RESEARCH ARTICLE OPEN ACCESS

Performance analysis of mail clients with RSA and Digital signature using SNORT

Mr. K Sreerama Murthy [1], Dr. S P.Setty [2] , Dr. G. Samuel Vara Pasad Raju [3]
Research Scholar [1],
Department of CS&SE [2], Andhra University [3]
Andhra University, Visakhapatnam
AP- India

ABSTRACT
With the increased dependence of organizations on technological solutions, the cyber threats became major concern for businesses to run. SNORT is one of the open source tool which is meant for detecting suspicious activities. It performance actions such as blocking the user or source IP address from accessing the network. SNORT can be configured as an Intrusion Prevention system (IPS) for monitoring and prevention of security attacks on networks. We applied encryption for text files by using cryptographic algorithms like Digital signature and RSA. We found that Snort is effective for compressed data for these algorithms. We observed that as the size of the file increases, the run time is constant for compressed data whereas in plain text, it varied drastically.

Keywords :— IDS, SNORT, Cryptography.

I. INTRODUCTION

An I**ntrusion Detection System** (**IDS**) is a device or software application that monitors network or system activities for malicious activities or policy violations and produces reports to a management station. Intrusion Detection and Prevention Systems (IDPS) are primarily focused on identifying possible incidents, logging information about them, and reporting attempts.

Free Intrusion Detection Systems:

- ACARM
- OSSEC HIDS
- AIDE
- Bro NIDS
- Prelude Hybrid DS
- Samhain
- Snort
- Suricata

NIDS:

Network Intrusion Detection Systems are placed at a strategic point or points within the network to monitor traffic to and from all devices on the network. Ideally one would scan all inbound and outbound traffic however doing so might create a bottleneck that would impair the overall speed of the network.

HIDS:

Host Intrusion Detection Systems are run on individual hosts or devices on the network. A HIDS monitors the inbound and outbound packets from the device only and will alert the user or administrator of suspicious activity.

Snort is an open source network intrusion detection system capable of performing real-time traffic analysis and packet logging on IP networks.

II. PROBLEM STATEMENT

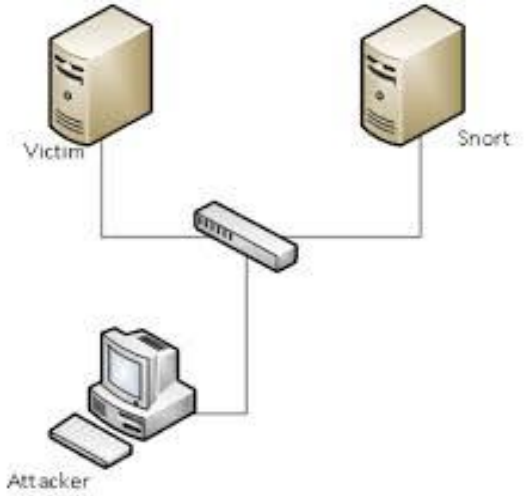
Snort is a multimode packet analysis tool. In the Snort, we mainly concentrate on sniffer mode. In Sniffer mode, Snort will read the network traffic and print them to the screen. Snort is considered a superior Network Intrusion Detection System when compared to the most commercial systems. In my project, we measured the performance of different mail clients by using Snort. In the simulation study, we selected three mail clients (Gmail, Yahoo, Hotmail). By varying the text sizes from 50 kb to 2mb for all the three mail clients, we found that runtime is less when the mail client Hotmail is used. Again Snort is applied by using cryptographic algorithms to encrypt plain text using Digital signature technique and RSA. From simulation results, we found that for compression data the impact of Snort is very less.

III. SNORT OVERVIEW

Snort is a free and open source Network Intrusion Detection System(NIDS) and Network Intrusion Prevention System (NIPS) and created by Martin Roesch in 1998. Snort's open source network-based intrusion detection system (NIDS) has the ability to perform real-time traffic analysis and packet logging on Internet Protocol (IP) networks. Snort performs protocol analysis, content searching, and content matching. The program can also be used to detect probes or attacks. Snort can be configured in three main modes: sniffer, packet logger, and network intrusion detection. In sniffer mode, the program will read network packets and display them on the

console. In packet logger mode, the program will log packets to the disk. In intrusion detection mode, the program will monitor network traffic and analyze it against a rule set defined by the user.

Sniffer mode is used for accessing the performance where as logger mode is used for comparing the performance.



Snort Modes:
Snort can run in three different modes:

A. Sniffer mode

Sniffer mode simply reads the packets from the network and displays them in a continuous stream on the console.
Options:
snort –v
Prints TCP/IP header onto screen (also for UDP / ICMP).
snort –vd
Prints the application data too.
./snort –vde
Prints the data link layer contents as well.

B. Packet Logger mode

Packet logger mode logs the packets to the disk.
Options:
snort –dev –l ./log
Logs the packets to the directory specified.

./snort –l ./log –b
Binary log, binary file may be read back using –r switch

C. IDS mode:

Snort provides near real-time intrusion detection capability with IDS mode.

Options:

snort -c /usr/local/share/snort_rules/rules/snort.conf

IV. MAIL CLIENTS

An email client, email reader, or more formally mail user agent (MUA), is a computer program used to access and manage a user's email.

