Virtual Private Networks over a Wireless Infrastructure: Evaluation and Performance Analysis

KUMUDU S. MUNASINGHE and SEYED A. SHAHRESTANI
School of Computing and Information Technology
University of Western Sydney, Locked Bad 1797
Penrith South DC, NSW 1797, Australia

Abstract: - This paper presents the analysis and experimental results for an evaluation of the performance of a Virtual Private Network (VPN) implementation over an IEEE 802.11b wireless infrastructure. The analysed performance measures comprises of application throughput, packet loss, round-trip delay and jitter. Furthermore, the contribution of the CPU, inter-packet generation rate, payload data size, and the number of simultaneously operating VPNs are investigated. The overall results and analysis of the investigations reflect the degree of contribution of the CPU processing power, payload data size, and packet generation rate on the performance of such VPN tunnel implementations.

Key-Words: - Virtual private networks, Wireless networking, Quality of service, Performance analysis.

1. Introduction

With the rapidly increasing acceptance of the Wireless Local Area Network (WLAN) technology security remains as an issue of the highest concern. The main reason for highest priority in security is the susceptibility of the wireless media to a number of possible security threats [1], [2]. The quest for interim solutions for securing WLANs gave rise to many security mechanisms. One such solution is the implementation of Virtual Private Networks over the wireless network. However, the implementation of a VPN over a wireless link may diminish its performance levels. Therefore, it is important for wireless network operators and application developers to understand the behavior trends of such performance parameters.

In our previous publications, performance issues and bottlenecks relevant to the use of a single and multiple Internet Protocol Security (IPSec) VPN tunnel in an IEEE 802.11b WLAN have been analyzed and reported [3], [4]. In this paper, the results of expanding our in-depth analysis to further investigate the effects of the CPU, inter-packet generation rate, payload data size, and the number of simultaneously operating VPNs in a wireless environment are reported. The remainder of this paper is organized as follows. The next section briefly discusses the setup used for measurements and the method of experimentation. In Section 3 an extensive analysis of the performance results are presented. Finally, the last section adds concluding remarks.

2. Experimental Platform and Methodology

The experimental setup includes two desktop PCs as shown in Figure 1. Both the wireless and the wired clients use MS Windows 2000 SP2. The wired client and the IEEE 802.11b Access Point (AP) are both connected to the network via 100 Mbps Ethernet interface cards. The wireless client has an IEEE 802.11b 11Mbps interface configured in infrastructure mode. A separate IPSec policy is configured for each VPN with pre-shared key authentication. IPSec, standardized by the Internet Engineering Task Force (IETF), is a suit of protocols that is widely used in VPNs to provide encryption, authentication and integrity services. Two of the main protocols defined in IPSec are Authentication Header (AH) [5] and Encapsulating Security Payload (ESP) [6].

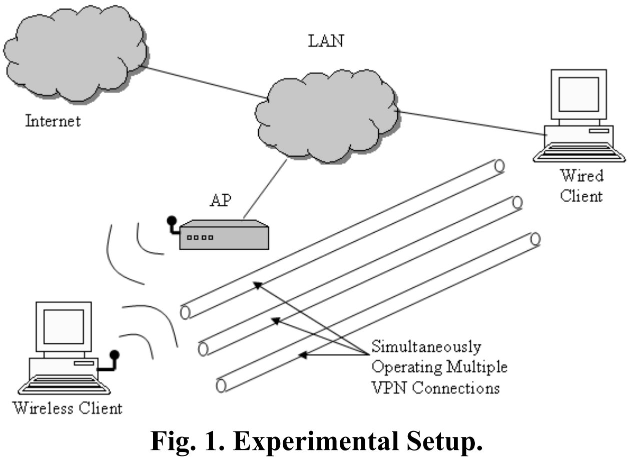

Fig. 1. Experimental Setup.

For traffic generation and capturing, LanTraffic V2 is used. The first stage of the investigation is to establish the baseline conditions. The wireless client is configured to generate streams of UDP traffic to the wired client. The destination captures the incoming traffic and then echoes it back to the source as shown in Fig. 1. UDP packet flows are generated with fixed payload and inter-packet generation gap of 1ms. Mean value of the measurements on application throughput, packet loss, round-trip delay, and CPU utilisation at the source are taken.

