networks, and provides tests for throughout and end-to-end latency. Netperf has also been used for similar researches such as on the impact of wireless LAN security on performance of different Windows operating systems [8].

The metric used in the experiment are throughput (measured in Mbps) and RTT (measured in ms). These metrics provide a valuable insight into network performance since they are the rate at which data get transmitted from one client side to another over a network. In addition, the maximum TCP window size (64KB) was used to ensure the optimum data transfer during the tests. The experimental results are presented and discussed next.

V. Experimental Results

Throughput and RTT results are measured for TCP protocol on an IEEE 802.11n network. Data packets are gradually increased in size (from 128 to 1408 Bytes) for TCP traffic and resulting throughput and RTT values are plotted. Streams of packets are generated and sent from one computer to the other included sending one million packets of a particular packet size and protocol (one run). For each packet size a total of 40 runs are carried out and the results are averaged and standard deviation of results are calculated.

The TCP throughput results on open system (OS) for the two operating systems are illustrated in Figure 2.

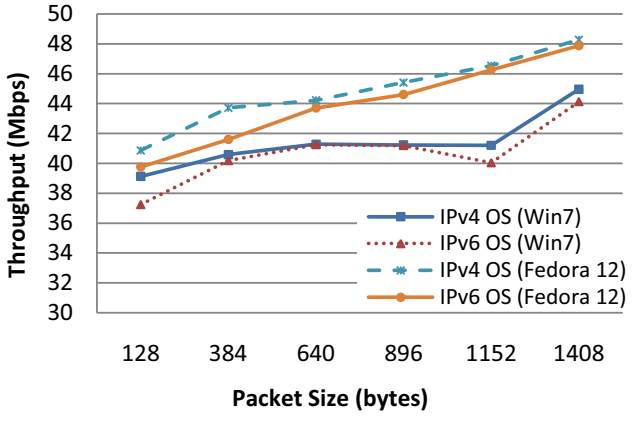

Figure 2: TCP Throughput of Windows 7 and Fedora 12 on Open System

Figure 2 shows on open system, IPv4 has higher TCP throughput results than IPv6 for all packet sizes for both Windows 7 and Fedora 12. On Windows 7, IPv4 TCP throughput results range from 39.12 to 44.96 Mbps and IPv6 TCP throughput values are from 37.24 to 44.13 Mbps. The maximum difference between IPv4 and IPv6 on open system is 1.88 Mbps on packet size 128 bytes. On Fedora 12, the TCP throughput TCP throughput t results of IPv4 range from 40.86 to 48.27 Mbps and IPv6 values are from 39.76 to 47.88 Mbps. The highest point of difference between IPv4 and IPv6 is 2.10 Mbps on packet size 384 bytes.

As can be seen from the figure 2, on open system, although Windows 7 and Fedora 12 results evidently show different TCP throughput values for IPv4 and IPv6, there are some similarities in their performance. IPv4 outperforms IPv6 for both operating systems, and throughput values increase as the size of packet increase for most packet sizes. In addition, on open system, the TCP throughput values are similar between the two operating systems for both IPv4 and IPv6, and the ranges are within +/-6 Mbps of each other. The maximum difference between Windows 7 and Fedora 12 are noticed at packet size 1152 bytes for both IPv4 and IPv6, where IPv4 on Windows 7 has 5.34 Mbps less throughput than IPv4 on Fedora 12, and IPv6 on Fedora 12 has 6.22 Mbps higher throughput than IPv6 on Windows 7.

Figure 3 shows TCP throughput results on WPA2 security enabled for the two operating systems with IPv4 and IPv6 protocols for different packet sizes.

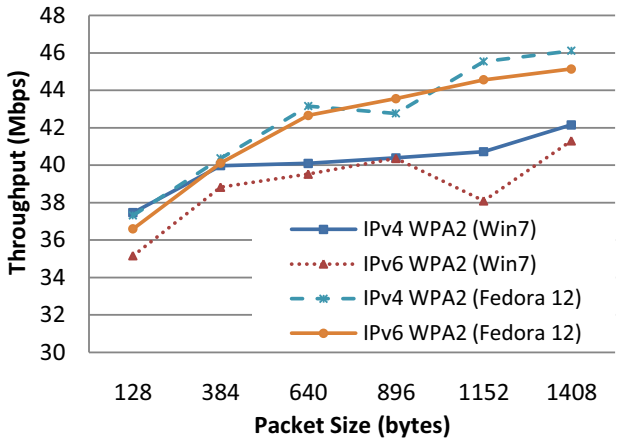

Figure 3: TCP Throughput of Windows 7 and Fedora 12 on WPA2 security enabled

On WPA2 security enabled, the TCP throughout results of IPv4 on Windows 7 range from 37.46 to 42.15 Mbps, and IPv6 values are from 35.16 to 41.30 Mbps. IPv4 outperforms IPv6 for all packet sizes. The highest point of difference between IPv4 and IPv6 on WPA2 security enabled is 2.63 Mbps on packet size 1152 bytes. For Fedora 12, IPv4 throughout results range from 37.31 to 46.11 Mbps, and IPv6 values are from 36.59 to 45.14 Mbps. The highest gap

between IPv4 and IPv6 is 0.98 Mbps on packet size 1152 bytes.