The term can refer to any system capable of accessing the user's email mailbox, regardless of it being a mail user agent, a relaying server, or a human typing on a terminal. In addition, a web application that provides message management, composition, and reception functions is sometimes also considered an email client, but more commonly referred to as webmail.

Popular web-based email clients:

Gmail, Yahoo! Mail, mail.com, Lycos mail, and Hotmail

A. Gmail

Gmail is a free, advertising-supported email service provided by Google. Users may access Gmail as secure webmail, as well as via POP3 or IMAP4 protocols. Gmail initially started as an invitation-only beta release on April 1, 2004 and it became available to the general public on February 7, 2007, though still in beta status at that time. The service was upgraded from beta status on July 7, 2009, along with the rest of the Google Apps suite.

B. Yahoo

Yahoo! Inc. is an American multinational internet corporation headquartered in Sunnyvale, California. It is widely known for its web portal, search engine Yahoo! Search, and related services, including Yahoo! Directory, Yahoo! Mail, Yahoo! News, Yahoo! Finance, Yahoo! Groups, Yahoo! Answers, advertising, online mapping, video sharing, fantasy sports and its social media website. It is one of the most popular sites in the United States.

C. Hotmail:

Outlook.com (previously MSN Hotmail, Windows Live Hotmail and Hotmail) is a free web-based email service operated by Microsoft. Hotmail was one of the first web-based email services, it was founded by Sabeer Bhatia and Jack Smith and launched in July 1996 as "HoTMaiL". It was acquired by Microsoft in 1997 for an estimated $400 million, and shortly after, it was rebranded as "MSN Hotmail". The last version was released in 2011. In February 2013, it was renamed to Outlook.com as part of the rebranding of the Windows Live suite of products.

V. ENCRYPTION

In cryptography, encryption is the process of encoding messages (or information) in such a way that eavesdroppers or hackers cannot read it, but that authorized parties can. In an encryption scheme, the message or information (referred to as plain text) is encrypted using an encryption algorithm, turning it into an unreadable cipher text (ibid.). This is usually done with the use of an encryption key, which specifies how the message is to be encoded. Any adversary that can see the

cipher text should not be able to determine anything about the original message. An authorized party, however, is able to decode the cipher text using a decryption algorithm that usually requires a secret decryption key that adversaries do not have access to. For technical reasons, an encryption scheme usually needs a key-generation algorithm to randomly produce keys.

There are two basic types of encryption schemes: Symmetric-key and public-key encryption. In symmetric-key schemes, the encryption and decryption keys are the same. Thus communicating parties must agree on a secret key before they wish to communicate. In public-key schemes, the encryption key is published for anyone to use and encrypt messages however only the receiving party has access to the decryption key and is capable of reading the encrypted messages. Public-key encryption is a relatively recent invention: historically, all encryption schemes have been symmetric-key (also called private-key) schemes

A. Digital Signature(DS) :

A *digital signature* is a mathematical scheme for demonstrating the authenticity of a digital message or document. A valid digital signature gives a recipient reason to believe that the message was created by a known sender, such that the sender cannot deny having sent the message (authentication and non-repudiation) and that the message was not altered in transit (integrity). Digital signatures are commonly used for software distribution, financial transactions, and in other cases where it is important to detect forgery or tampering.

A digital signature scheme typically consists of three algorithms:

- A *key generation* algorithm that selects a *private key* uniformly at random from a set of possible private keys. The algorithm outputs the private key and a corresponding *public key*.
- A *signing* algorithm that given a message and a private key, produces a signature.
- A *signature verifying* algorithm that, given a message, public key and a signature, either accepts or rejects the message's claim to authenticity.

Two main properties are required. First, a signature generated from a fixed message and fixed private key should verify the authenticity of that message by using the corresponding public key. Secondly, it should be computationally infeasible to generate a valid signature for a party without knowing that party's private key

Uses: There are several reasons to sign such a hash (or message digest) instead of the whole document.

For efficiency: The signature will be much shorter and thus save time since hashing is generally much faster than signing in practice.

For compatibility: Messages are typically bit strings, but some signature schemes operate on other domains (such as, in the case of RSA, numbers modulo a composite number *N*). A hash function can be used to convert an arbitrary input into the proper format.

For integrity: Without the hash function, the text "to be signed" may have to be split (separated) in blocks small enough for the signature scheme to act on them directly. However, the receiver of the signed blocks is not able to recognize if all the blocks are present and in the appropriate order.

B. RSA:

RSA is an algorithm for public-key cryptography that is based on the presumed difficulty of factoring large integers, the factoring problem. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described the algorithm in 1977. A user of RSA creates and then publishes the product of two large prime numbers, along with an auxiliary value, as their public key.

The prime factors must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, if the public key is large enough, only someone with knowledge of the prime factors can feasibly decode the message. Whether breaking RSA encryption is as hard as factoring is an open question known as the RSA problem.

The RSA algorithm involves three steps:

key generation, encryption and decryption.

