Performance Evaluation of Remote Access VPN Protocols on Wireless Networks

Ahmed A. Jaha
College of Industrial Technology
Misurata, Libya
Email: goha_99 [AT] yahoo.com

Abstract— **VPN solutions can be deployed on a wireless network infrastructure to secure transmission between wireless clients and their wired enterprise network. There are many software platforms that can be used to implement software-based VPN solution such as windows, Linux, Solaris, Mac, and BSD. In this paper, the performance evaluation of some remote access VPN solutions, namely Point to Point Tunneling Protocol (PPTP), Layer 2 Tunneling Protocol over Internet Protocol Security (L2TP/IPSec), and Secure Socket Layer (SSL) will be empirically investigated on wireless networks. Some of QoS performance metrics like throughput, latency, jitter, and packet loss are measured to explore the impact of these VPNs on the ultimate performance perceived by end user applications. All experiments were conducted using wireless VPN client (vpn01Client) connected to domain controller server (dc01Server) through VPN server (vpn01Server).**

Keywords t— ***WLAN; VPN; PPTP; L2TP; IPSec; OpenVPN.***

I. INTRODUCTION

Over the past several years, wireless technology has enhanced the computer networking. The Wireless Local Area Network (WLAN) technology represented a convenient alternative to conventional wired LANs due to everywhere network access without wires, growing data rates, improving quality of service, and decreasing the prices [1]. Vulnerability of the wireless medium to some security threats increasing the priority of security issues. A Virtual Private Network (VPN) is a simple solution to achieve secure communications over the use of a public network infrastructure such as the Internet, maintaining privacy through the use of a tunneling protocol and security procedures. It is also possible to similarly deploy VPNs on a wireless network infrastructure to secure transmission between wireless clients and their wired enterprise network. This method has been warmly accepted by the academia and industry as an alternative to securing WLANs. It involves in the creation of a VPN tunnel through the use of a tunneling protocol that encrypts traffic over the WLAN [2]. Although VPN servers are usually hardware-based devices, there are many software platforms that can be used to implement software-based VPN servers such as windows, Linux, BSD, and Solaris but the main two are Windows and Linux platforms.

II. REMOTE ACCESS VPN PROTOCOLS

For a tunnel to be established, both the tunnel client and the tunnel server must be using the same tunnelling protocol. Tunnelling technology can be based on either Layer 2, Layer 3, or Layer 5 which are corresponding to the Open Systems Interconnection (OSI) Reference Model [3].

A. Point to Point Tunnelling Protocol (PPTP)

PPTP is a standard Layer 2 tunnelling protocol developed by PPTP Forum which consists of Microsoft and some other remote access vendors. Basically, PPTP is an expansion of Point to Point Protocol (PPP), which encapsulates PPP frames in IP datagrams for transmission over an IP-based network, such as the Internet. PPTP is described in RFC 2637 in the IETF RFC Database [4]. Microsoft has included PPTP clients in all versions of Windows since Windows 95 and PPTP servers in all its server products since Windows NT 4.0. In addition, PPTP clients and servers are supported in Linux.

B. Layer Two Tunnelling Protocol (L2TP)

L2TP is a combination of PPTP and Layer Two Forwarding (L2F). Rather than having two incompatible tunnelling protocols that are competing each other and puzzling customers, the IETF recommended that the two technologies be combined into a single tunnelling protocol that represents the best features of both. L2TP is described in RFC 2661 in the IETF RFC Database [5].

C. Internet Protocol Security (IPSec)

IPSec is a framework of IETF open standards intend at securing traffic on the network layer. It does not identify the authentication and encryption protocol to use. This makes it flexible and able to support new authentication and encryption methods as they are developed. IPSec is described in RFCs 2401-2411 and 2451 in the IETF RFC Database [6]. IPSec is a standard for encrypting and authenticating IP packets at the network layer. IPSec has a set of cryptographic protocols for securing network packets and exchanging encryption keys. L2TP/IPSec combines L2TP's tunnel with IPSec's secure channel. Microsoft has provided L2TP/IPSec VPN clients in all products since Windows XP and L2TP/IPSec servers in all server versions since Windows 2000. Also, there are several open-source implementations of L2TP/IPSec for Linux [7].