In the next stage, a single IPSec VPN is setup and the same steps are repeated. Subsequently, as shown in Fig. 1, the number of simultaneously operating IPSec VPNs are increased to two and then to three respectively. The same steps as in the original experiment are repeated on the multiple VPN setups. Finally, the complete experiment is repeated for an increased inter-packet generation gap of 5ms.

3. Result Analysis

In this section a detailed analysis of the performance results from the experimentation described in the previous section is present. In most cases, the performance results are plotted, as graphs provide for easy comparisons and quick references.

3.1 Throughput

The measured average throughput can be defined as the average amount of data payload transferred over the time duration between two points in the same service area [7]. Figures 2 and 3 represent the average throughput graphs for multiple IPSec VPNs for UDP traffic flows with 1ms and 5ms inter-packet generation gaps respectively.

The results in Fig. 2 indicate that the baseline value for maximum achievable average throughput, under the given conditions, is 5.97 Mbps. This result is in line with some of the previously published results [8], [9]. When a single IPSec VPN is activated, this value drops to 4.94 Mbps. Fig. 3 shows that for increased inter-packet generation gaps, the corresponding highest achievable average throughput for baseline and the single IPSec VPN setups have relatively decreased. Nevertheless, it is worth noting that as a result of preambles, MAC headers, ACK frames, protocol overheads, processing and forwarding delays, and back off periods a substantial amount of throughput is compromised [9], [10].

The graphs in Fig. 2 reflect that there may be a possibility for reduction in the per VPN average throughput as the total number of simultaneously operating VPNs increase. Fig. 2 also reveals that for a UDP flow with payload sizes up to 150 bytes, the throughput graphs behave in the opposite direction. Within this region, the highest total average throughput values in Fig. 3 are recorded for the setup of 3 simultaneously operating IPSec VPNs. Furthermore, this region closely resembles the throughput graphs up to 1000 bytes in Fig. 3. These clearly illustrate how the peak performing multiple VPN setup increasingly suffers reductions in the rate of change (increase) in throughput. Refer to case in Fig. 3; this effect is experienced by UDP traffic with payloads of 600 bytes and over. Similar to Fig. 3, Fig. 2 also experiences a similar effect, at a much earlier stage. Form these two cases; it is clear that the packet generation rate and the packet payload size are the two dominant factors affecting the performance of simultaneously operating multiple VPN setups.

In all the above cases, as the throughput values reach a maximum point, a sudden (but temporary) drop is experienced. This is related to the fragmentation of the IP datagrams. Payloads larger than 1472 bytes get fragmented into more than one datagram reducing the net throughput rapidly [14]. A similar behaviour can be noticed for the average throughput graph of the IPSec VPN. Due to ESP headers, however, this happens when payload increases beyond 1438 bytes.

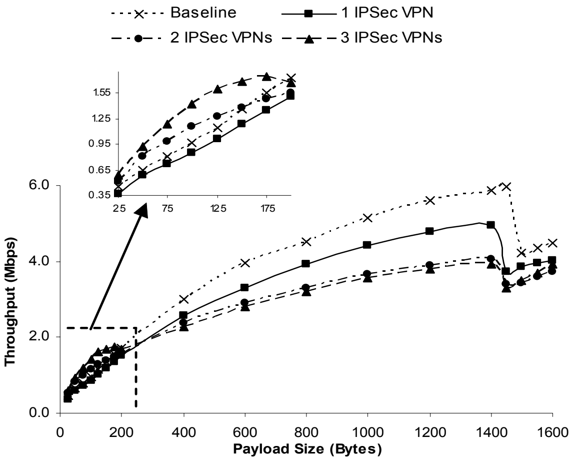

Fig. 2. The Average Throughput Graphs for Traffic Generated with a 1ms inter-packet Generation Gap.

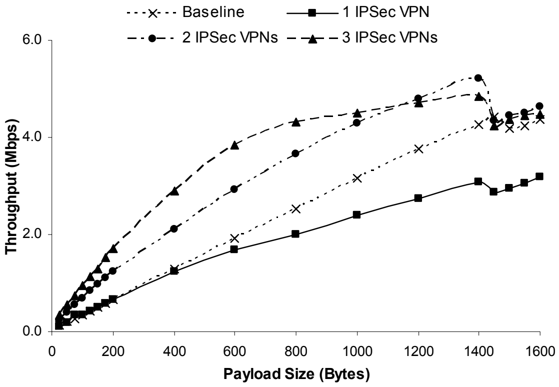

Fig. 3. The Average Throughput Graphs for Traffic Generated with a 5ms Inter-packet Generation Gap.