Comparing the two operating systems on WPA2 security enabled, as depicted in Figure 3, Fedora 12 has a higher throughput than Windows 7 for both IPv4 and IPv6. The maximum difference between Windows 7 and Fedora 12 are noticed at packet size 1152 bytes for both IPv4 and IPv6, where Fedora 12 has 4.82 Mbps higher throughput than Windows 7 for IPv4, and 6.47 Mbps higher throughput for IPv6.

As can be seen from the Figure 2 and Figure 3, analyzing the impact of security on the IEEE 802.11n network by comparing the performance of TCP throughput on IPv4 and IPv6 with WPA2 enabled to its performance in an open system environment, it can be concluded that enabling WPA2 can result in less throughput for both IPv4 and IPv6. On Windows 7, the highest point of difference between open system and WPA2 security enabled is on packet size 1408 bytes where IPv4 has 2.81 Mbps and IPv6 has 2.83 Mbps higher throughput on open system than WPA2 security enabled. For Fedora 12, the maximum difference is noticed at packet size 128 bytes for both IPv4 and IPv6 where IPv4 provides 3.55 Mbps and IPv6 provides 3.17 Mbps higher throughput in the open environment.

The lower throughput results obtain on WPA2 security enabled is due to the overhead header (16 bytes), which add extra data to the packets [13]. The impact of security seems to have an equal effect on both IPv4 and IPv6.

The standard deviation for the above throughput results are recorded in the following table:

TABLE 1
STANDARD DEVIATION FOR THROUGHPUT

Packet size (Bytes)	Windows 7				Fedora 12			
	Open System		WPA2		Open System		WPA2	
	IPv4	IPv6	IPv4	IPv6	IPv4	IPv6	IPv4	IPv6
128	0.95	0.83	0.99	0.89	0.97	0.85	0.98	0.99
384	1.00	1.13	0.97	0.88	0.65	0.93	0.56	0.97
640	0.99	1.07	0.94	0.90	0.96	0.64	0.98	0.86
896	0.75	1.05	0.99	0.85	0.96	0.97	0.84	0.89
1152	0.86	0.96	0.96	0.90	0.98	0.58	0.97	0.83
1408	0.97	0.99	0.95	0.58	0.98	0.83	0.98	0.81

Figure 4 shows TCP Round Trip Time for IPv4 and IPv6 on the Windows 7 and Fedora 12 operating systems with no security enabled. The TCP RTT results show a gain in delay for both IPv4 and IPv6 with the increase in each packet size.

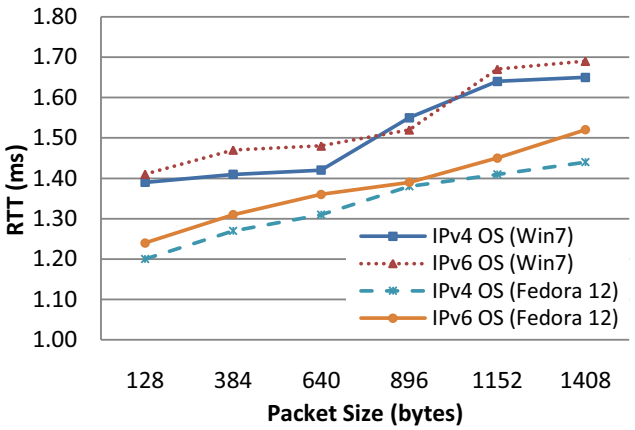

Figure 4: TCP RTT of Windows 7 and Fedora 12 on Open System

As can be seen from the figure 4, on Windows 7, IPv4 TCP RTT range from 1.40 to 1.48 ms, and IPv6 TCP RTT rates are from 1.44 to 1.83 ms. The maximum difference between IPv4 and IPv6 on open system is 0.11 ms on packet size 896 bytes. On open system, Fedora 12 RTT results exhibit that IPv4 RTT range from 1.20 to 1.44 ms, and IPv6 RTT values are from 1.24 to 1.52 Mbps. The highest point of difference between IPv4 and IPv6 is 0.08 Mbps on packet size 1408 bytes.

