TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol. 16, No. 1, October 2015, pp. 154 ~ 166
DOI: 10.11591/telkomnika.v16i1.8749

Contribution to the Security of the Information System

A. Ben Charke[1], M. Chabi[2], M. Fakir*[3]
[1,2]LMA Laboratory, Faculty of Science and Technology, Beni Mellal
B.P: 523 Beni - Mellal Morocco
[3]TIAD Laboratoiry, Faculty of Science and Technology, Beni Mellal
B.P: 523 Beni - Mellal Morocco
*Corresponding author, e-mail: a.bencharke@usms.ma[1], m.chabi@usms.ma[2], m.fakir@usms.ma[3]

Abstract

Security of information systems has become a critical problem of companies. In this paper, the principles of security and the description of some attacks that threatening the information system are given. After Techniques of cryptography, digital signature to ensure the confidentiality, integrity and authentication of data, are described. Some security protocol such as Secure Shell (SSH), Secure Socket Layer (SSL), Internet Protocol SECure (IPSEC), in order to ensure the security of connection resources, are described. Intrusion detection is implemented using free IDS "SNORT" software.

Keywords*: attacks, encryption, SSL/TLS, IPSEC, IDS*

1. Introduction

The information systems are increasingly a strategic place within companies. Thus notion of risk becomes a source of concern and important to consider given this starting from the design phase of an information system to its implementation and monitoring its operation.

A practice associated with the security of information systems is a point growing importance in the data processing ecosystem that is open and accessible by users, partners and service providers. It becomes essential for companies to know their resources for information system and set to protect sensitive in order to ensure a controlled and rational exploitation of these resources perimeters [1, 2].

The computer security is a set of technical, organizational, legal and human resources to maintain, restore and guarantee the security of information systems. The security of information is based primarily on: availability, confidentiality, integrity, and audibility [3].

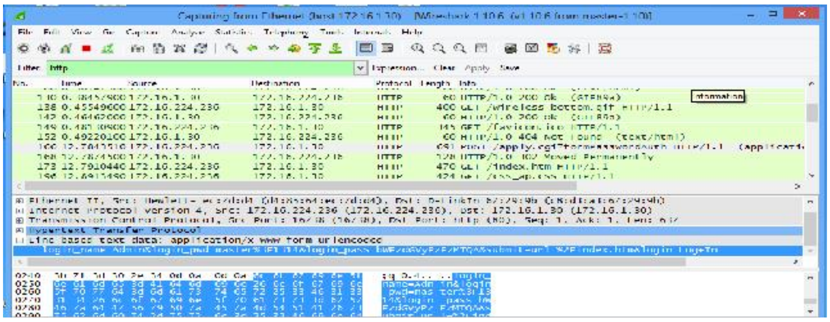
Security of information system based primarily on the protection of its equipment. The security of this equipment covers the following areas: Physical security, security of operation, logical security, application security, and security infrastructures of data processing and telecommunications [4].

There are different facets of security such as: Leading security, legal,and security architecture [5]. This paper is organized as follows: Section 2 deals with network attacks. Different measures of security are descibed in section 3. Section 4 deals with the signature numeric and authentication, the security protocols (SSH, SSL/TLS, Ipsec) are descibed in section 5. IDS open source "SNORT" is deployed in section 6

2. Network Attacks

An attack is the exploitation of vulnerability in a computer system for purposes unknown to the system operator and generally harmful. We can distinguish two types of attacks: Passive attack and active attack. We can distinguish seven types of attacks: Intrusion attack (Sniffing), Man-In-The-Middle, Denial Of Service attack, phishing attack, dictionary attack, brute force attack, and social engineering attack. To carry out attacks on computer systems, hackers use well known domain computer tools. These tools are also used by administrators and security specialists to test the robustness of their information systems, usually as part of a security audit. Among these tools, we can mention [1, 2]: Programs malicious (virus, worm, trojan, spyware), Sniffers (Figure 1), ARP spoofing (Figure 2), the backdoors, and spam (junk mail).

Received July 2, 2015; Revised September 8, 2015; Accepted September 25, 2015


Capturing from Ethernet (host 172.16.1.30) [Wireshark 1.10.6 (v1.10.6 from master-1.10)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Expression... Clear Apply Save

Filter: http

```
No.  Time        Source          Destination     Protocol Length Info
 130 0.38457900 172.16.1.30     172.16.224.236  HTTP       60 HTTP/1.0 200 Ok  (GIF89a)
 138 0.45549000 172.16.224.236  172.16.1.30     HTTP      400 GET /wireless bottom.gif HTTP/1.1
 142 0.46462000 172.16.1.30     172.16.224.236  HTTP       60 HTTP/1.0 200 Ok  (GIF89a)
 149 0.48730900 172.16.224.236  172.16.1.30     HTTP      345 GET /favicon.ico HTTP/1.1
 152 0.49220100 172.16.1.30     172.16.224.236  HTTP       60 HTTP/1.0 404 Not Found  (text/html)
 160 12.7843510 172.16.224.236  172.16.1.30     HTTP      691 POST /apply.cgi?form=passwordauth HTTP/1.1  (applicati
 168 12.7874500 172.16.1.30     172.16.224.236  HTTP      328 HTTP/1.0 302 Moved Permanently
 173 12.7910440 172.16.224.236  172.16.1.30     HTTP      470 GET /index.htm HTTP/1.1
 196 12.8913490 172.16.224.236  172.16.1.30     HTTP      424 GET /[illegible] HTTP/1.1

⊞ Ethernet II, Src: [illegible], Dst: [illegible]
⊞ Internet Protocol Version 4, Src: 172.16.224.236 (172.16.224.236), Dst: 172.16.1.30 (172.16.1.30)
⊞ Transmission Control Protocol, Src Port: 16788 (16788), Dst Port: http (80), Seq: 1, Ack: 1, Len: 637
⊞ Hypertext Transfer Protocol
⊟ Line-based text data: application/x-www-form-urlencoded
    login_name=Admin&login_pwd=master%3E[illegible]&login_pass=[illegible]&submit-url=%2Findex.htm&login=Login

0240  [illegible]                                        [illegible]login_
0250  6e 61 6d 65 3d 41 64 6d 69 6e 26 6c 6f 67 69 6e    name=Admin&login
0260  5f 70 77 64 3d 6d 61 73 74 65 72 25 33 45 31 33    _pwd=master%3E13
0270  [illegible]                                        [illegible]login_pass=[illegible]
0280  [illegible]                                        [illegible]
```