3.2 Packet Loss

The packet loss metric studied here, investigates the average per tunnel packet loss in transmission (outbound) and receiving (inbound). These relate to the wireless client for UDP traffic generated at two different rates represented by Figures 4 to 7.

As it is clear from Figures 4 and 5 there is a high packet loss (up to approximately 35% in transmission and 20% in receiving) experienced for UDP datagrams with light payloads (i.e., 25 to 50 bytes), generated at 1ms intervals. Similar trends in packet loss have already been identified and published [11],[12],[13]. The most reliable explanation to this phenomenon points towards the behaviour of the lower layers, i.e., data link layer or physical layer [12], [13]. It is argued that, when a wireless interface card encounters frames with relatively short payload sizes, the overheads may occupy most of the frame. The continuous bidirectional traffic causes increase in the total traffic and contention. The processing ability of the interface decelerates and the packets eventually drop off. Fig. 4 further indicates that as the number of simultaneously operating VPNs is increased, how this situation becomes worse. However, since a bursty traffic flow is not used, UDP buffer overflow cannot be ruled out as contributor for packet loss as pointed out in [11].

Comparing the results in Figures 4 and 5 with those in Figures 6 and 7 indicate that, for traffic with smaller inter-packet generation gaps, packet loss is higher. Furthermore, as the UDP payload size is gradually increased, the packet loss also increases. This is again in line with previous works in the field [11]. The cause can be explained in the following manner. As the payload data size increase the transmission delay at the interface increases. As a result, at high packet generation rates (i.e., at 1ms inter-packet delays), relatively larger packets may experience increased queuing delays. Consequently, a bottleneck situation is formed at the wireless interface. In general, when a UDP datagram is delayed as a result of queuing up to the maximum delay limit, it may get dropped [14].

Despite the high packet generation rate, it can be noted that IPSec VPNs in Fig. 4 achieve relatively low packet losses compared to its baseline situation. This condition only applies to UDP traffic with payloads of 400 bytes and over. This can be explained by noting that, at the network layer, a UDP datagram spends a comparatively longer time for the IPSec encryption process. This causes the fast UDP flow to actually slow down. This prevents or at least reduces the queuing at the interface. As a result, the packet losses are relatively less for an IPSec VPN, even at high packet generation rates. Fig. 6 shows how packet loss reduces to a minimum as the inter-packet delay of the UDP traffic is increased. As discussed before, as the inter-packet generation gap increases to 5ms, queuing at the interface may eventually be eliminated. This results in very low levels of packet loss.

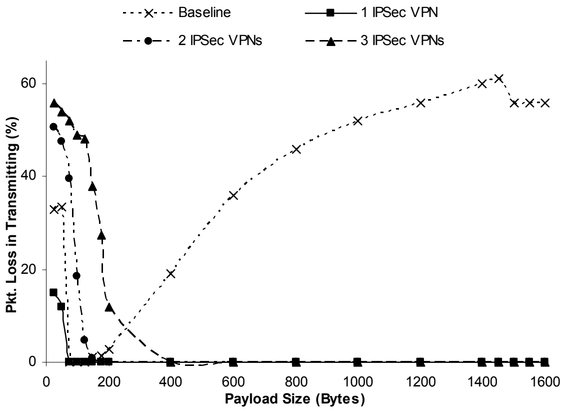

Fig. 4. Packet Loss in Transmitting Traffic Generated at 1ms Inter-packet Delay.

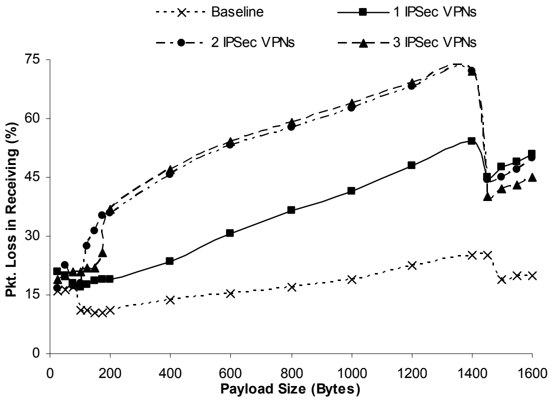

Fig. 5. Packet Loss in Receiving Traffic Generated at 1ms Inter-packet Delay.

