International Journal of Engineering Works
Kambohwell Publisher Enterprises
ISSN: 2409-2770 Vol. 2, Issue 9, PP. 84-92, Sept. 2015
www.kwpublisher.com

Security by IDS-AM-Clust, honeyd and honeycomb

CHAIMAE SAADI *, HABIBA CHAOUI **

Abstract-Various tools and methods are developed to secure our information systems against hackers.This work proposes a new security architecture of IS, using a combination of Honeyd and their plugin honeycomb with intrusion detection system based on mobile agent and data mining algorithm Clust-density. theprinciplal goal is to detect intrusions flowing through the network. also, we show that by using this architecture, we obtained a higher level of security and we can study the behavior of the pirates and their techniques to evaluate the system in which it is implemented by simulating a vulnerable machine and /or network.

Keywords: Honeypot, Honeyd, Honeycomb, IDS, Mobile Agent, Clust-Density, Attacks

I. INTRODUCTION

The major challenge of the security of information systems is the study of the hacker's behavior. For this,several researcher enhances the aspect of honeypots. LanceSpitzner has sets the honeypot as a secure resource implemented and aimed to attract hackers to be attacked or compromised [1]. It comes to establish a way to control attacks and the activities of the attackers by giving them access to some services, sometimes emulated, so they can interact with them while limiting the damage caused by these attacks that the attacker cannot not access the real production servers. However, the quantity and quality of the collected information are directly proportional to the degree of interaction offered by the honeypot. As a result, if the services offered are very limited, the honey pot will not be very attractive to gather good information .

IDS intrusion detection system provides an information base managed by an administrator who is responsible of the updating. As a result, the strength and weakness of IDS are related to the human factor. Indeed, the maintenance process requests a high level of competence of the administrator and the intensive updates to the data base in order to prevent new attacks[2]. These two requirements are unfortunately not always satisfied.

Affilation : Systems Engineering Laboratory, Data Analysis and Security Team National School of Applied Sciences, University Ibn Tofail, Kénitra, Morocco

Emails:
chaimaesaadi900@gmail.com *
mejhed90@gmail.com **

Moreover, the time response of IDS for the same attacks depends on how the detection rules are written. Accordingly, several IDS configured by different administrators will react differently to the same attacks. A simple IDS detects only the known and produced false positives by the attacks. The Honeypots provide to lighten both of these problems. Indeed, as the traffic flowing inside the honeypot has no other place to be, one can automatically considered suspect and thus remove false positives. More as we let the pirate in , we can analyze the attack he led even if it was in tell now unknown [3].

A Honeypot acts as a "network surveillance camera" and allows collecting and providing information of great value. This information is used to generate in a uniform manner of attack signatures and supplying in real-time the information databases of IDS. As a result, the Honeypot technique allowed to collect information on the activities of hackers. The Information that is analyzed, subsequently to create attacks models allowing enriching the IDS information base [4]. In this concept, this paper allows to improved of the techniques present by the honeypot due to a combination of a low interaction honeypot with an intrusion detection system. Based on mobile agent and Clust-density algorithms 'IDS-AM-Clust' which to detect a high rate of intrusion, to study the behavior of hackers and minimize the rates of false positive and negative .

This paper is organized as follows: section II describes some security tools adapted to our system and defines the function of each one. Section III presents the tests achieved and the interpretation of the results obtained by the proposed system.

II. TOOLS AND METHODS

In this section, we present the various security tools such as:

A. Intrusion detection based on mobile agents and Clust-density IDS-AM-Clust:

To improve the ability of intrusion detection systems based on mobile agents [5] or Clust density [6], the intrusion detection system aims to merge the two latest technology [5] [6] in a single IDS named "IDS-AM-Clust". This was the subject of a work already realized by our team [7].