Figure 1. Wireshark capture screen

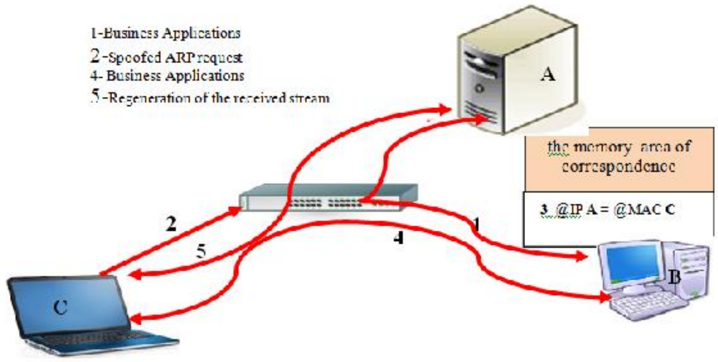


Figure 2. ARP spoofing

3. Measures of Security

The incidents of attacks on computer systems related to the increasing use of computer networks, increase exponentially. To remedy this, simply apply some security measures such as encryption, hashing, and use some security protocols such as SSH, SSL and IPSEC

Encryption is to deface a message by scrambling its members so it is very difficult to reconstruct the original if you do not know the applied transformation.

Encryption is based on two elements: a key and algorithm. The algorithms can be classified by their nature into two types: symmetric encryption and asymmetric encryption [1].

3.1. Symmetric Encryption

Symmetric encryption uses a shared secret key between two interlocutors. Encodes it and the message is decoded with the same key. The problem with this encryption is that we must find a way to transmit the secret key between two interlocutors. There are two categories of private key systems: block ciphers and stream ciphers. For example RC4 (stream cipher & Blowfish block cipher)

RC4 is a stream cipher algorithm for software applications. It was designed by R. Rivest in 1987 for RSA Laboratories [3] despite a number of shortcomings; it is still widely used today, especially because of its high speed. It is used for example in SSL / TLS protocol to ensure the confidentiality of Web transactions in the standard WEP (Wired Equivalent Privacy) for wireless networks [3].

RC4 is decomposed into two algorithms, the Key Scheduling Algorithm (KSA) and the Pseudo Random Generator Algorithm (PRGA) [4].

1) KSA algorithm: The objective of this algorithm is to combine an array called S containing 256 bytes by using a secret key.

```
KSA algorithm
Input : the secret key K of length L
For  i from  0 to 255
```

```
    S[i] := i
endfor
j := 0
for  i from 0 to 255
    j := (j + S[i] + K[i mod L]) mod 256
    swap(S[i], S[j])
endfor
```

***2) PRGA algorithm**:* Once KSA is executed, we get an array of 256 values swapped depending on the secret key. This table will then be used by the PRGA to generate a keystream byte. It is then XOR with the plaintext to get the ciphertext.

```
i := 0
j := 0
while GeneratingOutput:
    i := (i + 1) mod 256
    j := (j + S[i]) mod 256
    swap(S[i],S[j])
    byte_encryption:= S[(S[i] + S[j]) mod 256]
    result_encryption=    byte_encryption  XOR byte_message
endwhile
```

This algorithm ensures that each value of S is exchanged at least once every 256 iterations (Figure 3).

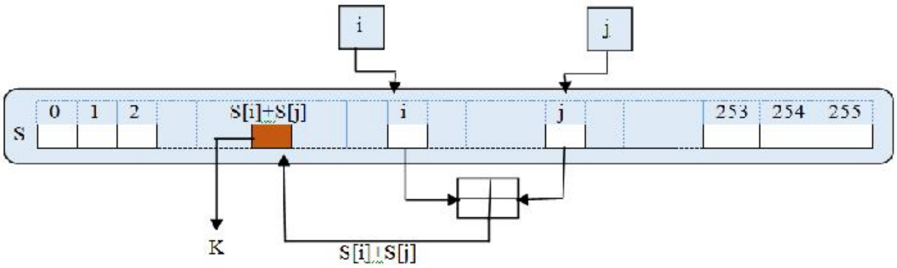


Figure 3. Diagram of generating a byte by RC4

3) **Blowfish**: was designed by Bruce Schneier in 1993 [2] as an alternative to existing algorithms, being fast and free. Blowfish is significantly faster than DES. It is a Feistel encryption, using an encryption function repeatedly 16 times. The block size is 64 bits [2].

It may take a key length varying between 32 bits and 448 bits [2]. Since its inception has been widely analyzed and is now considered a strong encryption algorithm. It is not patented and therefore its use is free of charge. It is used in open source software (including GnuPG and OpenSSH).

The Blowfish algorithm is divided into two parts: the first part handles the expansion of the key and the second part encrypts data.

The following diagram shows the main structure of Blowfish (Figure 4). Each line represents 32 bits. The algorithm maintains two sets of keys: 18 entries in the table P and four S-Boxes of 256 items, are initialized with values derived from the hexadecimal digits of PI (3,14).

The preparation of the structure from the key starts with the initialization of table P and S-Boxes with values from the number PI are expressed in hexadecimal [2]. It then performs an XOR between the secret key and the table entries P (with a cyclic extension of the key if necessary). Blocks of 64 bits, all zero, are then encrypted with the temporary version of Blowfish. The encrypted result replaces the first and second element of the array P. It reiterates the encryption operation with this new version and the previous result. This gives the third and fourth element of P. The algorithm continues thus replacing the entire table of elements P and S-Boxes.