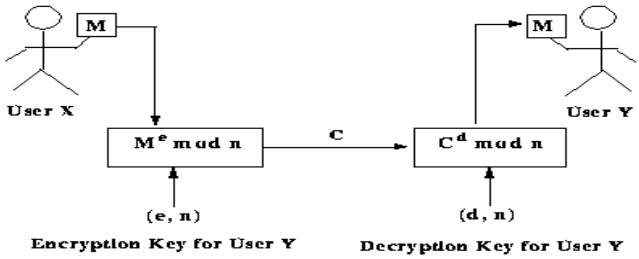


Fig: Encryption process

VI. EXPERIMENTAL RESULTS

1) SNORT WITH DIGITAL SIGNATURE TECHNIQUE

We have carried out experiment to analyse the data produced by snort during sending files(both plain text and encrypted text individually) of various sizes (50kb,100kb,500kb,1mb,2mb) in Gmail, Yahoo and Hotmail mail clients.

Initially the un-encrypted files are sent and the data produced by snort is compared to the data of encrypted files produced by snort.

A. Applying Digital Signature Technique in Gmail with Snort

Following is the table of analysis of data produced by snort when files are sent through Gmail. The first row is the data of plain text sent when snort is running. The second row is the data of corresponding encrypted text using digital signature.

1) *Total Packets(Received)-Digital Signature-Gmail*

TABLE 1:

TOTAL PACKETS-DIGITAL SIGNATURE-GMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain Text	225	295	776	1870	2835
Encrypted	190	203	182	104	203

Graph:

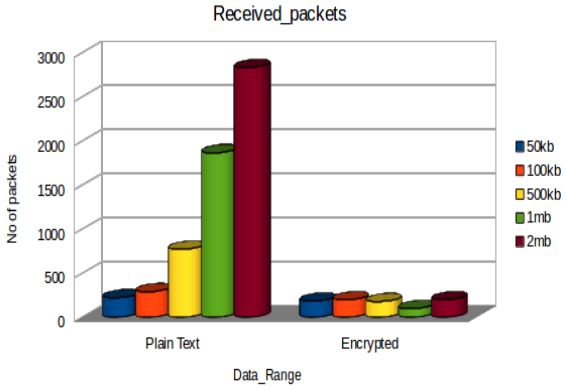


Fig:1 Total Received Packets –Digital Signature-Gmail

Total packets are the total number of packets received by the snort from the Ethernet.

2) *Analysed Packets:*

Analysed packets are the packets analysed by snort during runtime. Snort does not analyse all the packets received by the Ethernet, it drops some of the packets that are needed to be buffered for processing.

Following is the table of comparison of analysed packets in plain text and encrypted text of files of various sizes in Gmail. Greater the analysed packets, greater is the performance of the encrypted algorithm.

TABLE 2:
ANALYZED PACKETS-DS-GMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain Text	225	295	776	1866	2612
Encrypted	190	203	180	98	197

Graph :

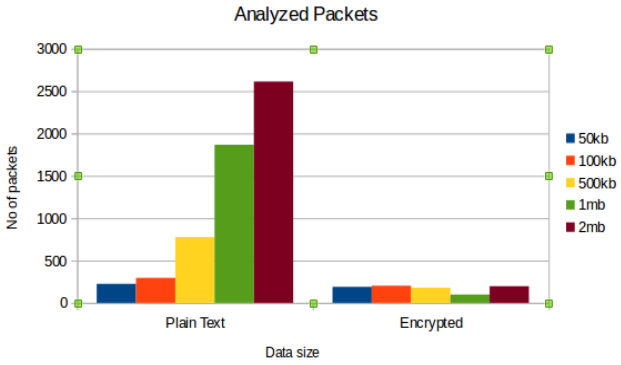


Fig:2 Analysed packets-DS-Gmail

In the above graph the number of analysed packets is more for plain text when compared to encrypted text.

3) *Run Time-Digital Signature-Gmail*

Run time is the total amount of time taken by Snort for analysing the packets received from the Ethernet. If the run time taken during sending of plain text is higher when compared to encrypted text, then encrypting the text saves the processing time and memory.

Following is the table of comparison of run time during sending plain text files and encrypted files.

TABLE 3:
RUNTIME –DIGITAL SIGNATURE –GMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain text	24.238	25.167	35.827	56.544	65.231
Encrypted	23.516	22.269	23.116	22.23	23.23

Graph:

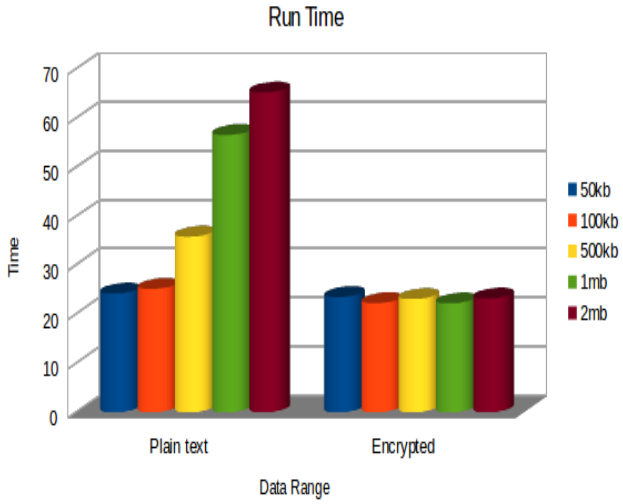


Fig3: Runtime –Digital Signature –Gmail

In the above graph the processing time taken by Snort is higher for plain text when compared to encrypted text.

