Communications
2015; 3(1): 11-23
Published online July 23, 2015 (http://www.sciencepublishinggroup.com/j/com)
doi: 10.11648/j.com.20150301.12
ISSN: 2328-5966 (Print); ISSN: 2328-5923 (Online)

Investigating Space Overhead by IPSec on IPv4 and IPv6 Communication Protocols

Muhammed Nura Yusuf, Ali Mohammed Baba

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University Bauchi

Email address:
yudasco@yahoo.com (M. N. Yusuf), ambabastas@gmail.com (A. M. Baba)

To cite this article:
Muhammed Nura Yusuf, Ali Mohammed Baba. Investigating Space Overhead by IPSec on IPv4 and IPv6 Communication Protocols. *Communications*. Vol. 3, No. 1, 2015, pp. 11-23. doi: 10.11648/j.com.20150301.12

Abstract: IPsec, an internet layer three-security protocol suite is often characterised with introducing an additional space and processing overhead when implemented on a network for secured communication using either IPv4 or IPv6. The use of IPsec on IPv4 is an alternative that offers solutions and addresses the security vulnerabilities in network layer of the OSI and TCP/IP protocol stack. In IPv6, IPsec is one among many other features added to the earlier internet protocol to enhance efficiency and security. Analysis in this research aim at observing the effect of additional space overhead incurred by internet protocols version 4 and 6 (IPv4, IPv6) as a result of selected IPsec configuration in relation to payload size in transport and tunnel mode of IPsec. It was observed that the cost of IPsec added overhead is relatively small when smaller packet sizes are involved for both protocols comparison with large packet sizes that were IPsec protected with the same configuration as the smaller packet, unless in the cases whereby the packet was very large which has to be fragmented. It is therefore, a guide for network administrators to trade up between processing cost and larger address space among other improvements specifically for transmission involving larger IP packets.

Keywords: IPv4, IPv6, IPsec, Transport Mode, Tunnel Mode

1. Introduction

Analysis of IPSec overheads has generated significant amount of research interest over the years. There are various publications of technical and peer reviewed papers and thesis that already worked on the area. This discussion and analysis cover areas such as basic network protocol performance ranging from protocol latency, throughput, CPU utilization of protocols, to TCP/IP IPSec protocol processing overheads

(Meenakshi 2006) in his work described protocol latency as the duration period whereby an IP Packet or datagram is kept on hold by certain network protocol or predecessor layer for processing prior to pushing it further to awaiting network component or next protocol. It was therefore, narrated that this brings about a processing overheads that the protocol introduces to the performance of the entire network, thus, degrading its efficiency, exerting more effect on a multilayer protocol hierarchy. According to him, this latency can be either start latency or stop latency. Start latency is described as the time between the initial bit of the payload data arriving at the top of the stack and it is part of stop latency. Stop latency refers to the time interval between the initial bit of the data payload arriving at the header of the protocol stack and up to the last bit of the payload going out through the bottom of the stack (Meenakshi 2006).

Hence in mathematical expressions it was given as:

$$\textit{Start latency } \Delta tstart = \textit{(time difference between } \Delta I \textit{ reaches SPn and } \Delta i \textit{ exits SPi)}$$

$$\textit{Stop latency } \Delta tstop = \textit{(time difference between } \Delta \textit{ reaches SPn and } \Delta n \textit{ exits SPi)}$$

$$\textit{Total latency } \Delta ttot = (\Delta tstop \textit{ (server side)} + \Delta tstop \textit{ (client side)} + \textit{transmission delay)}$$

Another performance parameter is throughput. In the same work, (Meenakshi 2006) described it as the amount of transmitting data a channel can accommodate and process it at a particular point in time; it is measured in Kbps. This is seriously affected as a result of any additional overheads that might be encountered or introduced on a network (Meenakshi 2006).

On the other hand, (Elkeelany *et al* 2002), on analytical performance investigation of IPSec overheads approached it with respect to time and space complexity. Ciphering

algorithms such as triple DES and authentication algorithms like HMAC-MD5 and HMAC-SHA1 were investigated in relation to AH and ESP, in the two modes of IPSec. Their overall effects were determined and compared.

(Seiji et al 2000) approached it experimentally and investigated the performance of IPSec over IPV4 with the objective of evaluating the end-to-end throughput of transmitting a very large data and real time traffic such as video application. Here is what he had found summarized in table 1.

Table 1.0 *Summarized Result of End-To-End Throughput of Transmitting a Very Large Data and Real Time Traffic Such As Video Application*

Measurement	Result/Comment
Larger Data Transfer	The throughput deteriorates to 1/9 when authentication and encryption are enforced.
Digital Video Transfer	While here, the throughput depreciates to 1/10 when authentication and encryption are enforced.
The overall end-to-end throughput deterioration discrepancy between the two IP protocols.	The difference is minor.
With IPSec enabled the divergence with respect to end-to-end throughput of TCP and UDP protocols.	It is equals
However, the throughput reduction in relation to response mode is substantially higher in than in the stream mode	

(Seiji et al 2002)

Similarly (George 2003) concentrated on looking at the constituents of the overheads IPv4 with IPSec enabled introduce to an email and web application over a wired and wireless communication channel using HMAC-MD5 and HMAC-SHA1 for authentication and 3DES for encryption. Below is what they had to presents.