Comparing the operating systems, as depicted in Figure 4, the RTT for TCP is lowest on Fedora 12 for IPv4. In contrast, the highest TCP RTT is noticed on Windows 7 for IPv6. In addition, Windows 7 has comparatively more delay than Fedora 12 for both IPv4 and IPv6. With no security, the highest gap between Windows 7 and Fedora 12 is noticed at packet size 1152 bytes for both IPv4 and IPv6, where IPv4 on Windows 7 has 0.23 ms more latency than IPv4 on Fedora 12, and IPv6 on Windows 7 had also 0.23 ms more latency than IPv6 on Fedora 12.

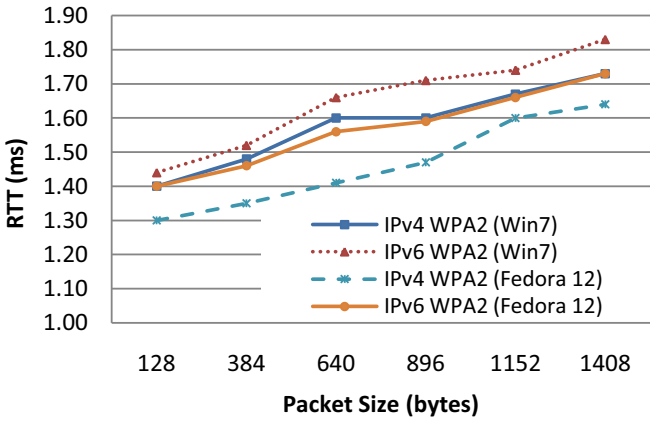

Figure 5: TCP RTT of Windows 7 and Fedora 12 on WPA2 security enabled

Figure 5 shows TCP RTT results on WPA2 security enabled for the two operating systems with IPv4 and IPv6 protocols for different packet sizes.

On WPA2 security enabled, the TCP RTT results of IPv4 on Windows 7 range from 1.20 to 1.44 ms, and IPv6 TCP

RTT is from 1.31 to 1.52 ms. the maximum difference between IPv4 and IPv6 on WPA2 security enabled is 0.08 ms on packet size 1408 bytes. For Fedora 12, IPv4 TCP RTT range from 1.30 to 1.64 ms, and IPv6 TCP RTT is from 1.40 to 1.73 ms. The highest gap between IPv4 and IPv6 is 0.15 ms on packet size 640 bytes.

On comparing the Windows 7 and Fedora 12's RTT results on WPA2 security enabled, it is clear that Fedora 12 outperforms Windows 7 for both IPv4 and IPv6. The highest points of difference between Windows 7 and Fedora 12 are noticed at packet sizes 640 bytes for IPv4 and 896 bytes for IPv6, where IPv4 on Windows 7 has 0.19 ms higher delay than IPv4 on Fedora 12, and IPv6 on Windows 7 has 0.12 ms higher delay than IPv6 on Fedora 12.

As can be seen from the Figure 4 and Figure 5, analyzing the impact of security on the IEEE 802.11n network by comparing the TCP RTT on IPv4 and IPv6 with WPA2 enabled to its performance in an open system environment, it can be concluded that enabling WPA2 can result in more delay rates for both IPv4 and IPv6. The graphs (Figure 4 and Figure 5) above show that on Windows 7, open system has less delay than WPA2 security enabled for all packet size on both IPv4 and IPv6. The highest points of difference between open system and WPA2 security enabled are on packet size 640 bytes for IPv4 and 896 bytes for IPv6, where IPv4 with no security has 0.18 ms and IPv6 with no security has0.19 ms less delay than on WPA2 security enabled. For Fedora 12, the maximum difference between open system and WPA2 security enabled is noticed at packet size 1408 bytes for IPv4 and 1152 and 1408 bytes for IPv6, where IPv4 with no security has 0.20 ms lower latency than WPA2, and IPv6 with no security has 0.21 ms lower latency than WPA2.