The following figure (fig.1) shows the flow of network traffic process in our mobile agents using Clust-density:

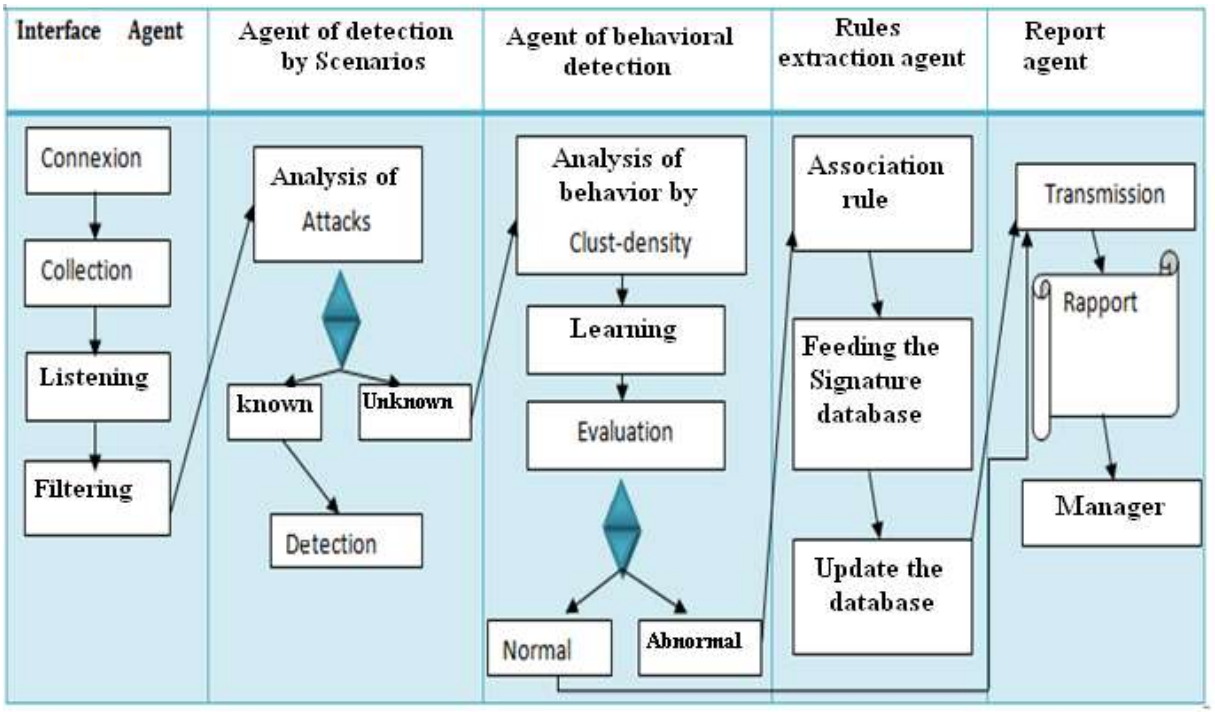

Figure1: Process of detecting intrusion by IDS-AM-Clust

During our first connection, interface agent listens to network traffic in order to put a filter on the packets collected. Then, the agent of detection by scenarios analysis collected and filtered traffic to detect network connections that match the attacks whose signatures are available; If the problem is resolved, the system triggers an alert detection, otherwise it passes this traffic to the behavioral detection who offers the combination of IDS distributed with a Clust-density-based behavioral detection technique that is divided into two phases: a learning phase to identify the normal behavior of the users of the network. After the learning phase, an evaluation phase comes to distinguish among new patterns identified, those who are normal and those that are abnormal. Subsequently, extraction of the Rules agent summarizes network which are identified as abnormal by the ADC connections and feeds the signature library and finally the report agent transmit messages (report, log, alert) to the system administrator. The development of this system was using Sun Java Develop Kit 7 and 3.7 platform JADE (Java Agent Development) that simplifies the implementation of multi-agent systems [7] .In addition, Open source library used is the JPCAP 0.7.

Honeyd

We have chosen to work with the Honeyd like Honeypot as weak interaction because it is able to emulate several services and can control millions of IP addresses at the same time. Honeyd is developed by Niels Provos of the University Michigan. The first version was launched in April 2002. Honyd simulates only the services. It is configured under the form of a production Honeypot and used for the detection of attacks and unauthorized activities [8].

In addition, it is possible to install one or more virtual honeypots in weak interaction with different personalities (systems) and services on a single machine, combining them with IP addresses that are not yet used in the actual network. Another advantage is its ability to control millions of IP addresses and declare a few thousands of others at the same time.