In the case of wired Network channel:

1. With the amalgamation of different encryption and authentication algorithms for transmitting HTTP and SMTP application protocol with file size higher than 10kb, the number of transaction increase by 5% between client and the server.
2. While in the events of client machine speed slower than the server, the number of time at which transaction takes place are 22% for SMTP and 10% for HTTP protocol
3. The network load increased between the range of 20 to 30% for 1KB and about 5% for 10MB when HTTP is in Play
4. While for SMTP the network load experience an additional increase of 31 to 40% for 1 KB and 6 to 12 % for 10MB
5. Meanwhile the increase in transfer time is between 18 to 52% for 1KB and 6 to 12% for 10MB files
6. While SMTP recoded an increase in transfer time with 3 to 11% for 1 KB and 1 to 6% with 10MB files

In the other case, the Wireless Network channel:

1. Even with the IPSec in play, there is no increase in the number of transactions
2. Network load is uniform for both slow and fast client
3. But transfer time is increased by about 6 to 285 for 10KB and 5 to 6% for 1MB with HTTP while 6 to 14% for 10KB and 3% for 1MB with SMTP. (George 2003)

On a similar approached (Lin 2003) investigated a VPN throughput of routers setup with IPv4 using IPSec technology, considering various file size to measure the overall throughput for FTP traffic and HTTP application protocols. For authenticating the connection HMAC-MD5 and HMAC-SHA1 algorithms were used while DES and 3DES were used for encryption. The outcome of the work is summarized as follows:

- Throughput disparity between FTP and HTTP protocol for file size of 100MB indicates that FTP Protocol is greater than HTTP throughput by 1 to 3%
- Using HMAC-MD5 authentication of file size 1MB the throughput of HTTP exceed the one of FTP by 1 to 2%. While it is almost equal when HMAC-SHA1 is used
- In summary the average deterioration of the throughput is 1/3 with HMAC-MD5, 1/3.5 for HMAC-SHA1,1/7 for DES plus MD5,1/8 for DES plus SHA1, 1/9 for 3DES plus MD5 and lastly 1/10 for 3Des plus SHA1 to all the application (FTP and HTTP)

For real time application such as VoIP and video conferencing, (Klause 2005) looked at the effects overheads have on measurement parameters such as data loss, jitter and delay in relation to quality of service. Briefly to summarize their findings, the IPSec overheads do not attribute any effect on perceptual quality, and the measurement parameters investigated did not experience any substantial disparity for packet greater than 256bytes with IPSec enabled or not enabled. (Klause 2005)

2. IPSec and Its Components

IPSec is an IP network mechanism that operates at the network layer of the TCP/IP and OSI protocol stack (Mujinga et al 2006). It is a standard that offers security parameters at the network layer of IP base network for secured end-2-end data transmission. It inspects the process of authentication for the communicating parties, and engages the services of various cryptographic algorithms to provide confidentiality and anti-replay attack to the data. IPSec is a tool that can be used to bring virtual private network (VPN) into being in an IPv4-based network (Wenhong et al 2006). The VPN allows you to create a secured and private communicating channel using unsecured and public network medium such as the Internet. The use of IPSec is optional in IPv4-based network, depending on the level of priority given to security at the network layer by users and their application in a given network. IPv4 in its plain form is noted to be vulnerable, susceptible and defenceless to any possible adversary attack targeted at the network layer of the OSI and TCP/IP protocol stack. I.e. attack such as IP spoofing, ping of death and cache

poisoning (Cheng 2011) etc. But the emergence of IPSec offers IPv4 the option to alleviate such threats and immunize itself against the possible attacks. This is achieved by adding IPSec headers to the IPv4 packets in VPN connections and provides secure tunnelling protocol connection between the communicating parties with the help of its two integral protocols, authentication header protocol (AH) and encapsulating security payload (ESP). However, unlike in IPv4, IPSec is an integral part of IPv6 protocol (Cheng 2011). It is one of the additional features introduced to the protocol to enhance network security. In IPv6 the implementation of IPSec is necessary and is achieved with the help of two sets of protocols; authentication header (AH) and encapsulating security payload (ESP) both of which are integral part of IPSec protocols that offer the choices of selecting desired security services available. Together they make up the IPSec transform, without them IPSec cannot give us the primary service it was developed for (Lammle 2010).

Similarly, it is important to note, that IPSec is compatible to operate in two different modes, the transport mode and the tunnel mode. AH and ESP can operate in both modes, and each of the mode has its own peculiar characteristics and uses that to distinguish itself from the other. The choice of the mode depends on the network design implementation and the path/route of the transmitted data. In transport mode IPSec provide its security to the end-2-end points of the transmission by applying encryption algorithm to the IP packet payload only. In other words, it encrypts the actual datagram only while leaving the IP header as open plain text. Contrary to the transport mode, in IPSec tunnel mode, the encryption algorithm is applied to the entire IP packet, both the payload and the IP header inclusive, and then encapsulates it in a new IP header, this means that no portion of the IP packet is exempted from IPSec protection during transmission, (Mujinga et al 2006), (Christos et al 2006).

As stated earlier, IPSec employ the service of two protocols to ensure the end-2-end secure channel for communication. These includes; Authentication Header (AH) and Encapsulating Security Payload (ESP) (Mujinga et al 2006). The AH contributes to sustain data origin authentication, connectionless integrity and an optional anti-reply service, but without one for confidentiality. AH render these services by generating a one-way hash function that is identical from the sender and the receiver. If the one-way hash function were to change in any way the packet originality couldn't be establish and authenticated, therefore, the packet would be dropped instantly (Lamme 2010). Similarly, for multicast transmission one-way hash function may be merged with a symmetric signature algorithm, hence performance and space constraints make the utilization of such algorithm impracticable, thus hindering it deployment (IETF/RFC 4305). However, the encapsulating security payload (ESP) in the other hand, being a twin of AH in IPSec transform, steps up and fills the gap left by AH. It is a protocol designed to maintain data confidentiality, data origin authentication, connectionless integrity, anti-reply immunity service and adjusted traffic flow confidentiality. ESP can be deployed exclusively or in conjunction with AH to form a stronger union of IPSec mechanism for securing the data being transmitted through the channel (Mujinga et al 2006).