D. Secure Socket Layer (SSL)

SSL is a higher-layer security protocol developed by Netscape. SSL is usually used with Hypertext Transaction Protocol HTTP to allow secure Web browsing, called HTTPS.

Most browsers and servers currently use SSL 3.0 to provide confidentiality, integrity, and authenticity between web-client and web-server [8]. However, SSL can also be used to construct a VPN tunnel. For example, OpenVPN is an open-source VPN package for Linux, BSD, Mac, and Windows, which uses SSL to grant encryption of both the data and control channels.

III. EXPERIMENTAL TESTBEDS

The work in this paper is based on three testbeds were built in the network lab at the College of Industrial Technology to evaluate the performance of some remote access VPNs on wireless networks. The first testbed has been designed to evaluate the performance of some remote access VPN solutions on windows server 2003 VPN server. Hardware and software components of this testbed are listed in Tables I and II, respectively. The testbed setup is shown in Fig. 1.The second testbed has been designed to evaluate the performance of some remote access VPN solutions on fedora core 6 VPN server. Hardware and software components of this testbed are listed in Tables I and III, respectively. The testbed setup is shown in Fig. 1. The third testbed has been designed to evaluate the performance of some remote access VPN solutions on e-Live IP-8000VPN Broadband VPN Router. Hardware and software components of this testbed are listed in Tables IV and V, respectively. The testbed setup is shown in Fig. 2.

TABLE I. W2K3 AND FC6 TESTBEDS HARDWARE COMPONENTS

Node	Description
dc01Server	Desktop equipped with double Genuine Intel 2600 MHz processor, 512 Mbytes of RAM, and VIA Rhine II Compatible Fast Ethernet Adapter built-in network interface card. It is act as a domain controller server.
vpn01Server	Desktop equipped with double Genuine Intel 3000 MHz processor, 512 Mbytes of RAM, Broadcom Extreme Gigabit Ethernet built-in network interface card, and VIA VT6105 Rhine III Compatible Fast Ethernet Adapter network interface card. It is act as a VPN server.
vpn01Client	Laptop equipped with Genuine Intel 1866 MHz processor, 512 Mbytes of RAM, and Intel(R) PRO/ Wireless 2200BG network connection. It is act as a VPN client.
Access Point	LINKSYS, wireless-G, Access Point with SES model WAP54G.
Switch	D-link, 10/100 Fast Ethernet Switch

TABLE II. W2K3 TESTBED SOFTWARE COMPONENTS

Node	Description
dc01Server	This node is loaded with windows server 2003. Routing and remote access server setup wizard is used to configure this node to act as a domain controller server [9].
vpn01Server	This node is loaded with windows server 2003. Routing and remote access server setup wizard is used to configure this node to act as PPTP and L2TP/IPSec VPN servers [10] and OpenVPN-2.0.9.exe is installed to configure this node to act as SSL VPN server [11].
vpn01Client	This node is loaded with windows XP SP/2. New connection wizard is used to configure this node to act as PPTP VPN client that is connected to vpn01Server node with MS-CHAPv2 authentication algorithm, MPPE encryption algorithm, and no compression algorithm [10]. New connection wizard is used to configure this node to act as L2TP/IPSec VPN client that is connected to vpn01Server node with preshared key, MS-CHAPv2 authentication algorithm, ESP-3DES encryption algorithm, and no compression algorithm [10]. OpenVPN-2.0.9.exe is installed to configure this node to act as SSL client that is connected to vpn01Server node with preshared key, SHA1 authentication algorithm, 3DES encryption algorithm, and no compression algorithm [11].