Honeyd can detect intrusion in all TCP ports, the emulated services are not required for detection, but only for interaction with the attackers [9].

The Demon honeyd can accommodate up to 65,536 virtual hosts, which allow the user to create the complex architectures and explores the large networks.

a. Operation of honeyd

The Honeyd is a possibility to detect the information that flows across the network, two methods are used to allow traffic redirection,two methods are used to allow the redirection of traffic:

- Blackholing: this method uses a simple router between the local network and the internet to redirect any traffic to Honeyd [10].

ISSN: 2409-2770

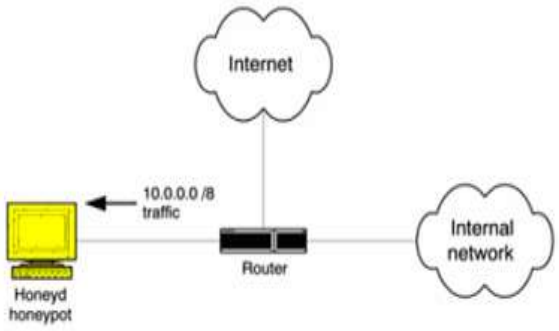

Figure 2 :Blackholing [10]

With the technique 'blackholing', Honeypot is put in a nonproducingnetwork, all IP addresses in the IP class of the networks are configured into the router to forward traffic from the network to the Honeypot.

- ARP Spoofing for this method, Honeyd uses the ARPD module. Each request "who is?" of the ARP protocol to find the MAC address corresponding to an IP address of a machine created by the Honeypot, owned by the honeynet (network of fictitious machines of Honeypost created by Honeyd), ARPD responds by giving MAC the address of the machine on which is installedthe honeypots in the network.
- Honeyd is not a notification or alert integrated solution. For this reason we used a third-party technology as zenmap: an OpenSource tool that can check the logs for specific messages [11].

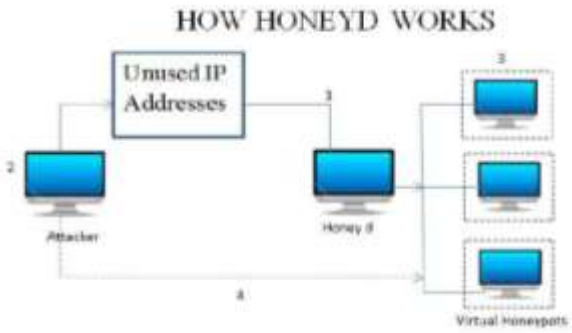

Figure 3: operation of honeyd [11]

Honeyd can create several virtual honeypots to mislead attackers on all unused addresses.

So that the operation of our system is rigid, we chose to add the honeycomb plugin to our honeyd.

b. *Honeycomb*

Honeycomb is a system that can automatically generate signatures by analyzing traffic on a Honeypot. The system produces signatures of good quality. This system is based on the LCS (Longest Common Substring) algorithm [12].

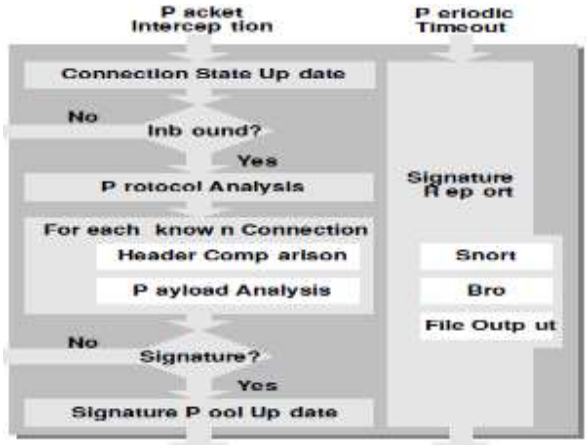

Figure 4: operation of honeycomb [12]

The LCS algorithm works as follows:

- If there is an state of existing connection to a new package, this State is updated,
- Otherwise a new State is created.
- Otherwise if package is not connected, the treatment stops here.
- Otherwise the Honeycomb performs an analysis of the network and transport layer protocol.
- For each stored connection:

- Honeycomb performs the comparison of header to detect matches IP networks, TCP sequence numbers,...ect.
- If the bindings have the same destination port, Honeycomb tries to detect the patterns on the exchanged messages.
- Otherwise if no useful signature created in the previous step, processing stops.
- Otherwise the signature is used to increase the database signature.