B. *Applying Digital Signature technique in Yahoo with Snort*

Following is the table of analysis of data produced by snort when files are sent through Yahoo. The first row is the data of plain text sent when snort is running. The second row is the data of corresponding encrypted text using digital signature.

1) *Total Packets-DS-Yahoo*

Total packets are the total number of packets received by the snort from the Ethernet.

TABLE4:
TOTAL PACKETS -DIGITAL SIGNATURE-YAHOO

Size	50kb	100kb	500kb	1mb	2mb
Plain text	400	598	1041	2263	3241
Encrypted	229	227	286	285	315

Graph:

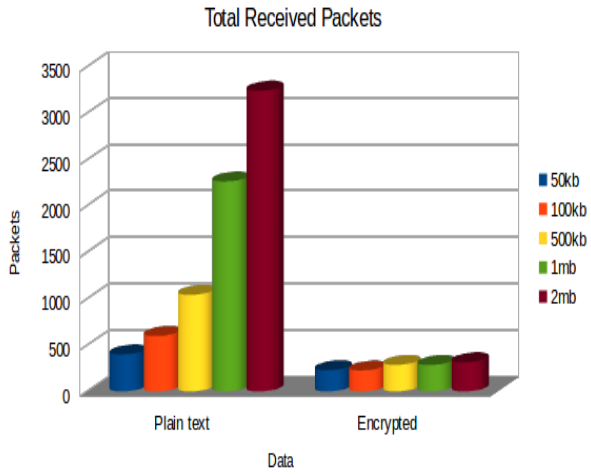


Fig4: Total packets -Digital Signature-Yahoo

In the above graph the total number of packets in encrypted text is less when compared to the number of packets in plain text. Hence if we encrypt the data less memory is consumed and run time is reduced,which inturn increases the performance of the system

2) *Analysed Packets-DS-Yahoo:*

Following is the table of comparison of analysed packets in plain text and encrypted text of files of various sizes in Yahoo.

TABLE5:
ANALYSED PACKETS -DIGITAL SIGNATURE-YAHOO

Size	50kb	100kb	500kb	1mb	2mb
Plain text	397	598	1041	2257	3223
Encrypted	229	227	278	284	312

Graph:

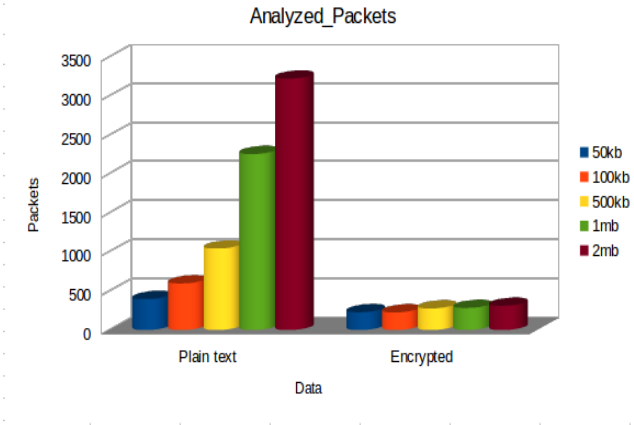


Fig5: Analysed packets -Digital Signature-Yahoo

In the above graph, the number of analysed packets increased according to the file sizes, but in encrypted text, the number of analysed packets remained consistent irrespective of the file size.

3) *Run Time for packet analysis using SNORT*

Run time is the total amount of time taken by Snort for analysing the packets received from the Ethernet. If the run time taken during sending of plain text is higher when compared to encrypted text, then encrypting the text saves the processing time and memory.

Following is the table of comparison of run time during sending plain text files and encrypted files.

TABLE 6:
RUNTIME- DIGITAL SIGNATURE- YAHOO MAIL.

Size	50kb	100kb	500kb	1mb	2mb
Plain text	29.568	31.95	42.153	56.66	71.356
Encrypted	20.58	21.61	22.83	26.7	19.524

Graph:

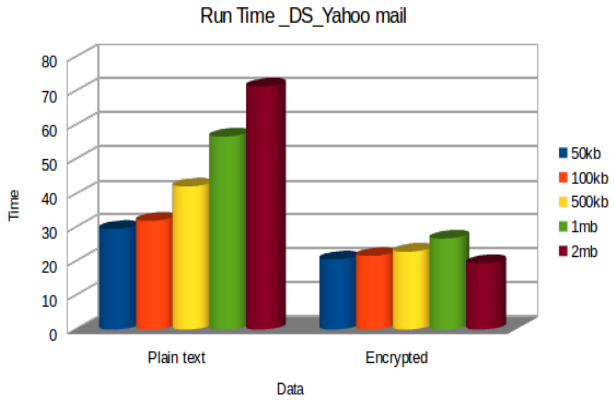


Fig: Runtime- Digital Signature- Yahoo Mail.

In the above graph, the processing time taken by Snort is higher for plain text when compared to encrypted text.

C. *Applying Digital Signature technique in Hotmail with Snort*

Following is the table of analysis of data produced by snort when files are sent through Hotmail. The first row is the data of plain text sent when snort is running. The second row is the data of corresponding encrypted text using digital signature.