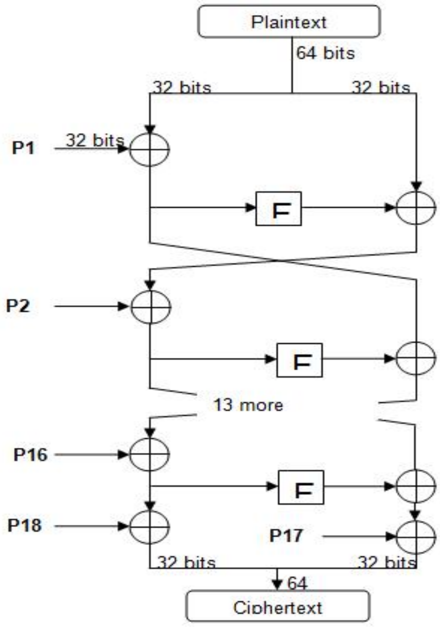


Figure 4. Feistel scheme in Blowfish

In the end, approximately 4KB data must be generated and Blowfish performs 512 iterations to achieve it. Due to these constraints, Blowfish is slow when changing key but very fast for encryption separately [2]. The pseudo code of Blowfishis is folows:

Let's be a 64-bit input plaintext is denoted "x" and Table P is denoted P [i], where "i" is the iteration.

```
Start encryption
   Divided x in 2 : xL and  xR
   For i from 1 to  16
                 xL = xL XOR P[i]
                 xR = F(xL) XOR xR
                 swap xL and  xR
   endfor
   swap  xL and xR
   xR = xR XOR P[17]
   xL = xL XOR P[18]
   x = xL + xR
   return x
   end encryption
```

The function F of the algorithm:

```
Function F( )
begin function  F(input : xL : 32 bits the data)
     Divideed xL in 4 : a, b, c, d
    return ((S1,a + S2,b MOD $2^{32}$) XOR S3,c) + S4,d  MOD $2^{32}$
End function F
```

3.2. Asymmetric Encryption

The concept of public key cryptography was invented by Whitfield Diffie and Martin Hellman in 1976 [3], in solving the problem of distribution of the secret key. This encryption uses two keys, a public key used to encrypt messages and a private key to decrypt it [7]. The public key algorithms are the most widely used by RSA and ALGAMAL [1].

3.2.1. RSA algorithm

Invented in 1978 by Ronald L. Rivest, Adi Shamir and Leonard M. Adleman, RSA is the most popular public key systems. It can be used to encrypt information and / or to sign (digital signature) [7].

Initialization

1) *Choose two primes, p and q.*
2) *Calculate n = p *q*
3) *Choose random **e** such that **e** and ((p - 1)* (q - 1))*
4) *Find d such that (e*d) is divisible by ((p - 1)* (q - 1)), thus: ed = 1 mod ((p - 1) (q - 1)).*
5) *Public key (n, e)*
6) *Private key (n, d) or (p, q, d) if we want to keep p and q*

Encryption/Decryption

1) *The sender creates the ciphertext C from the message m:*
   *$C = m^e$ mod (n**)**, where (n, e) is the public key of the recipient.*
2) *The recipient receives C and performs decryption:*
   $m = c^d$ mod (n), where (n, d) is the private key of the recipient.

Digital signature

1) *The sender creates the signature S from the message m:*
   $S = m^d$ mod (n), where (n, d) is the private key of the sender.
2) *The recipient receives S and m and perform the audit to m:*
   *$m = S^e$ mod(n**)**, where (n, e) is the public key of the sender*

3.3. Hybrid Encryption

This technique has been introduced to take advantage of the two techniques mentioned above, ie the speed of processing encoded symmetric (Table 1) and the power of message encryption by asymmetric cryptography.

Table 1. Comparison Speed Encryption Algorithms

Algorithm	Key length (bits)	speed (Mo/s)
RC4	128	113.35
DES	56	22.19
3DES	168	9.8
AES	128	62.02
RSA	1024	0.02

The principle is quite simple. The communication between two people A and B is symmetric cryptosystem, which makes fast enough to encrypt and decrypt communications. However, this encryption key overcomes the shortcoming of the transmission reliability of the symmetric key encryption/decryption, which is asymmetric.

4. Signature Numeric and Authentication

The principle of digital signature is to comminute the message transmitted by the transmitter. At the receiver side, the message is verified by the same hash function. A special hash function H is applied to the variable length message M. Message H(M), the length is fixed, is the footprint of the original message M. This fingerprint is then encrypted by the private key of the sender and becomes E(H(M)). The assembly (M, E(H(M)) is sent to the receiver [7].

A reception of all (M, E(H(M)) the recipient from E(H(M)) then decrypts it with the public key of the sender to have H'(M). It applies the same H function on the message M for H(M) and compares H(M) and H'(M). If both are identical, message integrity and authentication of the sender is verified [7].

4.1. Hash Function

The main hash functions are:

1) MD5 (Message Digest 5): Developed by Ronald Rivest in 1991, cutting the MD5 any message blocks of 512 bits and computes a 128-bit [7].
2) SHA-1 (Secure Hash Algorithm) is a cryptographic hash function designed by the National Security Agency of the United States (NSA) [17] and published by the Government of the United States as federal standard treatment information. It produces a result (called "hash" or condensate) 160 bits.
3) SHA-2 is, as SHA-1, developed by the NSA to fill gaps of the previous generation functions. A special feature is that this family of hash functions will give fingerprints of different sizes corresponding to the suffix of the function used: SHA-256 will give a checksum of 256 bits, SHA-512 will provide 512 bits [7].
4) The following table summarizes the characteristics of MD5 and SHA-1 [7].