TABLE III. FC6 TESTBED SOFTWARE COMPONENTS

Node	Description
dc01Server	Same as in the table II.
vpn01Server	This node is loaded with fedora core 6 [12]. Pptpd-1.3.3-1.fc6.i386.rpm is installed to configure this node to act as PPTP VPN server [13], xl2tpd-1.1.09-1.i386.fc6.rpm and OpensWan-2.4.5-2.1 are installed to configure this node to act as L2TP/IPSec VPN server [14], and OpenVPN-2.0.9.tar is installed to configure this node to act as SSL VPN server [15].
vpn01Client	Same as in the table II.

TABLE IV. E-LIVE TESTBED HARDWARE COMPONENTS

Node	Description
dc01Server	Same as in the table I.
vpn01Server	OvisLink, e-Live IP-8000VPN Broadband VPN Router.
vpn01Client	Same as in the table I.
Access Point	Same as in the table I.
Switch	Same as in the table I.

TABLE V. E-LIVE TESTBED SOFTWARE COMPONENTS

Node	Description
dc01Server	Same as in the table II.
vpn01Server	Web based configuration scheme is used to configure this node to act as PPTP and L2TP/IPSec VPN servers.
vpn01Client	Same as in the table II.

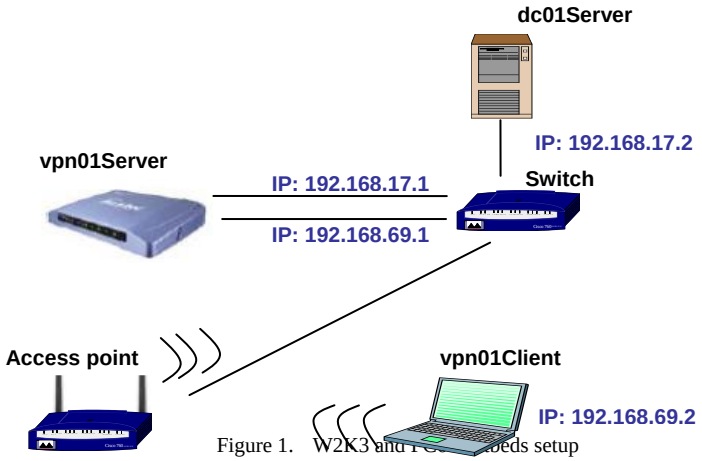

Figure 1. W2K3 and [illegible] beds setup

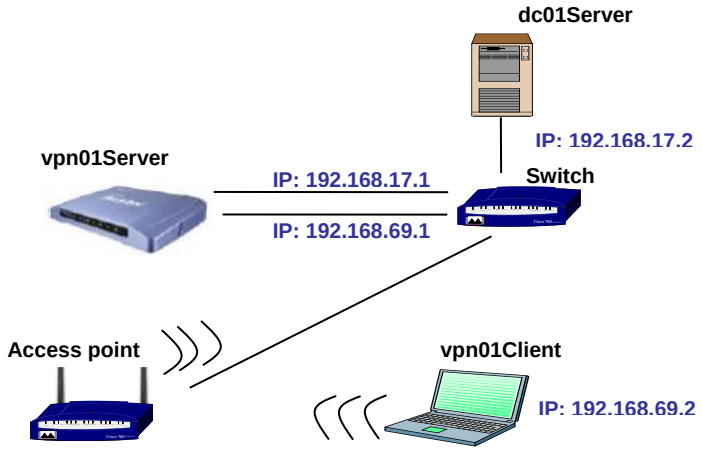

Figure 2. e-Live Testbed setup

During the experiments, the following parameters were used to quantify the QoS services provided [16]:

- Throughput is a measure of the amount of data that can be sent from end to end in a given amount of time. Throughput values are usually expressed in bits per second or packets per second. Throughput is affected by overhead and latency. While overhead decreases the amount of useful bytes transferred, latency affects the bandwidth-delay product for a TCP connection. Throughput is usually used as an estimate of the bandwidth of a network.
- Round Trip Time (RTT) is the amount of time it takes one packet to pass from one host to another and back to the originating host.
- Packet delay variation (Jitter) is measured for packets belonging to the same packet flow and illustrates the dissimilarity in the one-way delay that packets experience in the network. Jitter is efficiently a variation of packet delay where delays actually impact the quality of service.
- Packet loss is measured as the portion of packets transmitted but not received in the destination compared to the total number or packets transmitted. Packet loss is caused by line properties (Layer 1), full buffers (Layer 3) or late arrivals (at the application).

IV. EXPERIMENTAL TESTBEDS

Iperf tool has been used to measure both TCP throughput in TCP mode and UDP throughput, packet delay variation (jitter), and packet loss in UDP mode [17]. Hrping tool has been also used to measure Round Trip Time (RTT) [18].

The following results were collected from the above mentioned testbeds. The same experiments were repeated 10 times to find the average values.

TCP throughput is measured according to TCP window size and number of flows. The results of these experiments are illustrated in Fig. 3 and Fig. 4. These results indicate clearly that the PPTP on windows server 2003 has produced the best TCP throughput value (83.33 % of no VPN value), the PPTP on fedora core 6 has come on the second TCP throughput value (78.63 % of no VPN value), the OpenVPN on fedora core 6 has come on the third TCP throughput value (70.51 % of no VPN value), the L2TP/IPSec on windows server 2003 has come on the forth TCP throughput value (68.38 % of no VPN value), the L2TP/IPSec on fedora core 6 has come on the fifth TCP throughput value (65.38 % of no VPN value), the PPTP on e-Live IP-8000VPN Router has come on the sixth TCP throughput value (63.68 % of no VPN value), the L2TP on e-Live IP-8000VPN Router has come on the seventh TCP throughput value (59.83 % of no VPN value), and the OpenVPN on windows server 2003 has produced the lowest TCP throughput value (53.85 % of no VPN value).

RTT can be measured by sending packets with a variable packet size from a client to the server. The results of these experiments are illustrated in Fig. 5. These results illustrate clearly that the PPTP on windows server 2003 has produced the lowest RTT value (1.33 multiple of no VPN value), the PPTP on fedora core 6 has come on the second RTT value (1.45 multiple of no VPN value), the L2TP/IPSec on windows server 2003 has come on the third RTT value (1.50 multiple of

no VPN value), the OpenVPN on windows server 2003 has come on the forth RTT value (1.60 multiple of no VPN value), the OpenVPN on fedora core 6 has come on the fifth RTT value (1.65 multiple of no VPN value), the L2TP/IPSec on fedora core 6 has come on the sixth RTT value (1.73 multiple of no VPN value), the PPTP on e-Live IP-8000VPN Router has come on the seventh RTT value (1.82 multiple of no VPN value), and the L2TP on e-Live IP-8000VPN Router has produced the highest RTT value (1.92 multiple of no VPN value).

UDP throughput is measured according to transmission rate of packets. The results of these experiments are illustrated in Fig. 6. These results indicate clearly that the depression of the UDP throughput values of the PPTP on both windows server 2003 (65.68 % of no VPN value) and fedora core 6 (60.90 % of no VPN value) have been started when the transmission rate is exceeding beyond 10 Mbits/sec, the depression of the UDP throughput values of the L2TP/IPSec on both windows server 2003 (59.98 % of no VPN value) and fedora core 6 (57.09 % of no VPN value) have been started when the transmission rate is exceeding beyond 8 Mbits/sec, the depression of the UDP throughput values of both PPTP (24.89 % of no VPN value) and L2TP (23.59 % of no VPN value) on e-Live IP-8000VPN Router have been started when the transmission rate is exceeding beyond 6 Mbits/sec, and the depression of the UDP throughput values of the OpenVPN on both windows server 2003 (8.44 % of no VPN value) and fedora core 6 (7.69 % of no VPN value) have been started when the transmission rate is exceeding beyond 200 kbits/sec.