1) *Total Packets DS-Hotmail:*

Total packets are the total number of packets received by the snort from the Ethernet.

Table7:

TOTAL PACKETS DS –HOTMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain text	190	258	807	1788	2821
Encrypted	79	133	77	46	86

Graph:

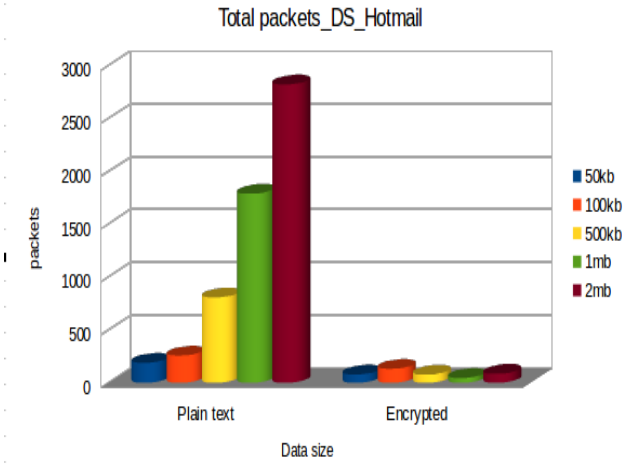


Fig7: Total packets DS –Hotmail

In the above graph, the total number of packets processed by snort in case of encrypted text is lower when compared to plain text. Hence encrypted text occupies less memory and consumes less time.

2) *Analysed Packets for DS-Hotmail*

Following is the table of comparison of analysed packets in plain text and encrypted text of files of various sizes in Hotmail

TABLE8:

ANALYSED PACKETS -DIGITAL SIGNATURE-HOTMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain text	190	258	796	1788	2475
Encrypted	86	44	77	44	86

Graph:

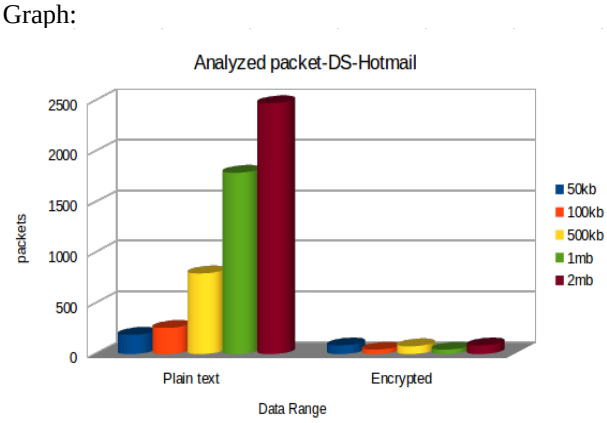


Fig8: Analysed Packets -Digital Signature-Hotmail

In the above graph the numbers of analysed packets are higher in plain text when compared to encrypted text.

3) *Run time:*

Run time is the total amount of time taken by Snort for analysing the packets received from the Ethernet. If the run time taken during sending of plain text is higher when compared to encrypted text, then encrypting the text saves the processing time and memory. Following is the table of comparison of run time during sending plain text files and encrypted files.

TABLE 9:

RUN TIME -DS-HOTMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain text	28.483	22.822	31.156	55.52	60.925
Encrypted text	15.943	20.68	18.18	17.846	16.224

Graph:

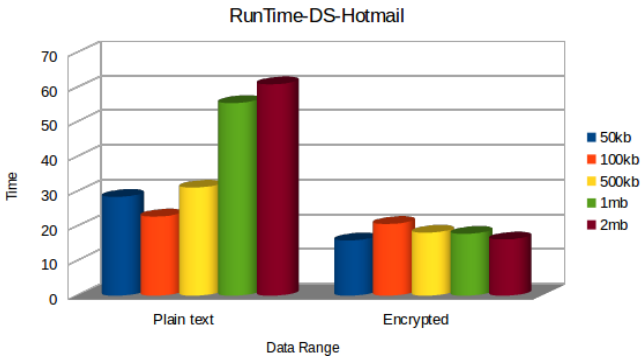


Fig9: Run Time -DS-Hotmail

In the above graph, the run time taken by Snort for plain text is more when compared to encrypted text. If runtime is less, the efficiency of the system is increased.

From the above graph the run time taken by encrypted text is more than plain text for 100kb size, which indicates that if we are using file size of 100 kb then sending the plain text saves us memory and time.

2) SNORT WITH RSA TECHNIQUE

We have carried out experiment to analyse the data produced by snort during sending files(both plain text and encrypted text individually) of various sizes (50kb ,100kb,500kb,1mb,2mb) in Gmail, Yahoo and Hotmail mail clients.

Initially the un-encrypted files are sent and the data produced by snort is compared to the data of encrypted files produced by snort.

A. Applying RSA Technique in Gmail with Snort

Following is the table of analysis of data produced by snort when files are sent through Gmail. The first row is the data of plain text sent when snort is running. The second row is the data of corresponding encrypted text using RSA algorithm.

1) Total Packets:

Total packets are the total number of packets received by the snort from the Ethernet.