To detect repetition of a sequence of bytes, the LCS algorithm can be applied in different packages or streams. This algorithm allows to automatically generate signatures format 'Snort' corresponding to the attempted attacks detected on the honeyd [13].

c. Using the plugin in the Honeycomb honeyd:

The authors of [14] present the honeycomb tools as a honeyd plugin tool to concatenate the sent messages of the attackers by using the LCS algorithm to generate automatically the attack signatures. This concatenation of messages is designed to build a unique signature. The choice of critical packages is based on the analysis of the payload of the information collected on attacks [14].

ISSN: 2409-2770

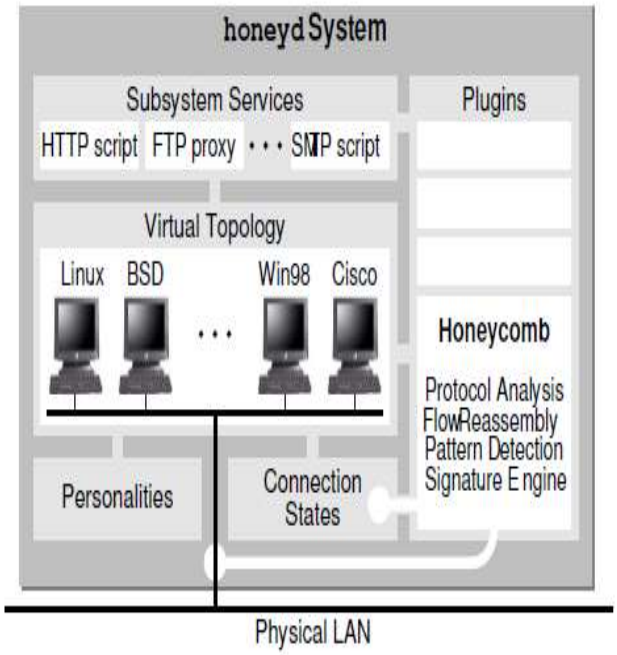

Figure 5 :Honeyd and honeycomb [14]

Honeycomb architecture is illustrated as a typical configuration of honeyd (fig.5). It simulates a number of various machines, each of these machines is running a number of preconfigured services. The Honeycomb plugin is hooked in the thread to see incoming and outgoing connections and subsequently generate these specific attack signatures.

III. TEST AND RESULTE.

Our work is aimed to develop a new system that combines with the IDS-AM-Clust honeyd and know their level of intrusion detection. Which is lead to better interpret the obtained results, we compared the previous results with oursystem: honeyd-IDS.

A. Proposed system

a. Model test

Our test environment consists of the following facilities:

- Honeyd: the honeypot chosen for our working model, including the plugin honeycomb that we configured in our honeydand is installed on Ubuntu 12.04;
- Snort: the first type of IDS selected for the first test model, it is installed on a machine Ubuntu 12.04;
- IDS based on mobile agents is CLust density (present in the previous section): the second IDS chosen for the assessment of the second model of test;
- Kali Linux: the machine that generates the attacks.
- Windows XP machine: the machine target