Jitter is measured according to the transmission rate of packets for UDP traffic. The results of these experiments are illustrated in Fig. 7. These results indicate clearly that the PPTP on windows server 2003 (1.64 multiple of no VPN value), the PPTP on fedora core 6 (1.70 multiple of no VPN value), the L2TP/IPSec on windows server 2003 (2.20 multiple of no VPN value), the L2TP/IPSec on fedora core 6 (2.30 multiple of no VPN value), the PPTP on e-Live IP-8000VPN Router (4.39 multiple of no VPN value), and the L2TP on e-Live IP-8000VPN Router (4.84 multiple of no VPN value) have produced a low Jitter values. Also, Fig. 7 indicates clearly that the OpenVPN on windows server 2003 (44.76 multiple of no VPN value) and the OpenVPN on fedora core 6 (44.91 multiple of no VPN value) have produced a higher Jitter values if the transmission rate is more than 200 kbits/sec.

Packet loss is measured according to the transmission rate of packets. The results of these experiments are illustrated in Fig. 8. These results illustrate clearly that the PPTP on windows server 2003 (1.43 multiple of no VPN value), the PPTP on fedora core 6 (1.49 multiple of no VPN value), the L2TP/IPSec on windows server 2003 (1.51 multiple of no VPN value), the L2TP/IPSec on fedora core 6 (1.57 multiple of no VPN value), the PPTP on e-Live IP-8000VPN Router (2.08 multiple of no VPN value), and the L2TP on e-Live IP-8000VPN Router (2.14 multiple of no VPN value) have produced a low Packet loss values. Also, Fig. 8 indicates clearly that the OpenVPN on both fedora core 6 (5.02 multiple of no VPN value) and windows server 2003 (5.07 multiple of no VPN value) have produced a higher Packet loss values if the transmission rate is more than 200 kbits/sec.