TABLE 10:

TOTAL PACKETS-RSA-GMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain text	225	295	776	1870	2835
Encrypted	166	155	149	188	144

Graph:

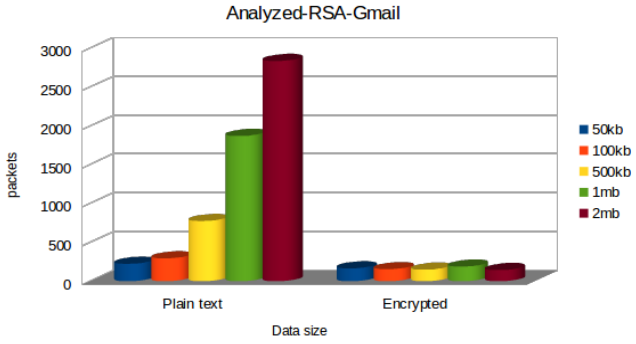


Fig: 10: Total packets-RSA-Gmail

In the above graph the total number of packets analysed by snort using RSA algorithm is consistent when compared to number of packets in plain text.

2) Analysed Packets:

Analysed packets are the packets analysed by snort during runtime. Snort does not analyse all the packets received by the Ethernet, it drops some of the packets that are needed to be buffered for processing.

Following is the table of comparison of analysed packets in plain text and encrypted text of files of various sizes in Gmail. Greater the analysed packets, greater is the performance of the encrypted algorithm.

TABLE11:

ANALYSED PACKETS-RSA-GMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain text	225	295	776	1866	2612
Encrypted	119	113	114	222	252

Graph:

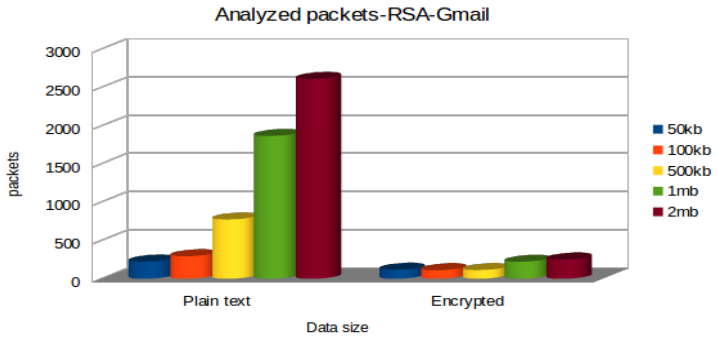


Fig11: Analysed packets-RSA-Gmail

In the above graph the number of packets analysed by Snort for encrypted is less when compared to plain text.

3) Run Time:

Run time is the total amount of time taken by Snort for analysing the packets received from the Ethernet. If the run time taken during sending of plain text is higher when compared to encrypted text, then encrypting the text saves the processing time and memory.

Following is the table of comparison of run time during sending plain text files and encrypted files.

TABLE 12:

RUN TIME -RSA-GMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain text	24.238	25.167	35.827	56.544	65.231
Encrypted	21.11	24.1	24.276	31.619	26.29

Graph:

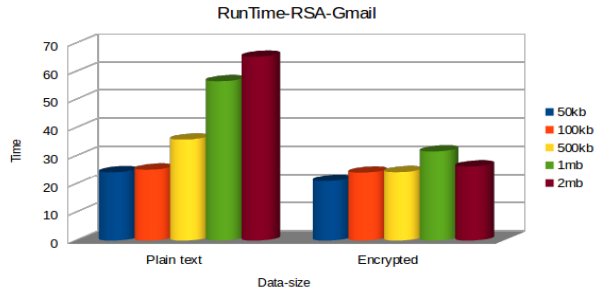


Fig12: Run Time -RSA-Gmail

In the above graph, the total run time taken by Snort for encrypted text is less when compared to plain text.

B. Applying RSA in Yahoo with Snort

Following is the table of analysis of data produced by snort when files are sent through Yahoo. The first row is the data of plain text sent when snort is running. The second row is the data of corresponding encrypted text using RSA encryption technique.

1) Total Packets:

Total packets are the total number of packets received by the snort from the Ethernet.

TABLE 13:

TOTAL PACKETS-RSA-YAHOO

Size	50kb	100kb	500kb	1mb	2mb
Plain text	400	598	1041	2263	3241
Encrypted	372	451	445	168	368

Graph:

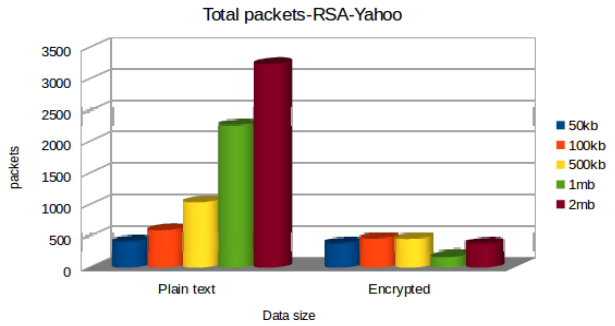


Fig13: Total packets-RSA-Yahoo

In the above graph, the total number of packets by snort for encrypted text is less when compared to plain text.

2) Analysed Packets:

Following is the table of comparison of analysed packets in plain text and encrypted text of files of various sizes in Yahoo.