The diagrams below demonstrate the IPSec authentication header (AH) transport mode and tunnel mode positioning and size for an IPv4 and IPv6 IP packets (IETF/ RFC 4305)

Fig 1. *IPv4 with IPSec (AH) Total Header Size, Tunnel Mode 64 Bytes.*

	0–3	**4–7**	**8–13**	**14-15**	**16–18**	**19–31**
Original IPv4 Header total Size = 20 bytes	Version (4 bit)	Internet Header Length (4 bit)	Differentiated Services Code Point (8 bit)	Explicit Congestion Notification()	Total Length (16 bit)	
	Identification (16 bit)				Flags (3 bit)	Fragment Offset (13 bit)
	Time to Live (8 bit)		Protocol (8 bit)		Header checksum (16bit)	
	Source IP Address (36bit)					
	Destination IP Address (36bit)					
	Options (if Header Length > 5)					
AH	44 bytes					
User Data	Data ()					

Source: IPv4 (IETF/ RFC 4305)

Fig 2. *IPv4 with IPSec (ESP) Total Header Size, Tunnel Mode 62 Bytes.*

	0–3	**4–7**	**8–13**	**14-15**	**16–18**	**19–31**
Original IPv4 Header total Size = (160 bits) 20 bytes	Version (4 bit)	Internet Header Length (4 bit)	Differentiated Services Code Point(8 bit)	Explicit Congestion Notification()	Total Length(16 bit)	
	Identification (16 bit)				Flags (3 bit)	Fragment Offset (13 bit)
	Time to Live (8 bit)		Protocol (8 bit)		Header checksum (16bit)	
	Source IP Address (36bit)					
	Destination IP Address (36)					
	Options (if Header Length > 5)					
ESP	42 bytes					
User Data	Data ()					

Source: IPv6 (IETF/ RFC 4305**)**

Fig 3. *IPv6 with IPSec (AH) Total Header Size, Transport Mode 64 Bytes.*

	Version (4 bits)	**Traffic class (8 bits)**	**Flow label (20 bits)**
Original	Payload Length (16 bits)	Next Header (8 bits)	Hop Limit (8 bits)
IPv6 Header. total size = 320 bits (40 bytes)	Source address (128 bits)		
	Destination Address (128 bits)		
AH	24 byte		
User Data			

Source: IPv6 (IETF/ RFC 4305)

Fig 4. *IPv6 with IPSec (ESP) Total Header Size Transport Mode 62 Byte.*

	Version (4 bits)	**Traffic class (8 bits)**	**Flow label (20 bits)**
Original	Payload Length (16 bits)	Next Header (8 bits)	Hop Limit (8 bits)
IPv6 Header. Total size = 320 bits (40 bytes)	Source address (128 bits)		
	Destination Address (128 bits)		
ESP	22 byte		
User Data			

Source: IPv6 (IETF/ RFC 4305)

Fig 5. *IPv6 with IPSec (AH) Total Header Size, Tunnel Mode 84 Bytes.*

	Version (4 bits)	**Traffic class (8 bits)**	**Flow label (20 bits)**
Original	Payload Length (16 bits)	Next Header (8 bits)	Hop Limit (8 bits)
IPv6 Header. Total size = 320 bits (40 bytes)	Source address (128 bits)		
	Destination Address (128 bits)		
AH	44 byte		
User Data			

Source: IPv6 (IETF/ RFC 4305)

Fig 6. *IPv6 with IPSec (ESP) Total Header Size, Transport Mode 82 Bytes.*

	Version (4 bits)	**Traffic class (8 bits)**	**Flow label (20 bits)**
Original	Payload Length (16 bits)	Next Header (8 bits)	Hop Limit (8 bits)
IPv6 Header. Total size = 320 bits (40 bytes)	Source address (128 bits)		
	Destination Address (128 bits)		
ESP	42 byte		
User Data			

Source: IPv6 (IETF/ RFC 4305).

3. Methodology for the Investigation

IPSec protected packet increased in final size leads to introduction of space overheads. The space overhead is established due to the IPSec supplementary fields that are further added to the plain IP packets to protect it against network layer attacks. The overheads size is relative to the IPSec security protocols adopted and the mode at which IPSec is setup. The investigation was implemented with different packet size, hence the packet size was determine according to the IPSec configuration parameters used (IPSec transform set) to yield the protected packet size. The protected packet introduced an additional space and processing overheads.

The procedure adopted for investigating the IPSec overheads imposed by IPv4 network and comparing the same experienced in IPv6 -based network communication was investigated in the two different modes of IPSec; Transport mode and Tunnel mode, by numerically quantifying the space overhead using the model developed by Christos *et al,* 2006 to arrive at the final size of the protected packets. The model is given in table 2 and table 3.

Nine different IPSec protected user payload files sizes were investigated using different IPSec configuration scenarios in the said modes; (transport mode and tunnel mode respectively). All the files are tested using IPv4 and IPv6. The selection of the files is based on its size. The files are categorized as “small file size”, “intermediate file size” and “large file size”. The small file sizes are 1byte, 10byte and 100byte while the intermediates files are between 1kb to 100kb and the large files are 1MB to 100MB.

The results obtained were subjected to R 3.1.3 statistical package for analysis of variance (ANOVA) to determine the contribution of IPSec configuration sets and the effects of the protocols, payload size on the incurred overhead in both the transport and tunnel mode.

The model for the design is given by

$$AO = \mu + PTC + PLS + IPSec + \varepsilon$$

Where,
AO → additional overhead
μ → constant independent of PTC, PLS, IPSec
PTC→ effect of protocol

PLS →effect of payload size
IPsec→ effect of IPsec configuration set
ε→ Error term

Where significant difference among PTC, PLS, and IPsec is observed, means separation technique using Bonferrion method is used to find the different classes in each case.

Table 2.0. *Description of the symbols and notations used in the model.*

Symbol notations used in the model	Description of the symbols and notations
AuTESP	Size of the ATH data field of the ESP protocol (12 bytes)
BL	Block size of the encryption algorithms (DES = 8 bytes while AES = 16 bytes)
HESP, HIP, HTCP	ESP = 8 bytes, IPv4 header =20 bytes, IPv6 header = 40 bytes and TCP header size = 20bytes
RP (Sd), RL (Sd)	Ratio of the user data to packet length with transport mode and tunnel mode
Sd	Actual user packet size in bytes
SP(Sd), SL(Sd)	Total size of IPSec applied packet in transport and tunnel mode respectively
Tr ESP	ESP Trailer size.