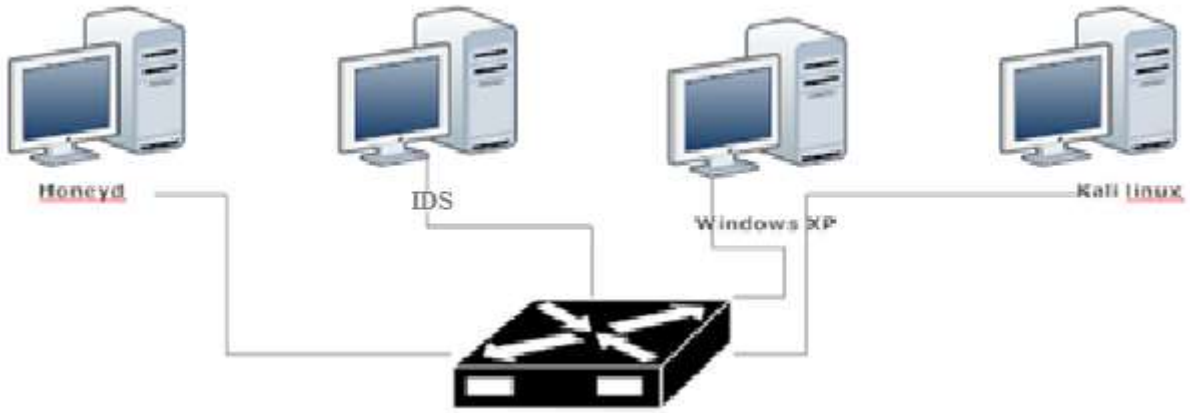

Figure 6 : model test

ISSN: 2409-2770

b. *Characteristics of equipment usedfor test*

TABLE 1: CHARACTERISTICS OF EQUIPMENT

Machine	@IP	Characteristics	Description
Kali linux	192.168.185.151	CPU : Core i5 Ram 1GB Hard disk 20GB 1 network interface : Host Only	machine generatingattacks
Honeyd& plug-in Honeycomb	192.168.185.201	CPU : Core i5 Ram 1GB Hard disk 20GB 1 network interface: Host Only	Installed on Ubuntu 12.04
IDS-AM-Clust	192.168.185.163	CPU: Core i5 Ram 1GB Hard disk 20GB 1 network interface: Host Only	Installed on Ubuntu 12.04
Windows XP	192.168.185.180	CPU : Core i5 Ram 1GB Hard disk 20GB 1 network interface : Host Only	target machine

c. *Components honeyd*

- Scripts feints :these are the scripts written for a specific purpose of each person, these scripts a little simulate reactions for each type of service (for example: telnet, ssh, or iis server). Their goal is to capture some information in the hacker queries.
- Proxy: to well analyze the activities of the pirate, the shipping method of an attack to a real machine is one of the most used methods in this field (for example: with the encrypted protocol SSH, using a sniffer or modifying the kernel for capturing data before they are encrypted to send to the pirate).
- Virtual host: Honeyd can simulate virtual hosts by emulating all existing ports and services, as well as different operating systems, which is called 'Personality' in Honeyd.
- Virtual network: Honeyd can simulate also networks and routers to connect these networks; also the link between the hosts and their IP addresses, each host is combined with an IP by the 'bind' definition.
- The installation of the honeyd allows generating a file configuration that contains the configuration of our virtual network‘honeyd.conf’. The configuration by default of filehoneydisshown in figure 7.

ISSN: 2409-2770

```
route entry 10.0.0.1
route 10.0.0.1 link 10.2.0.0/24
route 10.0.0.1 add net 10.3.0.0/16 10.3.0.1 latency 8ms bandwidth 10Mbps
route 10.3.0.1 link 10.3.0.0/24
route 10.3.0.1 add net 10.3.1.0/24 10.3.1.1 latency 7ms loss 0.5
route 10.3.1.1 link 10.3.1.0/24

# Example of a simple host template and its binding
create template
set template personality "Microsoft Windows XP Professional SP1"
set template uptime 1728650
set template maxfds 35
# For a complex IIS server
add template tcp port 80 "sh /usr/share/honeyd/scripts/win32/web.sh"
add template tcp port 22 "/usr/share/honeyd/scripts/test.sh $ipsrc $dport"
add template tcp port 23 proxy $ipsrc:23
add template udp port 53 proxy 141.211.92.141:53
set template default tcp action reset
# Use this if you are not running honeyd as 'honeyd' user:
# Debian-specific (use nobody = 65534 instead of 32767)
# set template uid 65534 gid 65534

create default
set default default tcp action block
set default default udp action block
set default default icmp action block

create router
set router personality "Cisco 1601R router running IOS 12.1(5)"
set router default tcp action reset
add router tcp port 22 "/usr/share/honeyd/scripts/test.sh"
add router tcp port 23 "/usr/share/honeyd/scripts/router-telnet.pl"

bind 10.3.0.1 router
bind 10.3.1.1 router
bind 10.3.1.12 template
bind 10.3.1.11 template
```

Figure 7: configuration file honeyd

We have created two machines,

• Windows

• Linux

A mac address has been assigned to each of them, a static IP address assigned to the linux machine while the addresses are dynamically allocated on the Windows machine. Several ports or services have been opened: ssh, apache,...ect.