Table 2. Comparing MD5 vs SHA-1

Specifications	MD5	SHA-1-160
Length footprint	128 bits	160 bits
Processing unit	512 bits	512 bits
Number of step	64	80
logic function	4	4
Speed (MB / s)	204,55	72.60

4.2. Authentication Code chopped Message (HMAC)

The HMAC algorithm incorporates a secret key in the hash function. It has been proposed as RFC 2104 and was chosen as a model for the implementation of digital signatures (MAC) to ensure the safety and IP messages in other Internet protocols such as TLS (Tansport Layer Security [8].

The HMAC [8] function is defined as follows:

$$HMAC_K(M)=H[(K\text{ă}opad)||H[(K\text{ă}ipad)||M]]$$

Where:

H : iterative hash function,

K : the secret key padded with zeros so that it reaches the block size of the function H,

M : the message to be authenticated,

"||" denotes concatenation and "ă" a« OR » exclusive,

ipad and *opad*, each of the block size, are defined by: *ipad* = 0x363636...3636 and *opad* = 0x5c5c5c...5c5c. So, if the block size of the hash function 512 is, *ipad* and *opad* are 64 repetitions bytes, respectively, 0x36 et 0x5c [8].

5. Security Protocols

5.1. SSH Protocol (Secure Shell)

SSH (Secure Shell), developed in 1995 by Tatum Ylönen, a Finnish professor. The first version of this protocol, SSH-1, was to secure remote Unix server communications, especially remote controls (rcp, rlogin, telnet, rsh, ftp, ...) [9].

IETF has set up a working group called SECSH (Secure SHell) to standardize and develop SSH protocol version 2.0 which address the security vulnerability of the SSH-1 release [9].

Architecture

SSH uses a client-server architecture to provide authentication, encryption and integrity of data transmitted in a network. Version 2 of the protocol specifies an architecture composed of three layer [9] working together (Figure 5).

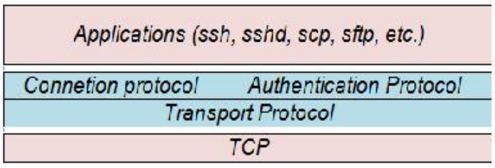


Figure 5. SSH Architecture

1) The SSH transport layer (SSH-TRANS): It provides server authentication, confidentiality and data integrity.

2) The authentication SSH (SSH-AUTH) layer: It used to certify the identity of the client to the server.

3) Layer SSH (SSH-CONN): This layer is based on the authentication layer. It offers a variety of rich services to clients, using the single tube provided by SSH-TRANS

5.2. SSL Protocol

SSL (Secure Socket Layer) was developed by Netscape Communications Corporation to enable client / server applications to communicate securely [9]. TLS (Transport Layer Security) is an evolution of SSL conducted by the IETF.

SSL is a protocol placing himself between the application layer and the transport layer to ensure the confidentiality, authentication, and data integrity when communicating.

SSL is a protocol that sits between TCP / IP and applications that rely on TCP. An SSL session is divided into two phases:

1) A phase handshake: During this phase the client and server authenticate by exchanging certificates between them. Then agree on the hash function, the compression algorithm, encryption algorithm and a key to encrypt messages that will be used in the next step [7].
2) A communication phase during which the data is transmitted as a series of fragments with a maximum size of 16 KB [7], each fragment is protected and transmitted individually. To send a fragment, we first calculate the MAC (Message Authentication Code). The concatenation of the fragment and its MAC is encrypted, thus obtaining an encrypted load (encrypted payload in English).

5.3. IPSEC Protocol

IPSec (IP Security) is a suite of protocols developed by the IETF (Internet Engineering Task Force) in 1992, intended to secure traffic at the IP level (IPv4 or IPv6). The security services are integrity, authentication, protection against replay and confidentiality. IPSEC optional in IPv4, and compulsory for all IPv6 implementation [9].

IPsec has two modes: transport mode which just protects the transported data and tunnel mode which additionally protects the header.

The AH (Authentication Header) is designed to ensure the integrity and authentication of IP datagrams without data encryption. The principle of AH is to add to the original IP datagram to an additional field for the approval to verify the authenticity of the data included in the datagram [7] (Figure 6).

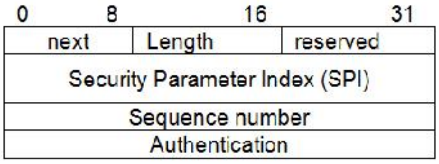


Figure 6. IPsec AH Header

The ESP protocol ensures privacy but can also ensure the authenticity of data. The ESP principle is to generate, from a conventional IP datagram, a new datagram in which the data and possibly the original header are encrypted [11]. ESP can also ensure the authenticity of a data block by adding the authentication and protection against replay through a sequence number (Figure 7).

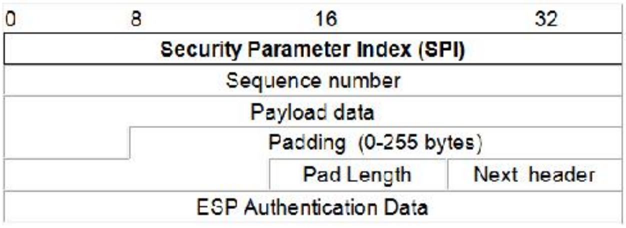


Figure 7. ESP Packet Format

Both extensions can be used separately or combined to obtain the required security services.

1) Security Association

The security association SA (Security Association) is a connection that provides security services to the traffic it carries. This is a data structure for storing all data in a communication, such as encryption algorithm and the hash function for a given communication between two entities associated security settings.