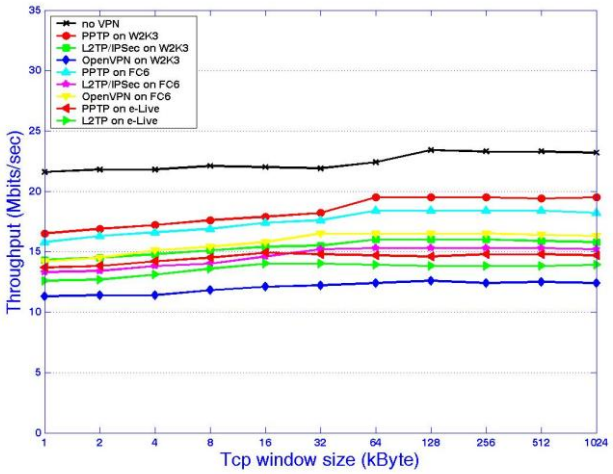

Figure 3. TCP throughput according to the window size

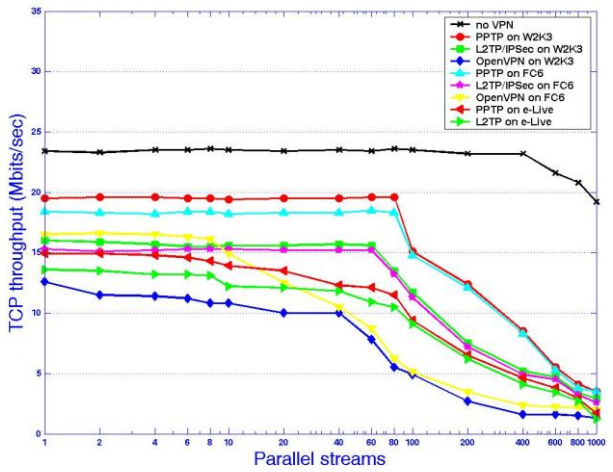

Figure 4. TCP throughput according to the parallel streams

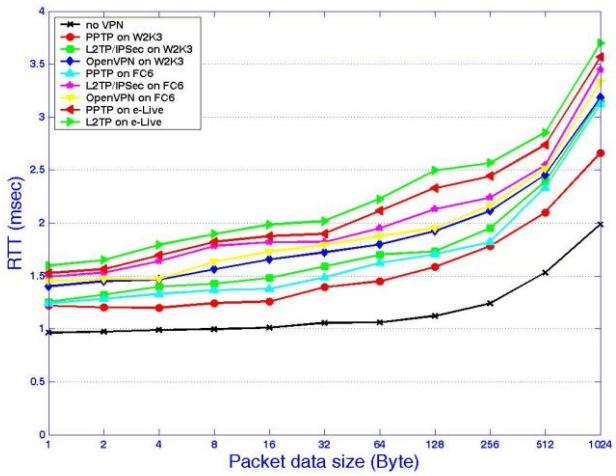

Figure 5. RTT according to the packet data size.

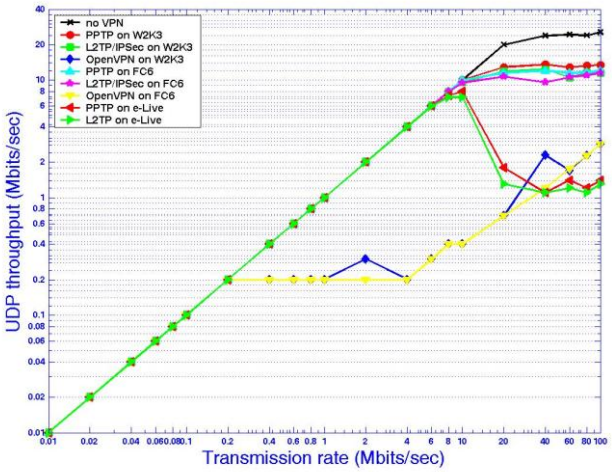

Figure 6. UDP throughput according to the transmission rate

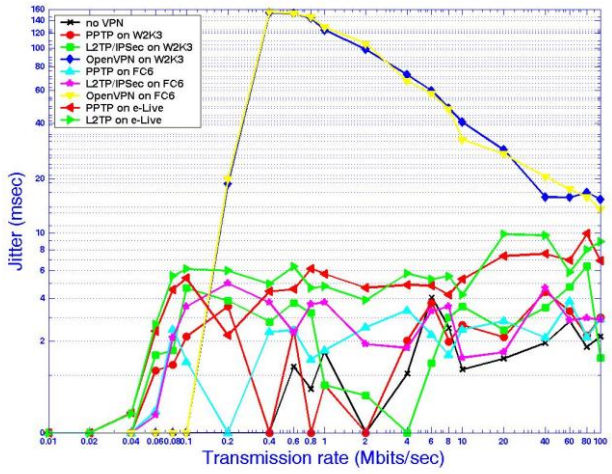

Figure 7. Jitter according to the transmission rate

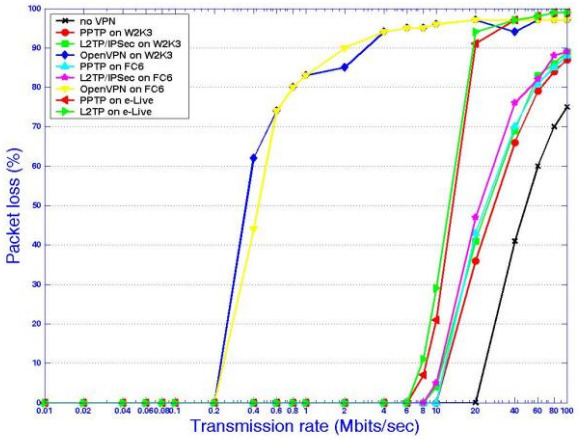

Figure 8. Packet loss according to the transmission rate.