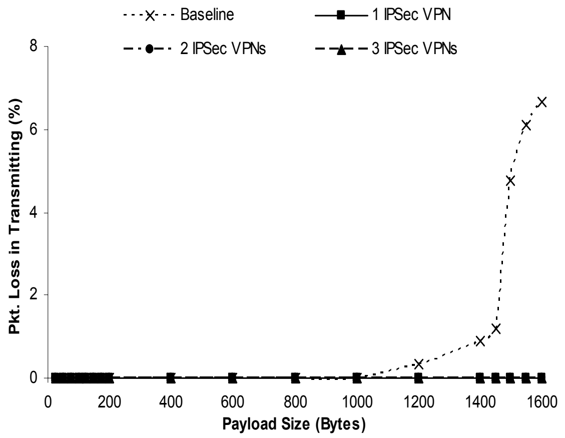

Fig. 6. Packet Loss in Transmitting Traffic Generated at 5ms Inter-packet Delay.

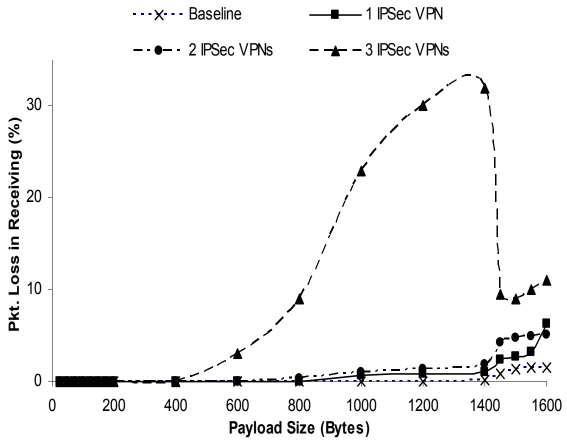

Fig. 7. Packet Loss in Receiving Traffic Generated at 5ms Inter-packet Delay.

A closer look at the inbound traffic in Fig. 7 indicates that when three VPN tunnels are functioning, a significant growth in packet loss for payload data over 600 bytes can be noted. Furthermore, referring back to Fig. 3, it can be noted that for payload sizes over 600 bytes, the same setup experiences a reduction in the rate of change (increase) in throughput. Thus, Fig. 8 illustrates how the packet loss in simultaneously operating IPSec VPNs increase with throughput. As the throughput approaches its peak, the packet loss for three simultaneous VPN implementations shows a noticeable increase. This also indicates that extreme throughput performance levels may result in higher packet loss rates for multiple VPN tunnel implementations. All packet loss results shown in Figures 4 to 7 indicate sudden, but temporary, changes at the point of fragmentation similar to the results in Figures 2 and 3.

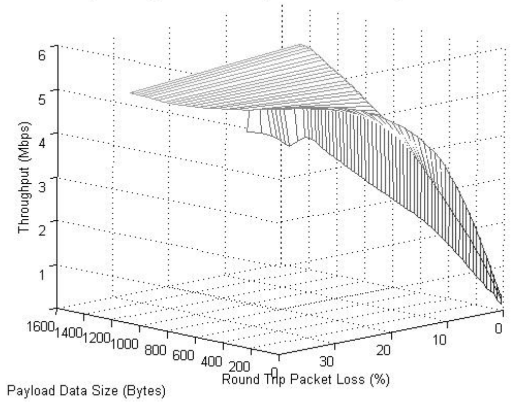

Fig. 8. Variation of Packet Loss and Throughput against Payload Data Size

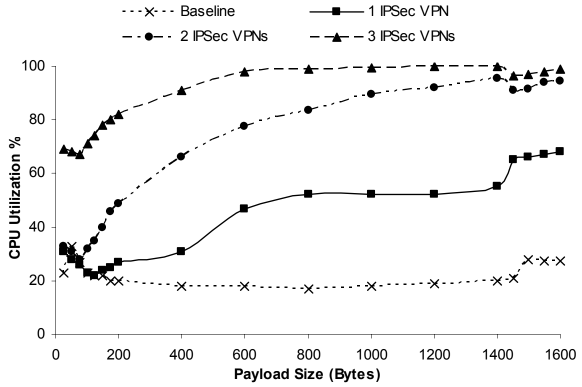

Fig. 9. CPU Utilisation for Traffic Generated at 5ms Inter-packet Delay.