When the Honeyd receives a packet directed to one of these addresses, he will use the associated profile and will respond according to its configuration.

The honeycomb tries to capture repetitions of a sequence of bytes, this sequence is represented as follows [15]:

Code	Definition	Actor	Protocole	Port	Signature

where:

- ➢ Code: is an identifying byte sequence that presents the basic action, it must be unique.
- ➢ Definition (definition of the action): is used to describe the action captured.
- ➢ Actor (actor of action): can be an attacker or a victim.
- ➢ Protocol: the protocols involved are: ARP, RARP, IP, ICMP, TCP, and UDP.
- ➢ Port (Port of destination of the action): this port will determine the service layer application attacked by the intruder. Indeed, there is no need to define the source of the port action since this port is generally chosen randomly by the source machine to the attacker, that is to say for the same type of action can have different port source for each attack, the same for source IP addresses and locations, they have no meaning in an elementary action, since both addresses are changed for each new attack even if it is the same type of action.
- ➢ Signature (Signature of the payload package of action): plays a very important role in the determination of the action captured; it will contain the raw data from the application layer protocol. Indeed, there are a very large number of application protocols among those who are standards like HTTP, Telnet, FTP, etc, and those who are not. Therefore, it is quite difficult to analyze packets of all these protocols to generate attack scenarios. To resolve this problem, it takes the load of the Protocol as it, and it is in the "Signature of the load of the action package.

Figure 8 shows an example of an attack signature generated by the honeycomb:

```
alert tcp 80.0.0.0/8 any -> 192.168.169.2/32 80 (msg: "Honeycomb Mon jully 30 14h30m10 2015 "; flags: A; flow: established; content: "|04 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01
```


ISSN: 2409-2770

01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 DC C9 B0|B|EB 0E 01 01 01 01 01 01 01|p|AE|B|01|p|AE|B|90 90 90 90 90 90 90 90|h |DC C9 B0|B|B8 01 01 01 01|1|C9 B1 18|P|E2 FD|5 |01 01 01 05|P|89 E5|Qh.dllhel32hkernQhounthickChGetTf|B9|llQh32.dhws2_f |B9|etQhsockf|B9|toQhsend|BE 18 10 AE|B|8D|E|D4|P|FF 16|P|8D|E|E0|P|8D|E|F0|P|FF 16|P|BE 10 10 AE|B|8B 1E 8B 03|=U |8B EC|Qt|05 BE 1C 10 AE|B|FF 16 FF D0|1|C9|QQP|81 F1 03 01 04 9B 81 F1 01 01 01 01|Q|8D|E|CC|P|8B|E|C0|P|FF 16|j|11| j|02|j|02 FF D0|P|8D|E|C4|P|8B|E|C0|P|FF 16 89 C6 09 DB 81 F3|

Figure8: Example of attack signature Snort create by the honeycomb

B. Results obtained

To study the effectiveness of our system which is based on collaboration between the honeyd bringing the honeycomb as plugin and our IDS-AM-Clust intrusion detection system, we first had to study the types of attacks that can be generated by our system. This is done by generating a file log present in the following figure:

```
root@ubuntu:~# nmap -p0-65535 192.168.185.201
Starting Nmap 5.61TEST4 ( http://nmap.org ) at 2015-06-30 20:15 PDT
Nmap scan report for 192.168.185.201
Host is up (0.00028s latency).
Not shown: 65506 closed ports
PORT      STATE    SERVICE
21/tcp    open     ftp
22/tcp    open     smurf
23/tcp    open     telnet
25/tcp    open     smtp
53/tcp    open     buffer
80/tcp    open     http
111/tcp   open     phf
3306/tcp  open     mysql
3632/tcp  open     distccd
5432/tcp  open     postgresql
5900/tcp  open     Ipsweep
139/tcp   open     satan
…….
445/tcp   open     microsoft-ds
512/tcp   open     nmap
513/tcp   open     login
514/tcp   open     pod
1099/tcp  open     rmiregistry
6000/tcp  open     X11
6667/tcp  open     irc
6697/tcp  open     unknown
8009/tcp  open     ajp13
1524/tcp  open     land
2049/tcp  open     nfs
2121/tcp  open     ftp-write
43729/tcp open     unknown
44813/tcp open     unknown
55852/tcp open     unknown
……..
8180/tcp  open     unknown
8787/tcp  open     unknown
39292/tcp open     unknown
MAC Address: 00:0C:29:9A:52:C1 (VMware)
```