2) Key Management-"IKE protocol"

IKE was defined in RFC 2407, RFC 2408 and RFC 2409 and is defined in RFC 4306 as IKEv2. IKE uses the Diffie-Hellman key exchange to establish a shared secret [12]. A technique of asymmetric cryptography is used to authenticate both parties. IKE (Internet Key Exchange) protocol is responsible for negotiating the connection [8]. Before a transmission may be possible IPSec, IKE is used to authenticate both ends of a secure tunnel exchanging shared keys. This protocol allows two types of authentication, PSK (Pre-Shared Key or shared secret) to generate session keys using RSA or certificates. Both methods are distinguished by the fact that the use of a certificate signed by a third-party called Certification Authority (CA) provides non-repudiation. Whereas with the use of RSA keys, a party may deny being at the origin of messages sent.

3) Operating modes

a. Transport mode

This mode is used to create a communication between two hosts that support IPSec. An SA (Security Association) is established between the two hosts. IP headers are not modified and the AH and ESP protocols are integrated between the header and the header of the transported protocol (Figure 8). This mode is often used to secure a Point-to-Point connection.

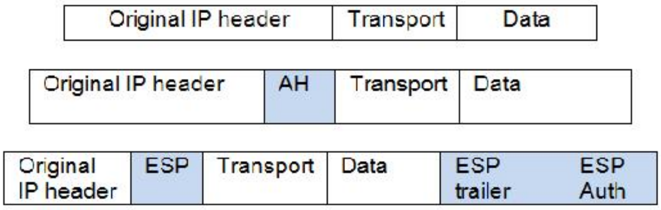


Figure 8. Position AH and ESP in transport mode (IPv4)

b. Tunnel mode

This mode is used to encapsulate IP datagrams in IPsec. SA is applied to an IP tunnel. Thus, the original IP headers are not changed and a specific IPsec header is added (Figure 9). Tunnel mode is used on both terminal equipment and security gateways. This mode ensures greater protection against traffic analysis, because it hides the addresses of the sender and the final destination [7].

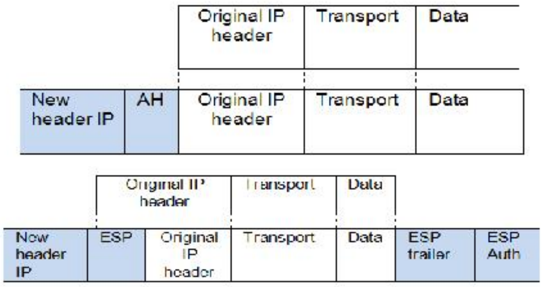


Figure 9. Position AH and ESP in tunnel mode (IPv4)

4) Principle of Operation

The Figure 10 shows all elements of the IPsec protocol, their positions and their interactions.

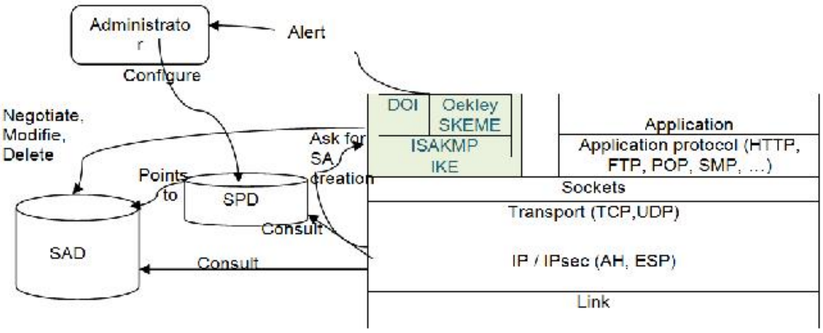


Figure 10. IPsec operating principle [9]

6. Intrusion Detection System

The security of the information system of the company is a big problem today. Unfortunately, hackers and intruders have made many successful attempts on networks of high-level business despite many methods have been developed to secure the network infrastructure and communication on the Internet, including the use of firewalls, encryption and virtual private networks. To complete this security policy, we need monitoring tools for auditing the information system and to detect possible intrusions.

6.1. Host Based IDS (HIDS)

A HIDS is an IDS installed on the resource to be protected and analysis in real time flows relating to this machine to detect intrusions on this resource (Figure 11) as well as logs.

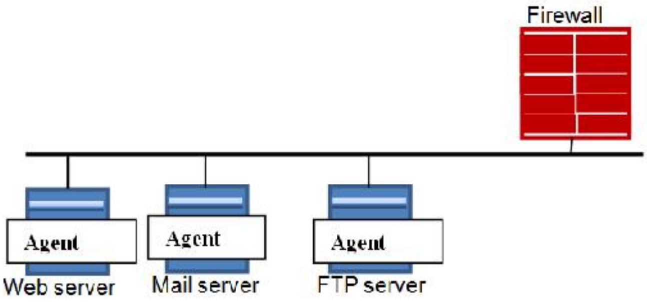


Figure 11. Host Based IDS (HIDS)

6.2. Network IDS (NIDS)

NIDS (Network IDS) is used to analyze and interpreted packets on the network (Figure 12). The implementation of a NIDS is through the placement of sensors at the most strategic levels of the network. These sensors generate alerts if it detects attacks and sends reports to a console that analysis. The sensors and the console are generally on a separate network from that of the company.

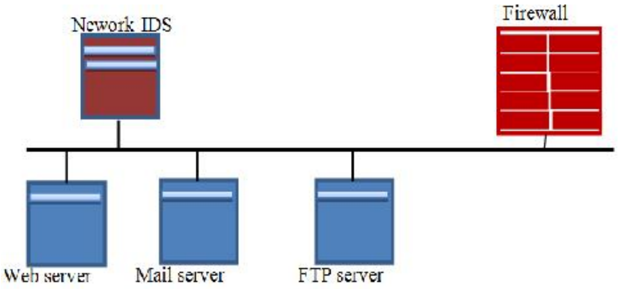


Figure 12. Network IDS

6.3. IDS Hybrid

Hybrid IDS bring together information from various sensors placed on the network. Their name "hybrid" comes from the fact that they are able to gather information from both a HIDS system of NIDS (Figure 13). IDS have two main mechanisms to detect intrusions. The first is the detection of attack signatures listed in a knowledge base system. The second detects suspicious of a user on a given post activity.