(Christos et,al 2006)

Table 3.0. *Model to Determine the Size of IPSec Applied User Packet.*

Symbols IPv4	The formula IPv4
SPESP-CNF(Sd)	Ceil (Sd + 22)/BL *BL+ 28
SLESP-CNF(Sd)	Ceil (Sd + 42)/BL *BL + 28
SPESP-ATH (Sd)	Ceil (Sd + 22)/4 *4+ 40
SLESP-ATH (Sd)	Ceil (Sd + 42)/4 *4 + 40
SP ESP-CNF-ATH (Sd)	Ceil (Sd + 42)/BL *BL + 40
SL ESP-CNF _ATH (Sd)	Ceil (Sd + 42)/BL *BL + 40
SP AH(Sd)	Sd + 64
SLAH (Sd)	Sd + 84
IPv6	IPv6
SPESP-CNF(Sd)	Ceil (Sd + 22)/BL *BL+ 48
SLESP-CNF(Sd)	Ceil (Sd + 62)/BL *BL + 48
`SPESP-ATH (Sd)	Ceil (Sd + 22)/4 *4+ 60
SLESP-ATH (Sd)	Ceil (Sd + 62)/4 *4 + 60
SP ESP-CNF-ATH (Sd)	Ceil (Sd + 42)/BL *BL + 60
SL ESP-CNF _ATH (Sd)	Ceil (Sd + 62)/BL *BL + 60
SP AH(Sd)	Sd + 84
SLAH (Sd)	Sd + 124

(Christos et,al 200)

4. Result of the Investigations: (Summarize in Table)

Table 4.0. *(Small size data): The Overheads Imposed In Transport Mode by 1byte, 10byte and 100byte IPSec Protected User Data.*

Protocols	Payload/file size	Packet size with no IPSec (bytes)	IPSec configuration set	IPSec Protected Packet size (bytes)	Additional overheads (bytes)	% Of overheads
IPv4	1byte	41	AH	65	24	58%
			ESP-CNF	60	19	46%
			ESP-ATH	64	23	56%
			ESP-CNF-ATH	72	31	75%
IPv6		61	AH	85	24	39%
			ESP-CNF	80	19	31%
			ESP-ATH	84	23	37%
			ESP-CNF-ATH	92	31	50%
IPv4	10byte	50	AH	74	24	48%
			ESP-CNF	60	10	20%
			ESP-ATH	72	22	44%
			ESP-CNF-ATH	72	27	54%
IPV6		70	AH	94	24	34%
			ESP-CNF	80	10	14%
			ESP-ATH	92	22	31%
			ESP-CNF-ATH	92	32	45%
IPv4	100byte	140	AH	164	24	17%
			ESP-CNF	156	16	11%
			ESP-ATH	164	24	17%
			ESP-CNF-ATH	168	28	20%
IPv6		160	AH	184	24	15%
			ESP-CNF	178	16	10%
			ESP-ATH	184	24	15%
			ESP-CNF-ATH	188	28	17.5%

R Table 4 Result: Analysis of variance (ANOVA) and means separation using Bonferrion method.

Df	Sum	Sq	Mean	Sq	F value Pr(>F)
PTCSDS	2	2.0	0.98	0.154	0.8582
PLSDS	2	39.3	19.67	3.101	0.0727.
IPSSDS	3	609.8	203.28	32.051	5.35e-07 ***
Residuals	16	101.5	6.34		

Signif. Codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
LSD t Test for AOHTSDS
P value adjustment method: bonferroni
Mean Square Error: 6.342438
IPSSDS, means and individual (95 %) CI

	AOHTSDS	std	r	LCL	UCL	Min	Max
AH	24.00000	0.000000	6	21.82044	26.17956	24	24
ESP-ATH	23.00000	1.000000	5	20.61241	25.38759	22	24
ESP-CNF	16.14286	4.810702	7	14.12498	18.16074	10	23
ESP-CNF-ATH	29.50000	2.073644	6	27.32044	31.67956	27	32

alpha: 0.05 ; Df Error: 16
Critical Value of t: 3.008334
Minimum difference changes for each comparison
Means with the same letter are not significantly different.
Groups, Treatments and means

a	ESP-CNF-ATH	29.5
b	AH	24
b	ESP-ATH	23
c	ESP-CNF	16.14

Table 5.0. *(Small size data): The Overheads Imposed In Tunnel Mode on 1byte, 10byte and 100byte IPSec Protected User Data.*

Protocols	Payload/file size	Packet size with no IPSec (bytes)	IPSec configuration set	IPSec Protected Packet size (bytes)	Additional overheads (bytes)	% Of overheads
IPv4	1byte	41	AH	85	44	107%
			ESP-CNF	78	35	85%
			ESP-ATH	84	43	104%
			ESP-CNF-ATH	88	47	114%
IPv6		61	AH	124	64	104%
			ESP-CNF	112	51	83%
			ESP-ATH	124	63	103%
			ESP-CNF-ATH	124	63	103%
IPv4	10byte	50	AH	94	44	88%
			ESP-CNF	92	42	84%
			ESP-ATH	92	42	84%
			ESP-CNF-ATH	104	54	108%
IPV6		70	AH	134	64	91%
			ESP-CNF	128	58	82%
			ESP-ATH	132	62	88%
			ESP-CNF-ATH	140	70	100%
IPv4	100byte	140	AH	184	44	31%
			ESP-CNF	172	32	22%
			ESP-ATH	184	44	31%
			ESP-CNF-ATH	184	44	31%
IPv6		160	AH	224	64	40%
			ESP-CNF	224	64	40%
			ESP-ATH	224	64	40%
			ESP-CNF-ATH	236	70	43%

R Table 5 Result: Analysis of variance (ANOVA) and means separation using Bonferrion method.