TABLE 14

ANALYSED PACKETS -RSA-YAHOO

Size	50kb	100kb	500kb	1mb	2mb
Plain text	397	598	1041	2257	3223
Encrypted	364	449	421	154	361

Graph:

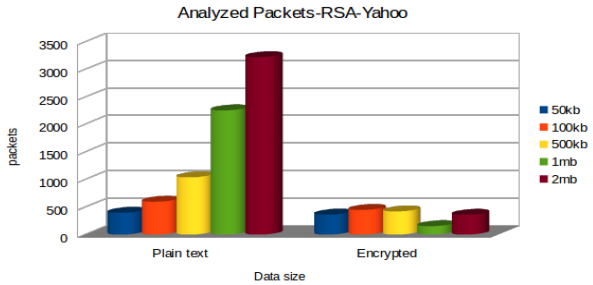


Fig14: Analysed Packets -RSA-Yahoo

In the above graph the number of analysed packets by Snort remained same in both plain text and encrypted text for 50 kb file size, but for other file sizes the number variation is higher.

3) Run Time:

Run time is the total amount of time taken by Snort for analysing the packets received from the Ethernet. If the run time taken during sending of plain text is higher when compared to encrypted text, then encrypting the text saves the processing time and memory.

Following is the table of comparison of run time during sending plain text files and encrypted files.

TABLE15:

RUN TIME-RSA-YAHOO

Size	50kb	100kb	500kb	1mb	2mb
Plain text	29.568	31.95	42.153	56.66	71.356
Encrypted	25.93	25.726	22.8	29.987	24.57

Graph:

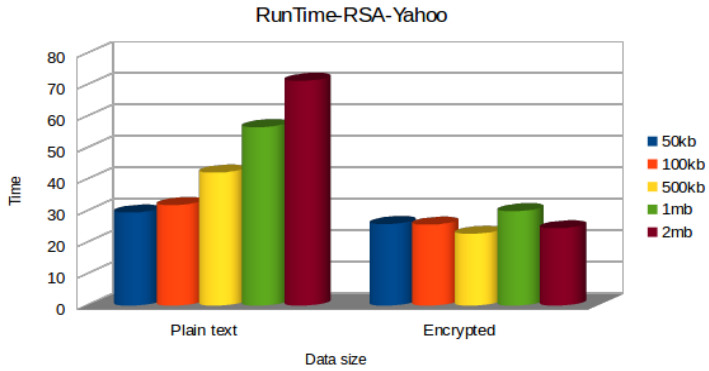


Fig: 15: Run Time-RSA-Yahoo.

C. Applying RSA technique in Hotmail with Snort

Following is the table of analysis of data produced by snort when files are sent through Hotmail. The first row is the data of plain text sent when snort is running. The second row is the data of corresponding encrypted text using RSA.

1) Total Packets:

Total packets are the total number of packets received by the snort from the Ethernet.

TABLE16:

TOTAL PACKETS-RSA-HOTMAIL.

Size	50kb	100kb	500kb	1mb	2mb
Plain text	190	258	807	1788	2821
Encrypted	132	115	94	169	144

Graph:

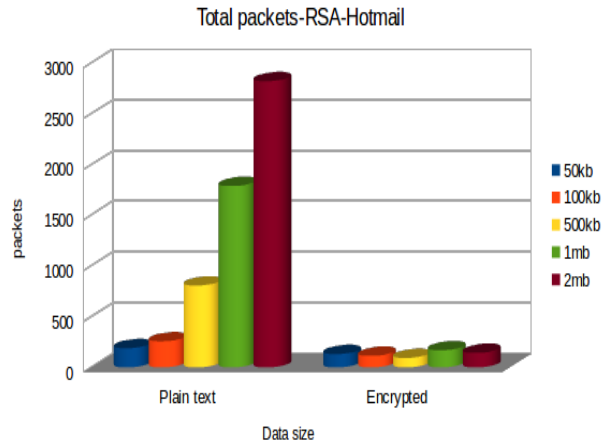


Fig16: Total packets-RSA-Hotmail.

In the above graph the number of packets received by Snort for encrypted text remained consistent irrespective of the file size.

2) *Analysed Packets:*

Following is the table of comparison of analysed packets in plain text and encrypted text of files of various sizes in Hotmail

TABLE17:

ANALYSED PACKETS -RSA-HOTMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain text	190	258	796	1788	2475
Encrypted	132	115	94	160	144

Graph:

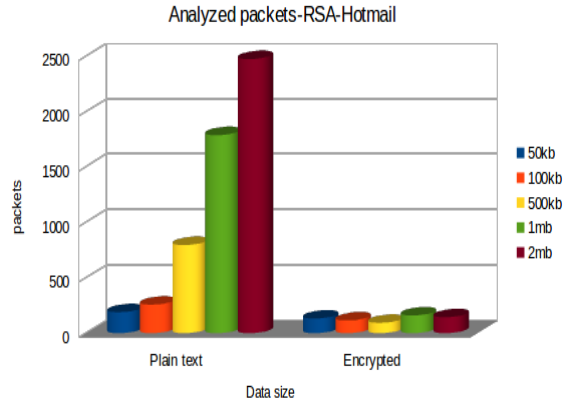


Fig 17: Analysed Packets -RSA-Hotmail

In the above graph, the number of packets analysed by snort for encrypted text remained consistent irrespective of various sizes whereas it varied drastically for plain text.