To determine the grade of the testbeds experimental results that are summarized in Table VI, the following TCP and UDP throughput grades are suggested: excellent values are included in the interval [85% , 100%] of no VPN values, very good values are included in the interval [75% , 85%) of no VPN values, good values are included in the interval [65% , 75%) of no VPN values, acceptable values are included in the interval [40% , 65%) of no VPN values, weak values are included in the interval [25% , 40%) of no VPN values, and very weak values are included in the interval [0% , 25%) of no VPN values. As a result of inversing the previous intervals, the following RTT, jitter, and packet loss grades are suggested: excellent values are included in the interval [1 , 1.176] multiple of no VPN values, very good values are included in the interval (1.176 , 1.333] multiple of no VPN values, good values are included in the interval (1.333 , 1.538] multiple of no VPN values, acceptable values are included in the interval (1.538 , 2.5] multiple of no VPN values, weak values are included in the interval (2.5 , 4] multiple of no VPN values, and very weak values are included in the interval (4, ∞) multiple of no VPN values.

The following background colors are used: dark green (excellent), light green (very good), dark yellow (good), light yellow (accepted), light red (weak), and dark red (very weak). Therefore, Table V indicates clearly that PPTP on both W2K3 and FC6 has produced a very good TCP throughput values, L2TP/IPSec on both W2K3 and FC6 and OpenVPN on FC6 have produced a good TCP throughput values, and PPTP on e-Live, L2TP on e-Live, and OpenVPN on W2K3 have produced an acceptable TCP throughput values.

TABLE VI. SUMMARY OF TESTBEDS RESULTS

Metrics / Testbed	TCP throughput values in % of no VPN	RTT values in multiple of no VPN	UDP throughput values in % of no VPN	Jitter values in multiple of no VPN	Packet loss values in multiple of no VPN

No VPN	100.00 %	1.00	100.00 %	1.00	1.00
PPTP on W2K3	83.33 %	1.33	65.68 %	1.64	1.43
PPTP on FC6	78.63 %	1.45	60.90 %	1.70	1.49
PPTP on e-Live	63.68%	1.82	24.89 %	4.39	2.08
L2TP/IPSec on W2K3	68.38 %	1.50	59.98 %	2.20	1.51
L2TP/IPSec on FC6	65.38 %	1.73	57.09 %	2.30	1.57
L2TP on e-Live	59.83 %	1.92	23.59 %	4.84	2.14
OpenVPN on W2K3	53.85%	1.60	8.44 %	44.76	5.02
OpenVPN on FC6	70.51%	1.65	7.69 %	44.91	5.07

V. CONCLUSION AND FUTURE WORK

From the results that were collected from the testbeds and the user applications requirements, the following conclusion remarks are gained:

- Due to the smallest overhead packets that have been introduced by PPTP with respect to other tested protocols, PPTP on both windows server 2003 and fedora core 6 have produced the best performance values.
- In order to have strong security, L2TP/IPSec combines L2TP's tunnel with IPSec's secure channel which increases the overhead packets. So, L2TP/IPSec on both windows server 2003 and fedora core 6 has produced a good performance values.
- The performance values of both PPTP and L2TP/IPSec on windows server 2003 are better than the performance values of both PPTP and L2TP/IPSec on fedora core 6.
- Because OpenVPN was written as a user space daemon rather than a kernel module, OpenVPN has produced a lower performance values in high traffic environments.
- Due to the design process and the lowest price of e-Live IP-8000VPN Router, its performance values are less than the performance values of both windows server 2003 and fedora core 6.
- The OpenVPN needs to be manipulated to improve its performance values in high traffic environments.
- Testbeds performance values indicate that the deployment of VPNs on a wireless network infrastructure could be considered as an acceptable choice to secure transmission between wireless clients and their enterprise network.
- This work should be extended to include performance evaluation on other software-based routers (such as BSD, Mac, Solaris, and Linux) and hardware-based routers (such as Cisco, 3Com, Juniper, and ADTRAN).