Figure 9 : log file generated by our system

a. Types of attacks detected

The results showed that our system is able to detect multi-attacks classified into four types.therefore, the comparison of results obtained with IDS-AM-Clust shows the percentages are high relative to the latter. Accordingly, the detectionscenarios our Honeyd-honeycomb-IDS-AM-Clust system is better than IDS-AM-Clust alone.

TABLEAU2 : TYPES OF ATTACKS DETECTED BY OUR SYSTEM

Types of attacks	**DOS**	**2UR**	**R2L**	**Prob**
IDS-AM-Clust [5-7]	20%	9%	5%	15%
Honeyd+honecomb+IDS-AM-Clust	23%	9%	6%	17%

ISSN: 2409-2770

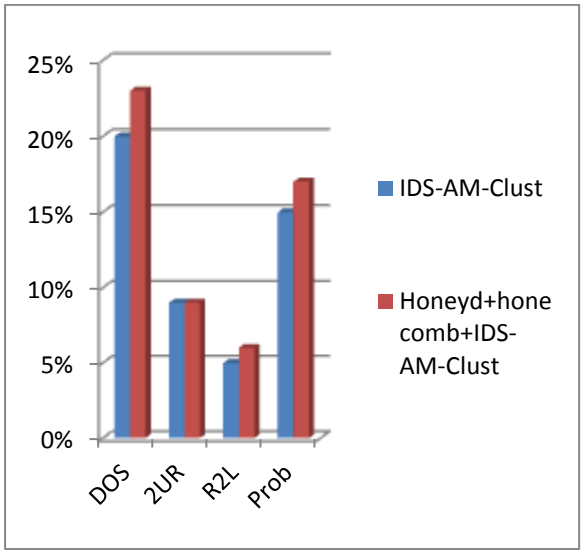

Figure10: Comparison between attacks detected by IDS-AM-ClustetHnoeyd+IDS-AM-Clust

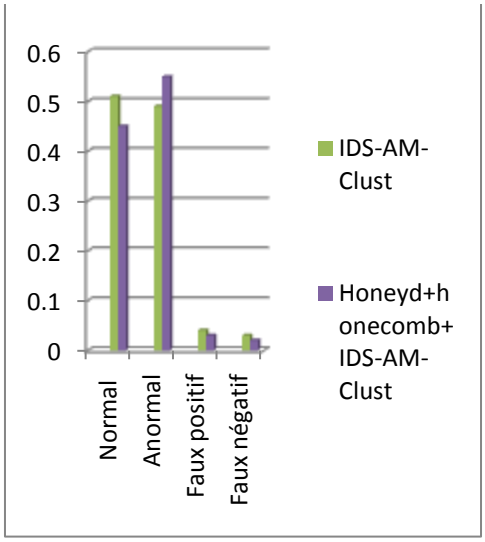

Figure 11: Comparison Intrusion detection rate and FP and FN rates

b. Detection rates, false positive and negative

In order to detect the abnormal connection and minimize false positive and negative rates, our log file generated a maximum of unknown attack, these attacks are detected by filtering the traffic network and analysis of attacks by the agent of behavioral detection that uses the algorithm Clus-density collaborated by the honeyd which presents one point to attract hackers to our system and also to generate new signatures by the honeycomb the following table summarizes the results obtained at the level of the anomaly detection, false positive and negative.

TABELAU: TYPES OF ATTACKS DETECTED BY OUR SYSTEM

Type of attack	Normal	Abnormal	False positive	Falsenegative
IDS-AM-Clust [5-7]	51%	49%	4%	3%
Honeyd+honecomb+ IDS-AM-Clust	45%	55%	3%	2%

IV. CONCLUSION

This paper presents a combination of the new generation of IDS "IDS-AMS-CLUST" with a low interaction honeypot 'honeyd and honeycomb', the choice of this combination was therefore to detect all types of known and unknown intrusion and also update our IDS signature database by the double operation of the IDS-AM-Clust and honeycomb automatically.