The standard deviation for the above RTT results are recorded in the following table:

TABLE 2
STANDARD DEVIATION FOR RTT

Packet size (Bytes)	Windows 7				Fedora 12			
	Open System		WPA2		Open System		WPA2	
	IPv4	IPv6	IPv4	IPv6	IPv4	IPv6	IPv4	IPv6
128	0.09	0.03	0.04	0.02	0.02	0.03	0.04	0.03
384	0.05	0.03	0.03	0.02	0.06	0.02	0.04	0.03
640	0.02	0.03	0.02	0.04	0.05	0.03	0.04	0.04
896	0.03	0.03	0.03	0.06	0.04	0.02	0.03	0.03
1152	0.03	0.04	0.04	0.05	0.03	0.02	0.04	0.04
1408	0.02	0.02	0.05	0.05	0.02	0.02	0.04	0.03

The gain in delay with the increase in each packet size is likely due to the amortization of overheads associated with larger packet sizes (larger user payloads) thus higher transmission time [6]. The WPA2 security has an equal impact on the two operating systems. The lower RTT obtained on WPA2 security enabled is due to the overhead header (16 bytes), which add extra data to the packets [13].

VI. CONCLUSION

The security overhead of WPA2 increased the amount of data sent, and negatively impact on the performance of overall TCP throughput and Round Trip Time over IEEE 802.11n wireless LAN. It can be concluded that enabling WPA2 results in approximately 2.8 Mbps less TCP throughput and 0.18 ms more delay than open system for both IPV4 and IPv6 on Windows 7. Same applies for the impact of WPA2 when Fedora 12 is used as an operating system, enabling WPA2 causes 3 Mbps less TCP throughput and 0.20 ms more TCP RTT than open system for both IPV4 and IPv6 on Fedora 12. The results indicate that Peer-Peer wireless LANs can provide upto 48Mbps as both of the links to the access points are wireless, this is much lass than what we achieved (180Mbps) in IEEE 802.11n wireless client-server networks experiments when one of the links was cable.

VII. FUTURE WORK

In future, we plan to extent this study by incorporating more operating systems and the ranges of metrics. In addition, the performance comparison of Windows and Linux Systems with IPv4 and IPv6 using both open systems and WPA2 security on 64-bit operating system will be investigated.

ACKNOWLEDGMENT

The authors would like to thank UNITEC Institute of Technology for funding the research team and providing the inventory needed.

REFERENCES

[1] IEEE P802.11n/D1.0 Draft Amendment to STANDARD[FOR] Information Technology-Telecommunications and information exchange between systems-Local and Metropolitan networks-Specific requirements-Part 11: Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) specifications: Enhancements Mar. 2006.

[2] OECD report, Internet Address Space: Economic Considerations in the Management of IPv4 and in the Deployment of IPv6, OECD Ministerial Meeting, Seoul, Korea, 17-18 June 2008.

[3] N. Baghaei and R. Hunt, "IEEE 802.11 Wireless LAN Security Performance Using Multiple Clients," in Proceedings of the 12th IEEE International Conference on *Networks*, Vol. 1, pp. 299-303, Nov. 2004.

[4] A. H. D. Lashkari, M.M.S. , Samadi, B., "A survey on wireless security protocols (WEP, WPA and WPA2/802.11i)," *Computer Science and Information Technology, 2nd IEEE International Conference on Beijing, ICCSIT 2009*, pp.48-52, 2009.

[5] NetApplications. *Operating System Market Share for May 2010*, June 2010. Available: http://marketshare.hitslink.com/operating-system-market-share.aspx?qprid=8

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

[7] E.J.M.A. Filho, P.N.L. Fonseca, M.J.S. Leitao, and P.S.F. de Barros, "Security versus Bandwidth: The Support of Mechanisms WEP e WPA in 802.11g Network," IFIP International Conference on Wireless and Optical Communications Networks, WOCN '07, pp. 1-5, 2007.

[8] S.S Kolahi, S. Narayan, D.D.T, Y. Sunarto, P. Mani, "The Impact of Wireless LAN Security on Performance of Different Windows

Operating Systems," *IEEE Symposium on Computers and Communications*, pp. 260-264, 2008.

[9] S. S. Kolahi, Z. Qu, B. K. Soorty, and N. Chand,"The Impact of Security on the Performance of IPv4 and IPv6 Using 802.11n Wireless LAN," *New Technologies, Mobility and Security (NTMS), 2009 3rd International Conference on*, pp. 1-4, 2009.

[10] P. Killelea, "Web Performance Tuning," http://www.amazon.ca/Web-Performance-Tuning-Patrick-Killelea/dp/product b/059600172X.

[11] D. Akin, and J. Geier, "802.11 PHY layers," CWAP - certified wireless analysis professional official study guide, Mc.Graw-Hill, pp. 353-355, 2004.

[12] R. Jones. Netperf 2.4.5. Available: http://www.netperf.org/netperf/NetperfNew.html

[13] A. D. Potorac and D. Balan. "The Impact of Security Overheads on 802.11 WLAN Throughput", *Computer Science and Control Systems*, vol. 2, pp. 47-52, 2009.