3.3 CPU Utilisation

The results of average CPU utilisation at the wireless client for a UDP flow generated with an inter-packet delay of 5 ms are shown in Fig. 9. It is evident that a considerable number of CPU cycles are necessary for the functioning and implementation of even a single IPSec VPN. As the number of simultaneous VPN implementations is increased, the CPU utilisation further increases. Fig. 9 shows 100% CPU utilisation for payloads of 1200 and 1400 bytes when 3 VPNs operate simultaneously.

Fig. 10 illustrates an integration of results of Figures 3 and 9. This exemplifies the effect of the CPU on the throughput of simultaneously operating IPSec VPNs. The 3D mesh graph represents the three throughput curves; a single, two and three simultaneous IPSec VPNs respectively. As the CPU approaches full utilisation (i.e., close to 100 %), the rate of increase in throughput shows a noticeable reduction. Therefore, it is clear that the CPU is a major contributing factor to the throughput performance of an IPSec VPN. It also indicates that as the number of simultaneously operating tunnels are increased, the per tunnel throughput performance degrades.

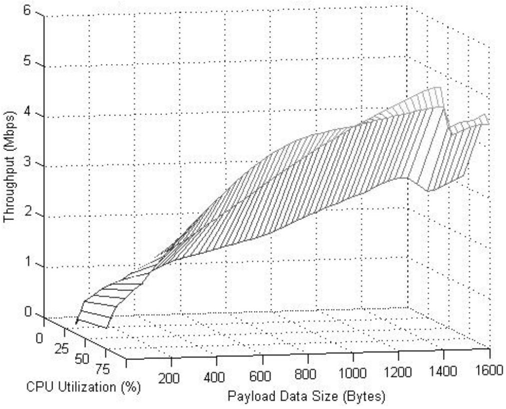

Fig. 10. Variation of Throughput and CPU Utilisation against Payload Data Size.

The effect of CPU on packet loss of simultaneously operating IPSec VPNs is illustrated by Fig. 11 by integrating the graphs on Figures 7 and 9. The 3D mesh graph represents the three packet loss curves; a single, two and three simultaneous IPSec VPNs respectively. As the CPU approaches full utilisation (i.e., close to 100 %), the packet loss shows a noticeable increase. Therefore, it is clear that the CPU may eventually become a major limiting factor to the packet loss of an IPSec VPN.

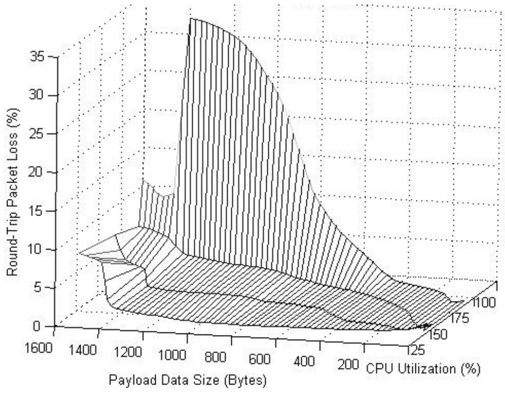

Fig. 11. Variation of Packet Loss and CPU Utilisation against Payload Data Size.

3.4 Round-Trip Delay and Jitter

Figures 12 and 13 represent the round-trip delay curves for UDP traffic with inter-packet delays of 1ms and 5ms respectively. The above graphs show that, at faster packet generation rates, the average time for a given UDP datagram to complete a round-trip is relatively longer. As mentioned in section 3.2, at faster packet generation rates, there is a relatively larger queue forming at the interface. Therefore, an average packet usually experiences a higher amount of queuing delay before it is actually transmitted by the interface. Thus, relatively high round-trip delays can be noted for VPNs in Fig. 12 in comparison to those shown in Fig. 13.