REFERENCES

[1] Mahbod Travallaee, "An Overview of WLAN Authentication Protocols," Faculty of Computer Science, University of New Brunswick. www.researchgate.net/publication/228885695_An_Overview_of_WLAN_Authentication_Protocols, last visited: July 2014.

[2] G. Eason,VBB Sabine Kébreau, Barbu Constantinescu, Samuel Pierre "A New Security Approach for WLAN" pp. 1801-1804, May 2006.

[3] Jon C. Snader, "VPNs ILLUSTRATED: Tunnels, VPNs, and IPSec," Addison-Wesley, 2006.

[4] K. Hamzeh, G. Pall, W. Verthein, J. Taarud, W. Little, and G. Zorn, "Point to Point Tunneling Protocol (PPTP)," IETF RFC 2637, July 1999. www.ietf.org/rfc/rfc2637.txt, last visited: July 2014.

[5] W. Townsley, A. Valencia, A. Rubens, G. Pall, G. Zorn, and B. Palter, "Layer Two Tunneling Protocol (L2TP)," IETF RFC 2661, August 1999. http://www.ietf.org/rfc/rfc2661.txt, last visited: July 2014.

[6] "IPSec," IETF RFCs 2401-2411, and 2451, 1999. www.ietf.org/rfc/, last visited: July 2014.

[7] B. Patel, B. Aboba, W. Dixon, G. Zorn, and S. Booth, "Securing L2TP using IPSec," IETF RFC 3193, November 2001. www.ieft.org/rfc/rfc3193.txt, last visited: July 2014.

[8] Freier, P.karlton, and P. Kocker, "The SSL protocol: Version 3.0 <draft-freier-ssl-version .txt>," IETF RFC-DRAFT, November 1996. www.ietf.org/rfc/rfc2547.txt, last visited: July 2014.

[9] "Installing Windows Server 2003 as a Domain Controller," Microsoft Corporation, September 2004. www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/directory/activedirectory/stepbystep/domcntrl.mspx, last visited: July 2014.

[10] "Virtual Private Networking with Windows Server 2003: Deploying Remote Access VPNs," Microsoft Corporation, October 2005. http://technet2.microsoft.com/windowsserver/en/library/7159a5cd-530b-4b8f-b54a-9a8adfdeac1b1033.mspx?mfr=true, last visited: July 2014.

[11] J. Neil, "OpenVPN Windows HowTo," RunPCRun, December 2006. www.runpcrun.com/howtoopenvpn, last visited: July 2014.

[12] F. Timme, "The Perfect Desktop - Part 1: Fedora Core 6," HowToForge, February 2007. http://www.howtoforge.com/the_perfect_desktop_fedora_core6, last visited: July 2014.

[13] W. Kwok, "PopTop + MSCHAPv2 + Samba + Radius + Microsoft Active Directory + Fedora Howto," September 2005. http://www.members.optushome.com.au/~wskwok/poptop_ads_howto_1.htm, last visited: July 2014.

[14] "Openswan and l2tp on Gentoo," January 2007. http://teh.sh.nu/HowTo/openswan-l2tp.html, last visited: July 2014.

[15] "OpenVPN HOWTO," OpenVPN Inc, 2008 http://openvpn.net/index.php/documentation/howto.html, last visited: July 2014.

[16] "IP Performance Metrics (IPPM) Working Group," IETF, February 2008. http://www.ietf.org/html.charters/ippm-charter.html, last visited: July 2014.

[17] "Iperf – The TCP/UDP Bandwidth Measurement Tool," http://sourceForge.net, last visited: July 2014.

[18] C. Lueders "hrPING v2.38,"cFos Software, http://www.cfos.de/ping/ping.htm, last visited: July 2014.