Df	Sum	Sq	Mean	Sq	F value Pr(>F)
PTCTN	2	2441.1	1220.5	106.543	5.66e-10 ***
PLTN	2	42.5	21.2	1.855	0.188632
IPSTN	3	375.1	125.0	10.915	0.000378 ***
Residuals	16	183.3	11.5		

Signif. Codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
LSD t Test for AOHTN
P value adjustment method: bonferroni
Mean Square Error: 11.45588
PTCTN, means and individual (95 %) CI

	AOHTN	std	r	LCL	UCL	Min	Max
IPv4	42.91667	5.468228	12	40.84538	44.98795	32	54
IPv6	63.00000	5.215362	11	60.83661	65.16339	51	70
IPV6	64.00000	NA	1	56.82485	71.17515	64	64

alpha: 0.05 ; Df Error: 16
Critical Value of t: 2.673032
Minimum difference changes for each comparison
Means with the same letter are not significantly different.
Groups, Treatments and means

a	IPV6	64
a	IPv6	63
b	IPv4	42.92

LSD t Test for AOHTN
P value adjustment method: bonferroni
Mean Square Error: 11.45588
IPSTN, means and individual (95 %) CI

	AOHTN	std	r	LCL	UCL	Min	Max
AH	54	10.95445	6	51.07076	56.92924	44	64
ESP-ATH	53	10.99091	6	50.07076	55.92924	42	64
ESP-CNF	47	12.80625	6	44.07076	49.92924	32	64
ESP-CNF-ATH	58	11.36662	6	55.07076	60.92924	44	70

alpha: 0.05 ; Df Error: 16
Critical Value of t: 3.008334
Least Significant Difference 5.878678
Means with the same letter are not significantly different.
Groups, Treatments and means

a	ESP-CNF-ATH	58
a	AH	54
a	ESP-ATH	53
b	ESP-CNF	47

Table 6.0. *The Overheads Imposed In Transport Modes by 1kb, 10kb and 100kb IPSec Protected User Data.*

Protocols	Payload/file size	Packet size with no IPSec (bytes)	IPSec configuration set	IPSec Protected Packet size (bytes)	Additional overheads (bytes)	% Of overheads
IPv4	1kb (1024 byte)	1064	AH	1088	24	2.3%
			ESP-CNF	1084	20	1.9%
			ESP-CNF	1088	24	2.3%
			ESP-CNF-ATH	1096	32	3.0%
IPv6		1084	AH	1108	24	2.2%
			ESP-CNF	1104	20	1.8%
			ESP-ATH	1108	24	2.2%
			ESP-CNF-ATH	1116	32	3.0%
IPv4	10kb (10240 byte)	10277	AH	10301	24	0.23%
			ESP-CNF	10300	23	0.22%
			ESP-ATH	10300	23	0.22%
			ESP-CNF-ATH	10312	35	0.34%
IPV6		10297	AH	10321	24	0.23%
			ESP-CNF	10320	23	0.22%
			ESP-ATH	10322	23	0.22%
			ESP-CNF-ATH	10322	35	0.23%
IPv4	100kb (102400 byte)	102440	AH	102504	64	0.06%
			ESP-CNF	102460	20	0.02%
			ESP-ATH	102464	24	0.02%
			ESP-CNF-ATH	102472	32	0.03%
IPv6		102460	AH	102524	64	0.06%
			ESP-CNF	102524	20	0.02%
			ESP-ATH	102484	24	0.02%
			ESP-CNF-ATH	102492	32	0.03%

R Table 6 Result: Analysis of variance (ANOVA) and means separation using Bonferrion method.

Df	Sum	Sq	Mean	Sq	F value Pr(>F)
PTCTKB	2	24.6	12.3	0.128	0.8808
PLTKB	2	456.5	228.2	2.373	0.1251
IPSTKB	3	1204.7	401.6	4.176	0.0231 *
Residuals	16	1538.7	96.2		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
LSD t Test for AOHTKB
P value adjustment method: bonferroni
Mean Square Error: 96.16862
IPSTKB, means and individual (95 %) CI

	AOHTKB	std	r	LCL	UCL	Min	Max
AH	37.33333	20.6559112	6	28.84627	45.82040	24	64
ESP-ATH	23.60000	0.5477226	5	14.30289	32.89711	23	24
ESP-CNF	21.42857	1.8126539	7	13.57108	29.28607	20	24
ESP-CNF-ATH	33.00000	1.5491933	6	24.51293	41.48707	32	35

alpha: 0.05 ; Df Error: 16
Critical Value of t: 3.008334
Minimum difference changes for each comparison
Means with the same letter are not significantly different.
Groups, Treatments and means

a	AH	37.33
a	ESP-CNF-ATH	33
a	ESP-ATH	23.6
a	ESP-CNF	21.43

Table 7.0. *The Overheads Imposed In Tunnel Mode by 1kb, 10kb and 100kb IPSec Protected User Data.*

Protocols	Payload/file size	Packet size with no IPSec (bytes)	IPSec configuration set	IPSec Protected Packet size (bytes)	Additional overheads (bytes)	% Of overheads
IPv4	1kb (1024 byte)	1064	AH	1108	44	4.14%
			ESP-CNF	1100	36	3.38%
			ESP-ATH	1108	44	4.14%
			ESP-CNF-ATH	1112	48	4.51%
IPv6		1084	AH	1148	64	5.90%
			ESP-CNF	1136	52	4.80%
			ESP-ATH	1148	64	5.90%
			ESP-CNF-ATH	1132	48	4.43%
IPv4	10kb (10240 byte)	10277	AH	10321	44	0.43%
			ESP-CNF	10316	39	0.38%
			ESP-ATH	10320	43	0.42%
			ESP-CNF-ATH	10328	51	0.49%
IPV6		10297	AH	10361	64	0.62%
			ESP-CNF	10352	55	0.53%
			ESP-ATH	10360	63	0.61%
			ESP-CNF-ATH	10364	67	0.65%
IPv4	100kb (102400 byte)	102440	AH	102484	44	0.04%
			ESP-CNF	102476	36	0.03%
			ESP-ATH	102484	44	0.04%
			ESP-CNF-ATH	102524	48	0.05%
IPv6		102460	AH	102524	64	0.06%
			ESP-CNF	102512	52	0.05%
			ESP-ATH	102524	64	0.06%
			ESP-CNF-ATH	102524	64	0.06%

R Table 7 Result: Analysis of variance (ANOVA) and means separation using Bonferrion method.