The detection scenario is based on a database of known attack signatures (knowledge based detection) scenarios. This detection mechanism is effective against attacks clearly listed, but not against new attacks.

The purpose of the detection by behavioral approach (anomaly detection) is to signal any anomalies on an information system. Indeed, it goes through a phase called learning the normal behavior of the system. Suspicious behavior will be declared and will be based on models or profiles established during the first phase of a normal behavior.

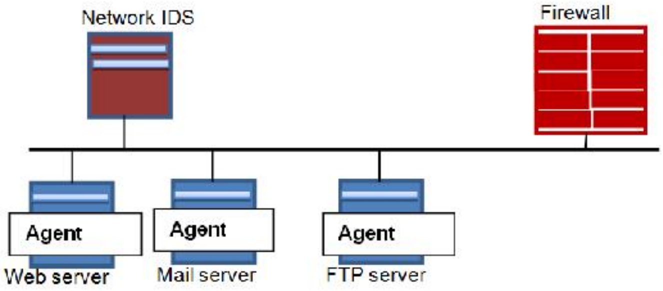
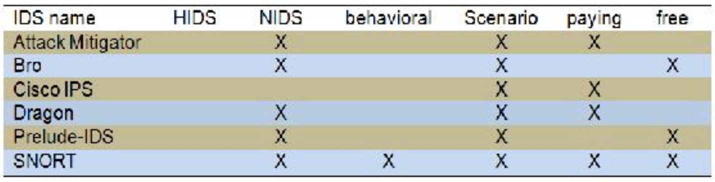


Figure 13. IDS Hybrid

Currently, intrusion detection systems have really become indispensable. So they always fit in a context and an architecture that imposes constraints can be very diverse. That is why there is no single assessment grid for this type of tool. Yet a number of criteria can be met (Reliability, Responsiveness, Easy implementation and adaptability, Performance, Multichannel Classification).

Here are some tools that are available; they are distinguished by their method of detection as well as their economic model [13]. Then, SNORT is used (Figure 14).



IDS name	HIDS	NIDS	behavioral	Scenario	paying	free
Attack Mitigator		X		X	X	
Bro		X		X		X
Cisco IPS				X	X	
Dragon		X		X	X	
Prelude-IDS		X		X		X
SNORT		X	X	X	X	X

Figure 14. Table of characteristics of some IDS

Snort is a system for detecting intrusion free under GPL, originally written by Martin Roesch. Commercial versions integrating hardware and media services are sold by Sourcefire.

Snort can also be used with other open source projects such as SnortSnarf, ACID, Sguil and BASE (Basic Analysis and Security Engine) to provide a visual representation of data regarding potential intrusions [1].

When suspicious behavior is intercepted by Snort based signatures, events are registered by Snort. It is possible to write the logs in the database directly. However, in an optimization concern (release of resources), we will use Barnyard, an abstraction layer Snort. Thus, Snort record the events in the logs unified format (Fast Unified Logging) and these will be operated by Barnyard for inclusion in the database [2].

Guardian ensures blocking IP addresses detected as suspicious. In addition, BASE and/or Sguil will analyze these attacks (Figure15).

The physical location of the probe SNORT on the network has a significant impact on its effectiveness. In the case of classical architecture, consisting of a Firewall and DMZ, three positions are generally possible (Figure 16):