3) *Run time:*

Run time is the total amount of time taken by Snort for analysing the packets received from the Ethernet. If the run time taken during sending of plain text is higher when compared to encrypted text, then encrypting the text saves the processing time and memory.

Following is the table of comparison of run time during sending plain text files and encrypted files.

TABLE 18:

RUN TIME -RSA-HOTMAIL

Size	50kb	100kb	500kb	1mb	2mb
Plain text	28.483	22.822	31.156	55.52	60.925
Encrypted	19.923	18.357	18.77	22.932	20.343

Graph:

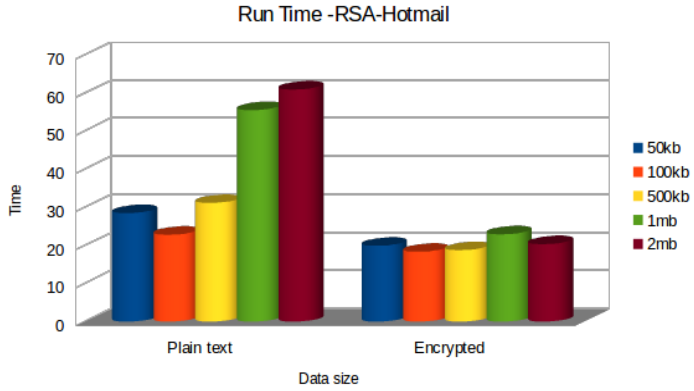


Fig18: Run Time -RSA-Hotmail.

In the above graph, the run time taken by snort is less for encrypted text when compared to plain text.

VII. CONCLUSION

From this paper, we found that snort is effective for analysing the performance of various mail clients. We observed the performance of all mail clients by sending ranging from 50 kb to 2 MB through all the three mail clients. We observed that as a size of file increases the runtime is varying drastically in case of plain text. Then the same plain text files are encrypted using cryptographic algorithms like Digital Signature and RSA, and were sent through the same three mail clients. From simulation scenarios, we observed that as the size of the file increase, the runtime is constant for compressed data.

VIII. FUTURE ENHANCEMENT

In this paper, we have experimented with RSA and Digital Signature algorithms. We can extend our experiments with other algorithms for encryption. There is still scope to find out

the best suitable mail clients for transferring of data based on the size.

REFERENCES

[1] G. Varghese, "Network Algorithmic: An Interdisciplinary Approach to Designing Fast Networked Devices", San Francisco, CA: Morgan Kaufmann, 2005.

[2] J. Cleary, S. Donnelly, I. Graham, "Design Principles for Accurate Passive Measurement," in Proc. PAM 2000 Passive and Active Measurement Workshop (Apr. 2000).

[3] A. Dabir, A. Matrawy, "Bottleneck Analysis of Traffic Monitoring Using Wireshark", 4th International Conference on Innovations in Information Technology, 2007, IEEE Innovations '07, 18-20 Nov. 2007.

[4] S. Ansari, Rajeev S.G. and Chandrasekhar H.S, "Packet Sniffing: A brief Introduction", IEEE Potentials, Dec 2002- Jan 2003, Volume:21, Issue:5, pp:17 – 19

[5] Daiji Sanai, "Detection of Promiscuous Nodes Using ARP Packet", http://www.securityfriday.com/

[6] Ryan Spangler , Packet Sniffer Detection with AntiSniff, University of Wisconsin – Whitewater, Department of Computer and Network Administration, May 2003

[7] Zouheir Trabelsi, Hamza Rahmani, Kamel Kaouech, Mounir Frikha, "Malicious Sniffing System Detection Platform", Proceedings of the 2004 International Symposium on Applications and the Internet (SAINT'04), IEEE Computer Society.

[8] Hornig, C., "A Standard for the Transmission of IP Data grams over Ethernet Networks", RFC-894, Symbolic Cambridge Research Center,April 1984.

[9] Lin Tan, Timothy Sherwood. A High Throughput String Matching Architecture for Intrusion Detection and Prevention, Proceedings of the 32 nd Annual International Symposium on Computer Architecture (ISCA 2005).

[10] S. Mrdovic, E. Zajko. Secured Intrusion Detection System Infrastructure, University of Sarajevo/Faculty of Electrical Engineering, Sarajevo, Bosnia and Herzegovina (ICAT 2005).

[11] Yeubin Bai, Hidetsune Kobayashi. Intrusion Detection Systems: technology and Development, 17 th International Conference of Advanced Information Networking and Applications, (AINA 2003).

[12] Sang-Jun Han and Sung-Bae Cho. Combining Multiple Host-Based Detectors Using Decision Tree, Australian Joint Artificial Intelligence Conference, (AUSAI 2003

[13] Ramaprabhu Janakiraman, Marcel Waldvogel, Qi Zhang. Indra: A peer-to-peer approach to network intrusion detection and prevention, Enabling Technologies: Infrastructure for Collaborative Enterprises, WET ICE 2003

[14] M. Laureano, C. Maziero1, E. Jamhour. Protecting Host-Based Intrusion Detectors through Virtual Machines, The International Journal of Computer and Telecommunications Networking (2007).

[15] Matt Carlson and Andrew Scharlott. Intrusion detection and prevention systems, (2006).