Df	Sum	Sq	Mean	Sq	F value Pr(>F)
PTCTNKB	3	640.5	213.50	2.471	0.0992
PLTNKB	2	0.0	0.00	0.000	1.0000
IPSTNKB	2	325.3	162.67	1.882	0.1844
Residuals	16	1382.7	86.42		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*'

Table 8.0. *(Large size data): The Overheads Imposed in Transport Modes by 1Mb, 10Mb and 100Mb IPSec Protected User Data.*

Protocol	Payload/file size in byte	Packet size with no IPSec in bytes	IPSec configuration set	IPSec Protected Packet size in bytes	Additional overhead in bytes	% Of overheads
IPv4	(1MB) 1048576bytes	1048616	AH	1048640	24	0.002%
			ESP-CNF	1048336	20	0.001%
			ESP-CNF	1048640	24	0.002%
			ESP-CNF-ATH	1048648	32	0.003%
IPv6		1048636	AH	1048660	24	0.002%
			ESP-CNF	1048656	20	0.001%
			ESP-ATH	1048660	24	0.002%
			ESP-CNF-ATH	1048668	32	0.003%
IPv4	(10MB) 10485760byte	10485800	AH	10485824	24	0.0002%
			ESP-CNF	10485820	20	0.0001%
			ESP-ATH	10485824	24	0.0002%
			ESP-CNF-ATH	10485832	32	0.0003%
IPV6		10485820	AH	10485845	25	0.0002%
			ESP-CNF	10485840	20	0.0001%
			ESP-ATH	10485844	24	0.0002%
			ESP-CNF-ATH	10485844	32	0.0003%
IPv4	(100MB) 10485760byte	104857640	AH	104857685	24	0.00002%
			ESP-CNF	104857660	20	0.00001%
			ESP-ATH	104857664	24	0.00001%
			ESP-CNF-ATH	104857672	32	0.00003%
IPv6		104857660	AH	104857684	24	0.00002%
			ESP-CNF	104857680	20	0.00001%
			ESP-ATH	104857684	24	0.00002%
			ESP-CNF-ATH	104857692	32	0.00003%

R Table 8 Result: Analysis of variance (ANOVA) and means separation using Bonferrion method.

Df	Sum	Sq	Mean	Sq	F value Pr(>F)
PTCSD	2	0.0	0.02	0.033	0.968
PLCSD	2	0.1	0.05	0.072	0.930
IPSSD	3	442.7	147.58	195.913	8.03e-13 ***
Residuals	16	12.1	0.75		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

LSD t Test for AOHSD

P value adjustment method: bonferroni

Mean Square Error: 0.753304

IPSSD, means and individual (95 %) CI

	AOHSD	std	r	LCL	UCL	Min	Max
AH	24.16667	0.4082483	6	23.41552	24.91782	24	25
ESP-ATH	24.00000	0.0000000	5	23.17716	24.82284	24	24
ESP-CNF	20.57143	1.5118579	7	19.87600	21.26686	20	24
ESP-CNF-ATH	32.00000	0.0000000	6	31.24885	32.75115	32	32

alpha: 0.05 ; Df Error: 16

Critical Value of t: 3.008334

Minimum difference changes for each comparison

Means with the same letter are not significantly different.

Groups, Treatments and means

a	ESP-CNF-ATH	32
b	AH	24.17
b	ESP-ATH	24
c	ESP-CNF	20.57

Table 9.0. *(Large size data): The Overheads Imposed in Tunnel Modes by 1Mb, 10Mb and 100Mb IPSec Protected User Data.*

Protocols	Payload/file size (byte)	Packet size with no IPSec (bytes)	IPSec configuration set	IPSec Protected Packet size (bytes)	Additional overheads (bytes)	% Of overheads
IPv4	(1MB) 1048576bytes	1048616	AH	1048660	44	0.004%
			ESP-CNF	1048648	32	0.003%
			ESP-CNF	1048660	44	0.004%
			ESP-CNF-ATH	1048664	48	0.004%
IPv6		1048636	AH	1048704	68	0.006%
			ESP-CNF	1048688	52	0.005%
			ESP-ATH	1048700	64	0.006%
			ESP-CNF-ATH	1048700	64	0.006%
IPv4	(10MB) 10485760bytes	10485800	AH	10485844	44	0.0004%
			ESP-CNF	10485836	36	0.0003%
			ESP-ATH	10485844	44	0.0004%
			ESP-CNF-ATH	10485848	48	0.0004%
IPV6		10485820	AH	10485888	68	0.0006%
			ESP-CNF	10485872	52	0.0005%
			ESP-ATH	10485884	64	0.0006%
			ESP-CNF-ATH	10485884	64	0.0006%
IPv4	(100MB) 10485760bytes	104857640	AH	104857684	44	0.00004%
			ESP-CNF	104857676	36	0.00003%
			ESP-ATH	104857684	44	0.00004%
			ESP-CNF-ATH	104857688	48	0.00004%
IPv6		104857660	AH	104857728	68	0.00006%
			ESP-CNF	104857712	52	0.00005%
			ESP-ATH	104857724	64	0.00006%
			ESP-CNF-ATH	104857724	64	0.00006%

R Table 9 Result: Analysis of variance (ANOVA) and means separation using Bonferrion method.