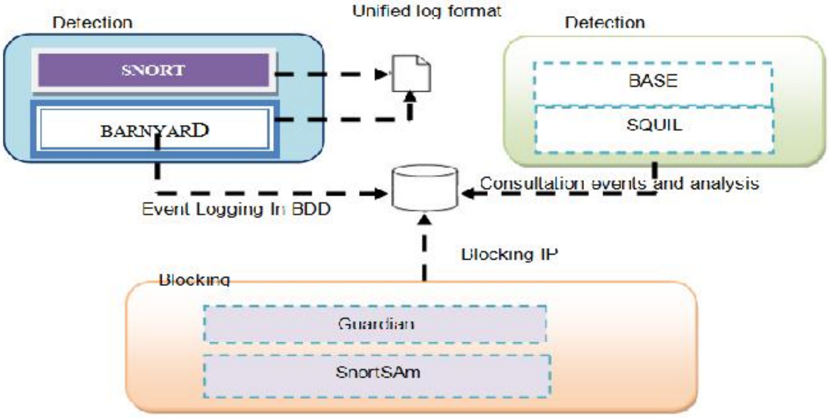


Figure 15. Snort architecture

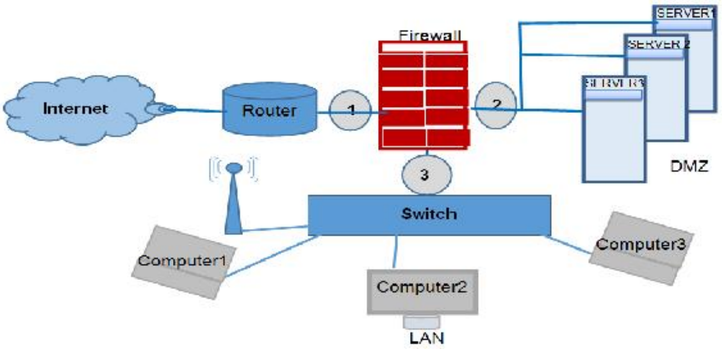


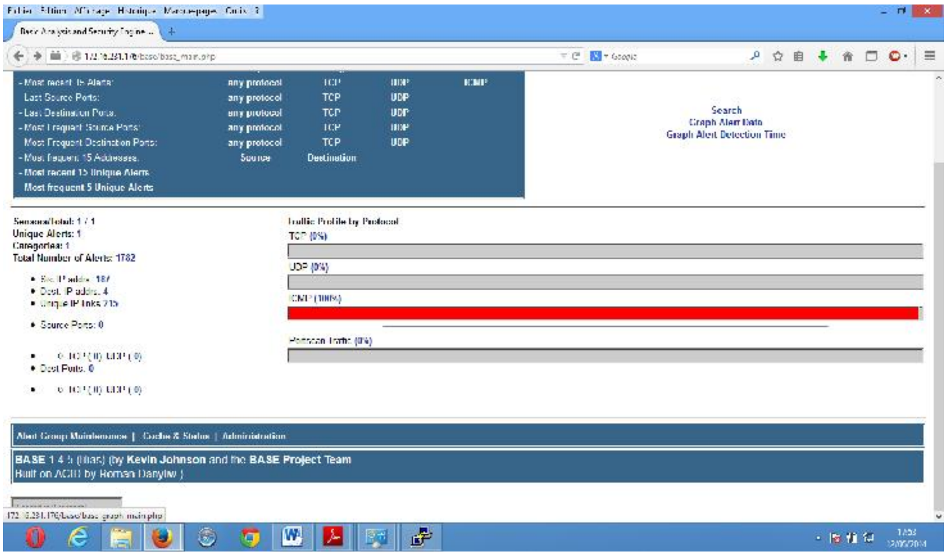
Figure 16. Positioning Snort

There are 4 modes of executions Snort: Sniffer mode, packet logger mode, NIDS Mode, and IPS Mode

SNORT rules are composed of two distinct parts: the header and options. The header specifies the type of alert to generate (alert, log and pass) and indicate the basic fields necessary filtering: the protocol and IP addresses and source ports and destination. The options specified in brackets are used to refine the analysis by decomposing signature in different values to be observed among some fields in the header or in the data [16].

Basic (Basic Analysis and Security Engine) is written in PHP web under GPL license and built on the ACID code, which allows you to view alerts generated by the software Snort intrusion detection interface. The interface allows the classification of group alerts, display charts and alerts search by various criteria (Figure 16).

SnortReport is a graphical interface (Figure 17) used to display and analyze the alerts (Figure 18) generated by intrusion detection system Snort, stored in a MySql database.


Fichier Edition Affichage Historique Marque-pages Outils ?
Basic Analysis and Security Engine ...
172.16.231.176/base/base_main.php

- Most recent 15 Alerts: any protocol TCP UDP ICMP
Last Source Ports: any protocol TCP UDP
- Last Destination Ports: any protocol TCP UDP
- Most Frequent Source Ports: any protocol TCP UDP
Most Frequent Destination Ports: any protocol TCP UDP
- Most frequent 15 Addresses: Source Destination
- Most recent 15 Unique Alerts
Most frequent 5 Unique Alerts

Search
Graph Alert Data
Graph Alert Detection Time

Sensors/Total: 1 / 1
Unique Alerts: 1
Categories: 1
Total Number of Alerts: 1782

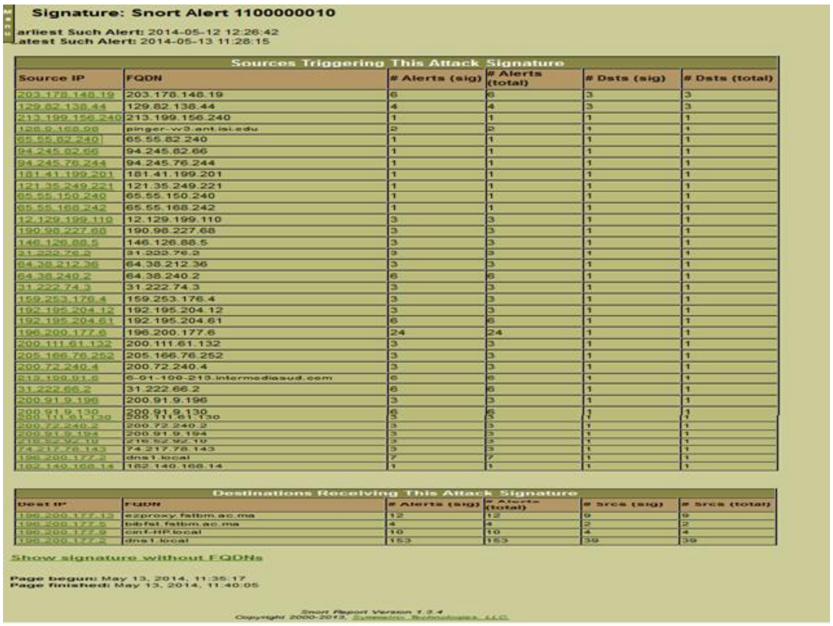
- Src. IP addrs: 187
- Dest. IP addrs: 4
- Unique IP links 215
- Source Ports: 0
- o TCP (0) UDP (0)
- Dest Ports: 0
- o TCP (0) UDP (0)

Traffic Profile by Protocol
TCP (0%)
UDP (0%)
ICMP (100%)
Portscan Traffic (0%)

Alert Group Maintenance | Cache & Status | Administration

BASE 1.4.5 (lilias) (by Kevin Johnson and the BASE Project Team
Built on ACID by Roman Danyliw)