Finally, Fig. 14 illustrates the effect of the CPU on the round-trip delay of simultaneously operating IPSec VPNs. They correspond to the baseline, a single, double, and triple IPSec VPN connections respectively. As the CPU approaches full utilisation (that is, close to 100%), the round-trip delay shows a noticeable increase. Moreover, this phenomenon can only be noticed for relatively large payload sizes (that is, 600 bytes and above). Hence it is clear that the CPU can act as a major limiting factor on the round-trip delay of an IPSec VPN.

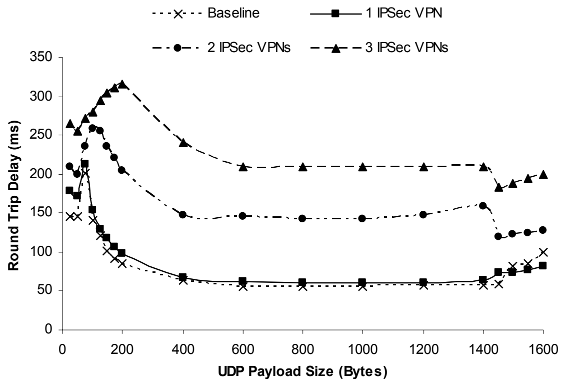

Fig. 12. Round-Trip Delay for Traffic Generated at 1ms Inter-packet Delay.

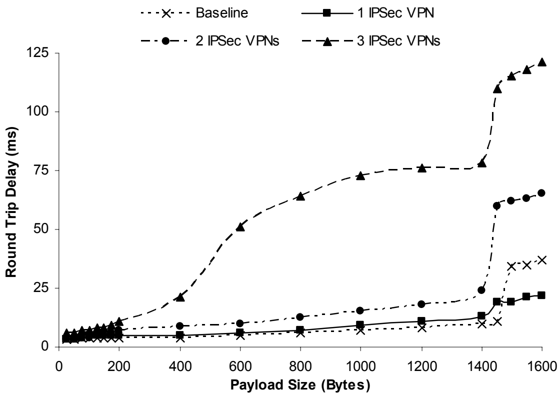

Fig. 13. Round-Trip Delay for Traffic Generated at 5ms Inter-packet Delay.

Jitter is the mean variation of delays on packets received. The delay metric considered in computing jitter is the round-trip delay measured and analysed in Section 3.4. Figures 15 and 16 represent the jitter graphs for UDP traffic with inter-packet delays of 1 ms and 5 ms respectively. These graphs show that at faster packet generation rates, the average variation of delay for a UDP datagram to complete one full round trip is relatively longer. As mentioned previously, at faster packet generation rates, there are relatively larger queues and delays forming at the interface in multiple VPN setups. This delay also seems to increase with the increasing number of VPNs. Therefore, an average packet usually experiences a higher amount of queuing delay before it is actually transmitted by the interface. Thus, relatively high jitter can be noted for VPNs in Fig. 15, in comparison to those shown in Fig. 16.

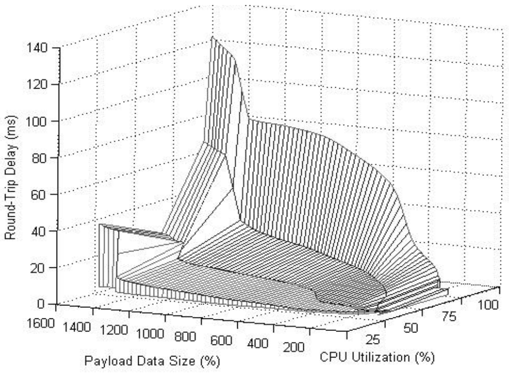

Fig. 14. Variation of Round-Trip Delay and CPU Utilisation against Payload Data Size.

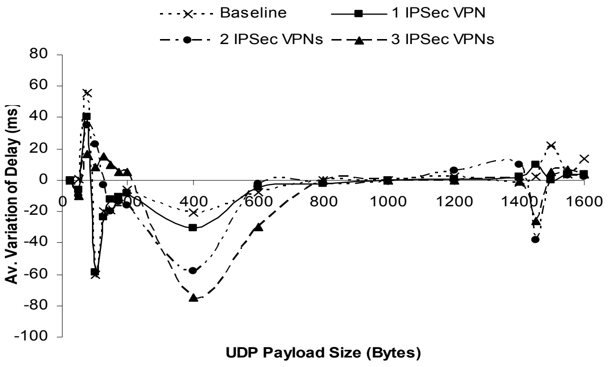

Fig. 15. Jitter for Traffic Generated at 1ms Inter-packet Delay.