The simulation results illustrate that the intrusion detection rate was increased and the rate of false positives and negatives have been reduced. This allows us to confimre the perfrmance and the effectiveness of the proposed system honeyd-IDS-AM-clust to enhance the system security.

ISSN: 2409-2770

REFERENCES

[1] L. Zpitzner, Honeypots: Tracking Hackers, Addison Wasley Professional, ISBN-10: 0321108957, (septembre 2002).

[2] Ashish Girdhar et Al : Comparative Study of Different Honeypots System, Volume 2, Issue 10 (August 2012), PP. 23-27.

[3] S. S. Muhammad, S. H. Choong, A Novel Architecture for Real-time Automated Intrusion Detection Fingerprinting using Honeypot, 27th KIPS Spring Conference, Korea, pp.1093-1095, (mai 2007).

[4] Bill Cheswick, "An Evening with Berferd: In Which a Cracker is Lured, Endured, and Studied." 1991.

[5] Chaimae Saadi, Habiba Chaoui and Hassan Erguig Security Analysis Using IDs Based on Mobile Agents and Data Mining Algorithms / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (1), 597- 602, 2015.

[6] Chaimae Saadi, Habiba Chaoui, Hassan Erguig, Contribution to Abnormality Detection by Use of Clust-Density Algorithm DOI: http://dx.doi.org/10.15866/irecos.v10i4.5699/**2015**

[7] Chaimae saadi and Habiba Chaoui, IDS based interaction on mobile agents and Clust-density algorithm IDS-AM-Clust curent accepted .

[8] Cohen, Fred. "Deception ToolKit". circa 2001 URL: http://www.all.net/dtk/dtk.html , March 13, 2003.

[9] J. Tian, J. Wang, X. Yang, R. Li, A Study of Intrusion Signature Based on Honeypot, Sixth International Conference on Parallel and Distributed Computing Applications and Technologies (PDCAT'05), pages 125 – 129, (2008).

[10] C, Chi, M. Li, D. Liu, A Method to Obtain Signatures from Honeypot Data, Lecture Notes in Computer Science, Volume 3222/2004, 435-442, DOI: 10.1007/978-3-540- 30141-7_61, (2004).

[11] Ram Kumar Singh : Intrusion Detection System Using Advanced Honeypots, (IJCSIS) International Journal of Computer Science and Information Security, Vol. 2, No. 1, 2009.

[12] S. Riebach, B. Toedtmann, E. Rathgeb. Combining IDS and Honeynet Methods for Improved Detection and Automatic Isolation of Compromised Systems, Computer Networking Technology Group, Institute for Experimental Mathematics, University Duisburg-Essen, Germany, (2006).

[13] C. Kreibich, J. Crowcroft, Honeycomb – Creating Intrusion Detection Signatures Using Honeypots, ACM SIGCOMM Computer Communication Review, 34, 51 – 56, (2004).

[14] C. Kreibich and J. Crowcroft. Honeycomb — Creating Intrusion Detection Signatures Using Honeypots 2nd Workshop on Hot Topics in Networks (HotNets-II), 2003, Boston, USA.

[15] Hatem Bouzayani : Modèle quantitatif pour la détection d'intrusion. Une architecture collaborative IDS-HONEYPOT (Juin 2012).

Authors

ChaimaeSaadi, is the Phd student of Computer Engineering at Ibn Tofail University – National High School of Applied Science (ENSA), Kenita – Morocco. Researcher at the Systems Engineering Laboratory. Her current research interest includes Cloud Computing Security, Evaluation of IDS using Mobile Agents and Data mining algorithms.

HabibaChaoui, is a Professor of Computer Engineering at Ibn Tofail University – National High School of Applied Science (ENSA),Kenitra - Morocco. She is the head of a Master program on information Systems Security. She is member of the Systems Engineering Laboratory and head of Data Analysis and Security Team.She conducts research in Cloud Computing Security, Evaluation of IDS using Mobile Agents and Data mining algorithms, Big Data and Data mining technology: Analysis, Security and Privacy.