Figure 17. Screenshot of BASE


Signature: Snort Alert 1100000010

arliest Such Alert: 2014-05-12 12:26:42
atest Such Alert: 2014-05-13 11:28:15

Sources Triggering This Attack Signature

Source IP	FQDN	# Alerts (sig)	# Alerts (total)	# Dsts (sig)	# Dsts (total)
203.178.148.19	203.178.148.19	6	6	3	3
129.82.138.44	129.82.138.44	4	4	3	3
213.199.156.240	213.199.156.240	1	1	1	1
128.9.168.98	pinger-w3.ant.isi.edu	2	2	1	1
65.55.82.240	65.55.82.240	1	1	1	1
94.245.82.66	94.245.82.66	1	1	1	1
94.245.76.244	94.245.76.244	1	1	1	1
181.41.199.201	181.41.199.201	1	1	1	1
121.35.249.221	121.35.249.221	1	1	1	1
65.55.150.240	65.55.150.240	1	1	1	1
65.55.168.242	65.55.168.242	1	1	1	1
12.129.199.110	12.129.199.110	3	3	1	1
190.98.227.68	190.98.227.68	3	3	1	1
146.126.88.5	146.126.88.5	3	3	1	1
31.222.76.2	31.222.76.2	3	3	1	1
64.38.212.36	64.38.212.36	3	3	1	1
64.38.240.2	64.38.240.2	6	6	1	1
31.222.74.3	31.222.74.3	3	3	1	1
159.253.176.4	159.253.176.4	3	3	1	1
192.195.204.12	192.195.204.12	3	3	1	1
192.195.204.61	192.195.204.61	6	6	1	1
196.200.177.6	196.200.177.6	24	24	1	1
200.111.61.132	200.111.61.132	3	3	1	1
205.166.76.252	205.166.76.252	3	3	1	1
200.72.240.4	200.72.240.4	3	3	1	1
213.199.91.6	6-91-199-213.intermediasud.com	6	6	1	1
31.222.66.2	31.222.66.2	6	6	1	1
200.91.9.196	200.91.9.196	3	3	1	1
200.91.9.130	200.91.9.130	6	6	1	1
200.111.61.130	200.111.61.130	3	3	1	1
200.72.240.2	200.72.240.2	3	3	1	1
200.91.9.194	200.91.9.194	3	3	1	1
216.52.92.10	216.52.92.10	3	3	1	1
74.217.78.143	74.217.78.143	3	3	1	1
196.200.177.2	dns1.local	7	7	1	1
182.140.168.14	182.140.168.14	1	1	1	1

Destinations Receiving This Attack Signature

Dest IP	FQDN	# Alerts (sig)	# Alerts (total)	# Srcs (sig)	# Srcs (total)
196.200.177.13	ezproxy.fstbm.ac.ma	12	12	9	9
196.200.177.5	bibfst.fstbm.ac.ma	4	4	2	2
196.200.177.9	cinf-HP.local	10	10	4	4
196.200.177.2	dns1.local	153	153	39	39

Show signature without FQDNs

Page begun: May 13, 2014, 11:35:17
Page finished: May 13, 2014, 11:40:05

Snort Report Version 1.3.4
Copyright 2000-2013, Symmetrix Technologies, LLC

Figure 18. Alert details displayed by Snort Report

7. Conclusion

The information system of a company can be vital to its operation. It is therefore necessary to ensure its protection, to fight against the threats to the integrity, confidentiality and availability of resources. The computer abuse is often the cause of these threats, for data theft or espionage; anyone can improvise hacker with the right tools.

In this paper we addressed the security principles and finally we set the SNORT IDS detected for different flaw in our Firewall to define a new strategy for security and define a new policy at our Firewall.

Many skills are needed to ensure optimal security, but it is impossible to guarantee the security of information system 100%. In the end we conclude by saying "Prevention is better than cure".

References

[1] Laurent Bloch Christophe Wolfhugel. Sécurité informatiquePrincipes et méthode à l'usage des DSI, RSSI et administrateurs. Second Edition.

[2] Cédric Llorens, Laurent Levier, Denis Valois. Tableaux de bord de la sécurité réseau. Second Edition.

[3] Santanu Sarkar. Proving empirical key-correlations in RC4. *Information Processing Letters*. 2014; 114(5): 234-238.

[4] Allam Mousa, Ahmad Hamad. Evaluation of the RC4 Algorithm for Data Encryption. *International Journal of Computer Science & Applications*. 2006; 3(2): 44-56.

[5] P Karthigai Kumar, K Baskaran. An ASIC implementation of low power and high throughput blowfish crypto algorithm. *Microelectronics Journal*. 2010; 41(6): 347-355.

[6] Mohd Anuar Mat Isaa, Miza Mumtaz Ahmadb, Nor Fazlida Mohd Sanib, Habibah Hashima, Ranllan Mahmodb. *Cryptographie Key Exehange Protocol with Message thentieation Codes (MAC) using Finite State Machine*. Procedia Computer Science. 2014; 42: 263-270.

[7] Van Quang Dao. Contribution à l'étude de la qualité de service pour les protocoles sécurisés de télécommunication. Thesis. Paris, Univeristy Paris XII - Val of Marne; 2005.

[8] Pierre-Alain Fouque, Gaëtan Leurent, Phong Q Nguyen. Full Key-Recovery Attacks on HMAC/NMAC-MD4 and NMAC-MD5. *LNCS*. 2007; 4622: 13-30.

[9] Ibrahim HAJJEH. Sécurité des échanges: Conception et validation d'un nouveau protocole pour la sécurisation des échanges. Thesis. Paris, Superior National School of Telecommunications in Paris; 2004.

[10] A Avi Turiel, Commtouch. IPv6: new technology, new threats. *Network security*. 2011; 2011: 13-15.

[11] Kenneth G Paterson. A cryptographic tour of the IPsec standards. *Information security technical report*. 2006. 11(2): 72-81

[12] C Kaufman. Internet Key Exchange (IKEv2) Protocol. *RFC4306*. 2005.

[13] Michaël AMAND, Mohamed NSIRI. *Etude d'un système de détection d'intrusion*. IENAC 08. 2011.

[14] K Salah, A Kahtani. Performance evaluation comparison of Snort NIDS under Linux and Windows Server. *Journal of Network and Computer Applications*. 2010; 33(1): 6-15.

[15] http://www.aldeid.com/wiki/Snort. 2014.

[16] Kyu Hee Lee, Sang Kyun Yun. Hybrid memory-efficient multimatch packet classification for NIDS. *Microprocessors and Microsystems*. 2015; 39(2): 113-121.

[17] http://fr.wikipedia.org/wiki/SHA-1. 2015.