Df	Sum	Sq	Mean	Sq	F value Pr(>F)
ptct	3	2255.0	751.7	77.593	2.31e-09 ***
pls	2	27.7	13.8	1.428	0.271
ipst	3	545.3	181.8	18.765	2.45e-05 ***
Residuals	15	145.3	9.7		

Signif. Codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

LSD t Test for aoht

P value adjustment method: bonferroni

Mean Square Error: 9.68732

ptct, means and individual (95 %) CI

	aoht	std	r	LCL	UCL	Min	Max
IPv4	42.66667	5.465040	6	39.95834	45.37499	32	48
IPV4	42.66667	5.465040	6	39.95834	45.37499	36	48
IPv6	63.20000	6.572671	5	60.23318	66.16682	52	68
IPV6	61.14286	6.414270	7	58.63543	63.65028	52	68

alpha: 0.05 ; Df Error: 15

Critical Value of t: 3.036283

Minimum difference changes for each comparison

Means with the same letter are not significantly different.

Groups, Treatments and means

a	IPv6	63.2
a	IPV6	61.14
b	IPv4	42.67
b	IPV4	42.67

LSD t Test for aoht

P value adjustment method: bonferroni

Mean Square Error: 9.68732

ipst, means and individual (95 %) CI

	aoht	std	r	LCL	UCL	Min	Max
AH	56.00000	13.145341	6	53.29167	58.70833	44	68

ESP-ATH	56.00000	10.954451	5	53.03318	58.96682	44	64
ESP-CNF	43.42857	8.772251	7	40.92115	45.93600	32	52
ESP-CNF-ATH	56.00000	8.763561	6	53.29167	58.70833	48	64

alpha: 0.05 ; Df Error: 15
Critical Value of t: 3.036283
Minimum difference changes for each comparison
Means with the same letter are not significantly different.
Groups, Treatments and means

a	AH	56
a	ESP-ATH	56
a	ESP-CNF-ATH	56
b	ESP-CNF	43.43

5. Observations: Interpretation and Analysis of Result

Analysis in this research aim at observing the effect of additional space overhead incurred by internet protocols version 4 and 6 (IPv4, IPv6) as a result of selected IPsec configuration in relation to payload size in transport and tunnel mode of IPsec.

Table 4 shows the analysis of variance on the overhead imposed in transport mode by the said protocols (IPv4 and IPv6) configured with AH, ESP-CNF, ESP- ATH, ESP-CNF and CNF – ATH transmitting/transporting a payload size of 1 byte, 10 byte, and 100 byte.

The R result reveals that the choice of IPsec security configuration is significant at 0.1% level, but the selection of a particular protocol i.e. IPv4 or IPv6 or payload size is not significant.

However, when the IPsec security configuration are subjected to means separation using Bonferrion method, the result reveals that ESP-CNF has the least added overhead flowed by AH and ESP-ATH having the same performance and finally ESP-CNF-ATH having the highest. Similarly, when the selected IPsec security configurations are compared between IPv4 vs. IPv6 and the result tabulated in table 4, it indicated that from all the scenarios IPv4 introduced higher space overheads than IPv6. Hence, this is only when the IPSec transform set are applied on small user data (payload) that are not fragmented; because the actual IPSec headers size is equal on both protocols but if the files need to be fragmented as in the case of larger files, IPv6 incurred higher overhead. This can be reference to the manner at which IPSec header is implemented on the packets that require fragmentation. In IPv4 the IPSec header is applied only to the initial fragment, while in IPv6 the IPSec header is applied to all the fragmented portions of the packet. Therefore, the total cumulative result suppose IPv6 introduce higher overhead when large data are involved.

Table 5 shows the ANOVA of overhead imposed in tunnel mode by the same variables as in above. The result reveals that the protocols IPv4, IPv6 and IPsec security configuration set are both significant at 0.1% level while the payload size is not.

Further test of means separation using Bonferrion method for the protocols shows IPv4 has lower added overhead as compared to IPv6

Table 5 also shows the means separation using Bonferrion method of the IPsec security configuration set, with ESP-CNF having the lowest added overhead followed by ESP-ATH. AH and ESP-CNF-ATH incurred the same performance.

However, when the selected file size of 1 byte, 10 byte and 100 byte are factored in, an increase in the overheads is introduced. As can be seen in table 5, on one byte of data the least additional overhead incurred is 85% when ESP-CNF is used but AH, ESP-CNF and ESP-ATH introduce more than 100% additional space overhead when IPv4 is used. The situation is similar even with IPv6. The added overhead experienced higher increase compared to the one recorded in transport mode, this increase can be attributed to the fact that in tunnel mode the header and the payload are both encrypted and encapsulated in a newly created packet. Another observation made was that as the payload size increases the overhead begins to reduce slowly, as in the case of 100byte. With 100byte IPSec protected data in transport mode, the overhead caused by AH, ESP-CNF and ESP-ATH is 17% for all in IPv4 and 15%, 10%, 15% with IPv6 respectively. On the other hand ESP-CNF-ATH gives 20% and 17.5% with IPv4 and IPv6 respectively. The same falls is noticed in tunnel mode.

When the payload sizes are increased to 1 kilobyte, 10 kilobyte and 100 kilobyte, Table 6 shows the ANOVA for the overhead incurred in transport mode by IPsec. The result reveals that only IPsec is significant, this time at 5% level with ESP-CNF having the least addition compared to the rest when subjected to means separation using Bonferrion method. Table 6 also reveals that with 1 kilobyte, AH and ESP-CNF show 2.3% overhead in IPv4 and 0.23%, 0.22% with 10kb. With 100kb, only 0.06%, 0.02% and 0.02% are caused by AH, ESP-CNF and ESP-ATH. Meanwhile with IPV6 in play the same trend is observed as only 2.2% and 3.0% overheads are recorded on 1kb because of AH, ESP-CNF, ESP-ATH and ESP-CNF-ATH respectively. Similarly, with 10kb big falls in the overhead is noticed because AH and ESP-ATH-CNF gives 0.23%, ESP-CNF and ESP-ATH is 0.22%. Bigger reduction is observed when 100KB is used.