4. Conclusions

In this paper, the analysis and experimental results for an evaluation of the performance of VPNs over a wireless infrastructure are presented. The payload data size, inter-packet generation rate, number of simultaneously operating VPN connections, and CPU processing power are identified as major contributing factors towards the performance of a wireless VPN. The results of the analysis reflect that the throughput, packet loss and delay show an increasing trend as the payload data size and the inter-packet generation rate of the data flow increases. The analysis also explains how a data flow with a relatively shorter inter-packet generation gap may give rise to a potential bottleneck at the interface, which will eventually contribute to the packet loss and delay.

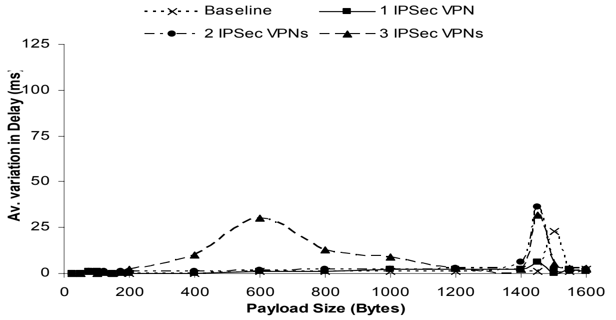

Fig. 16. Jitter for Traffic Generated at 1ms Inter-packet Delay.

References:

[1] W. A. Arbaugh, et al., "Your 802.11 Wireless Network has No Clothes," *IEEE Wireless Communications,* vol. 9, 2002, pp. 44-51.

[2] N. Borisov, et al., "Intercepting Mobile Communications: The Insecurity of 802.11," in *Proc. Int. Conf. on Mobile Computing and Networking*, 2001, pp. 180-189.

[3] K. S. Munasinghe and S. A. Shahrestani, "Evaluation of an IPSec VPN over a Wireless Infrastructure," in *Proc. ATNAC*, 2004, pp. 315-320.

[4] K. S. Munasinghe and S. A. Shahrestani, "Analysis of Multiple Virtual Private Network Tunnels over Wireless LANs," in *Proc. IBIMA*, 2004, pp. 206-211.

[5] S. Kent and R. Atkinson, "IP Authentication Header," in *IETF: RFC 2402*, 1998.

[6] S. Kent and R. Atkinson, "IP Encapsulating Payload," in *IETF: RFC 2406*, 1998.

[7] S. Ci and H. Sharif, "A Link Adaptation Scheme for Improving Throughput in the IEEE 802.11 Wireless LAN," in *Proc. IEEE LCN*, 2002, pp. 205-208.

[8] M. G. Arranz, R. Aguero, L. Munoz, and P. Mahonen, "Behavior of UDP-based Applications over IEEE 802.11 Wireless Networks," in *Proc. IEEE PIMRC*, 2001, pp. F72-F77.

[9] B. Bing, "Measured Performance of the IEEE 802.11 Wireless LAN," in *Proc. IEEE LCN*, 1999, pp. 34-42.

[10] A. Kamerman and G. Aben, "Throughput Performance of Wireless LANs Operating at 2.4 and 5 GHz," in *IEEE PIMRC*, 2000, pp. 190-195.

[11] G. Xylomenos and G. C. Polyzos, "TCP and UDP Performance over a Wireless LAN," in *Proc. IEEE INFOCOM*, 1999, pp. 439-446.

[12] J. Wu and J. Ilow, "A Wireless Multimedia LAN Testbed," in *Proc. Canadian Conf. on Elect. and Comp. Eng.*, 2000, pp. 826-830.

[13] J. Lee, G. Kim, and S. Park, "Optimum UDP Packet Sizes in Ad-hoc Networks," in *Proc. Merging Optical and IP Technologies Workshop*, 2002, pp. 214-218.

[14] M. Petrovic and M. Aboelaze, "Performance of TCP/UDP under Ad-hoc IEEE 802.11," in *Proc. ICT*, 2003, pp. 700-708.