The result shows that ESP-CNF and ESP-ATH recorded 0.02 % overhead while AH and ESP-CNF-ATH recorded 0.06% and 0.03% respectively and as usual the overhead incurred in tunnel mode when the same file size is used is slightly greater due to the encapsulation. The means separation using Bonferrion method suggests that this is not a significant difference.

When tunnel mode is considered for the same configuration, Table 7 shows the ANOVA for the overhead imposed; the result reveals that all the three variables i.e. the IP protocols, IPsec configuration set and the payload size are not significant at 5%. Therefore, no further means separation test is required.

Lastly, when very large files are used the space overhead reduced drastically. Table 9 shows the ANOVA for the overhead imposed in transport while transmitting packets of sizes 1mb, 10mb and 100mb. The result reveals that only IPsec is significant at 0.1%.

The means separation using Bonferrion method for the IPsec reveals that ESP-CNF has the least overhead followed by ESP-ATH, AH and lastly ESP-CNF-ATH. This was the observation made when 1mb, 10mb and 100mb are used as the payload data. With IPv4 and in IPSec transport mode 1mb IPSec protected data caused only 0.002%, 0.001%, 0.002% and 0.003% by AH, ESP-CNF, ESP-ATH, ESP-CNF-ATH. The same happened in IPv6 according to the record in table 8, but it is important to note that for IPv6, since the packet is very large the packet must be fragmented and IPSec header is attached to each fragment. After taking the overall summation the total space overhead will be equal to the (no of fragments * the overhead caused by single fragment) (i.e. as in the case in the record, since it represent the overhead caused by single fragment) this is the reason that explain why IPv6 suffer more IPSec space overhead when large files are involved, since in IPv4 the IPSec header is applied to first fragment only while in IPv6 the header is applied to all.

For the ANOVA of overhead imposed in tunnel mode by the same payload size. The results indicated that protocols and IPsec are significant at 0.1% level. The means separation using Bonferrion method for the protocols shows IPv4 has the lower overhead as compared to IPv6. While the mean separation of IPsec configuration set, reveals ESP-CNF with the least overhead and the remaining configuration having the same performance matrix

Furthermore, looking down the table, it can be noticed that 100mb user data caused very little space overhead especially in IPv4 setup because AH, ESP-CNF-ESP-ATH and ESP-CNF-ATH add only 0.00002%, 0.00001%, 0.00001% and 0.00003% in transport mode. It is little higher in tunnel mode because in IPSec tunnel mode, the encryption algorithm is applied to the entire IP packet, both the payload and the IP header inclusive, and then encapsulates it in a new IP header, this means that no portion of the IP packet is exempted from IPSec protection during transmission unlike in transport mode where only the IP packet payload is encrypted during transmission.

6. Conclusion

This paper demonstrated how IPsec headers under different protocol configurations setup introduced additional processing and space overhead with respect to different file size on two different Internet protocols; IPv4 and IPv6. The study indicated that the cost of IPsec added overhead was smaller when smaller packet sizes were involved for both protocols as compared to larger packet sizes that are IPsec protected with the same configuration as the smaller packet. The only exception was in the cases whereby the packet is very large that it has to be fragmented. In such case IPv6 experienced higher overhead than IPv4. This happened due to the fact that the manner at which IPv6 handles fragmented packet when IPsec is involved was completely different with the way IPv4 tackles its. IPv6 applied IPsec header to all fragmented portion of the packet while IPv4 applied it to the very initial fragment only.

References

[1] Cheng Min (2011) Research On network Security Based on IPv6 Architecture. 2011 international Conference on Electronics and Optoelectrnics (ICEOE 2011) (PP 1-3) Institute of Electrical Electronics Engineers (IEEE Xplore Digital Library).

[2] Christos Xenakis, Nikolaos Laoutaris, Lazaros Merakos, Ioannis Stavrakakis (2006) A generic characteristics of the overheads imposed by IPSec and associated cryptographic algorithms. ScienceDirect computer networks 50 (2006) 3225-3241

[3] George C. Hadjichristofi Nathaniel J. Davis, IVScott F. Midkiff (2003) IPSec Overhead in Wireline and Wireless Networks for Web and Email Applications.(PP 1-5) IEEE Institute of Electrical Electronics Engineers (IEEEXplore Digital Library).

[4] Eastlake 3rd D. Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH) IETF RFC 4305 (December 2005)

[5] J. C. Lin, C. T. Chang and W. T. Chung, (2003) "Design, Implementation and Performance Evaluation of IP-VPN", In Proc.of AINA 2003, pp. 206 - 209,.

[6] J. Klaue and A. Hess, "On the Impact of IPSec on Interactive Communications" (2005) In Proc. of IPDPS 2005, 8 pp.,

[7] Meenakshi S. P, Raghavan S.V(2006) Impact of IPSec Overhead on Web Application Servers.(PP1-6) Institute of Electrical Electronics Engineers (IEEE Xplore Digital Library).

[8] Mujinga M.H, Muyingi G.S.V.R, Krishna R (2006) IPSec Overhead Analysis in Dual Stack IPv4/IPv6 Transition Mechanisms. (PP 1-6) Institute of Electrical Electronics Engineers (IEEE Xplore Digital Library).

[9] O. Elkeelany, M.M.Matalgah, K.P. Sheikh, G. Chaudhry, D. Medhi and J. Qaddour, (2002)"Performance Analysis of IPSec Protocol: Encryption and Authentication", In Proc. of IEEE Communication Conference ICC2002, Vol. 2, pp. 1164-1168.

[10] Seiji ARIGA, Masaki MINAMI, Hiroshi ESAKI and Jun MURAI, (2000) "Performance Evaluation of Data Transmission Using IPSec over IPv6 Networks", In Proc. of the 10th Annual Internet Society Conference (INET 2000), Yokohama, Japan.

[11] Todd Lammle (2010) Cisco Certified Network Associate. Wiley Publishing. Inc USA

[12] Wenhong Liu, Zhen Jiang, Hongke Zhang (2006). A Secure Mobile-IPv6 Network Model. ICWMMN 2006 Proceedings (PP 1-4)