International Journal on New Computer Architectures and Their Applications (IJNCAA) 1(1): 25-33
The Society of Digital Information and Wireless Communications, 2011 (ISSN: 2220-9085)

A Mutli-Agent System for Firewall Forensics Analysis

Hassina Bensefia[1, 2] and Nacira Ghoualmi[1]
[1] Networks and Systems Laboratory, Badji Mokhtar University
Sidi Amar, 23000 Annaba, Algeria
[2]Computer Science Department, Bachir El Ibrahimi University
Annaser, 34000 Bordj Bou Arreridj, Algeria
bensefia_hassina@yahoo.fr, Ghoualmi@yahoo.fr

ABSTRACT
Computer Forensics applies law to fight against unlawful and illegitimate use of computers and networks. It employs investigation methods to solve computer crimes. Knowing that the firewall is the unique input and output in a network, it is considered as the ideal location for recording network activities. The firewall log files trace all incoming and outgoing events in a network. Its content can include details about attacks and penetration attempts in the network. For this reason firewall forensics becomes a principal branch in computer forensics field. It uses the firewall log files content so as a source of evidence to lead an investigation in the aim to identify computer attacks. The investigation in firewall forensics consists of analyzing and interpreting the relevant information related to computer attacks which is contained in firewall log files. But the log files content is generally mysterious and difficult to decode. Its interpretation requires a qualified expertise. This paper proposes an intelligent system that automates the firewall forensics process and helps the security administrator to manage, exploit and interpret the firewall log files content. This system will assist the security administrator to make suitable decisions and judgments during the investigation step.

KEY WORDS
Firewall Forensics, Computer Forensics, Investigation, Evidence, Log files, Firewall, Multi-agent System

1. Introduction

The computer crime is a serious and spiny problem. Several organizations lost their productivity and reputation because of various direct and indirect attacks without any legal recourse. As a reaction to computer crime, forensic science was introduced in the computer security field to establish a judicial system able to resolve computer crimes and prosecute their perpetrators. Then computer forensics emerges as a new discipline in computer security field. It enables the collection of information from computer systems and networks and applies investigation methods to determine the information which proves that a given computer crime has occurred. This information is considered as evidence and could be submitted to the court of law [4]. Log files which are an important source of audit in a computer system trace all events that have occurred during the activity of a host. Their content can include details about any exceptional, suspected or unwanted event [3]. Then log files generated by different network components like servers, routers and firewalls are sources of evidence for computer forensics [5]. As the firewall is the single entry and exit point of a network, it represents the ideal location for recording all events in a network. Regarding its important role and position in the network, firewall forensics imposes itself as a branch in computer forensics field. The investigation in firewall forensics is based on the inspection and revision of firewall log files content which constitute a vital and necessary source of evidence. But log files content is huge and it has ASCII (American Standard Code for Information Interchange) format. It is mysterious to read and difficult to manage. Its interpretation requires knowledge related to the log file format itself and qualified skills in information, network administration, protocols, vulnerabilities, attacks and hacking techniques [3]. A security administrator is implied in security incidents. The review and inspection of log files is a basic daily task for him to maintain the security of a host or a network. This task is tedious, so difficult and takes much time. To help the security administrator to exploit firewall log files and conduct automatically the firewall forensics process, we propose a multi-agent intelligent system. The rest of the paper is organized as follows: Section 2 defines the computer forensics concept. Section 3 describes the log files and demonstrates their relevance to computer forensics and the difficulties encountered during their handling. Section 4 introduces the firewall forensics. Section 5 explains our adopted approach. Section 6 describes the architecture of the proposed system and details its components functioning. Section 7 gives a preview on the implementation of the system and its execution results. Section 8 summarizes our conclusion and future work.

2. Computer forensics

Computer forensics is an emergent science in computer security field [1]. It applies law to illegitimate use of computer systems in the aim to solve the computer crime and make it admissible in a tribunal [3]. The computer forensics process consists first of collecting data from computer systems and network components. Then it employs an investigation to retrace malicious events and identify attacks. The finality is to discover the identity of the attacker and obtain accusatory judicial evidence [3]. The evidence is the set of data that traces systems and networks activities and can confirm or refute the attack occurrence [1]. The evidence depends on attack type and may exist in three main locations: the victim system, the attacker system or in the network components which are situated between the victim system and the attacker one.

The investigation is an important step in computer forensics. It is a procedure that allows solving computer attack after it has occurred [2]. It analyzes the collected information to verify if an attack has occurred. The investigation can determine the intrusion time, the attack type, the techniques used to accomplish the attack, the attack author, the traces that he left behind him, the penetrated systems and the path borrowed by the attacker [1]. Its objective is to provide the sufficient judicial evidence to prosecute the attack author.

3. Log files

Logging is a functionality which is ubiquitous in the majority of the modern operating systems. Unix supports the Syslog and Windows supports the Event log which are two logging utilities. Logging records all the events happening in a host during the execution of an application or a network service such as a mail server, a web server or a Domain Name Server (DNS). The recording takes the form of a file called log file whose content depends on the recording level and the host activity [7]. The log file has an ASCII (American Standard Code for Information and Interchange) structure. Each input in the log file represents a line which designates a request received by the host, the host response and the processing time of this request. It included elementary information fields. Each field indicates information related to the request like the host IP address, the date and time of the request submission and the used protocol. So the log file reports all the host TCP/IP incoming and outgoing packets. The relevant information in log files content has major interest during the resolution of an attack [4]. Indeed the source IP address may reveal the attacker identity and the destination IP address may reveal the target victim system. The payload in a TCP/IP packet identifies the attack type. Therefore the log files can trace any suspected activity. They may include the attacker track after his penetration into a host or a network. Thus log files are an important source of evidence for computer forensics [3][12][13]. When an attack has occurred, the information contained in log files must be carefully analyzed during the investigation step in order to obtain accusatory evidence.

4. Firewall Forensics

The firewall is a vital element for the security of a private network [7][8]. It is placed at the drop-off of the private network and internet. It implements an access control policy for the TCP/IP traffic exchanged between the two networks. All the packets exchanged between the private network and internet must imperatively pass through the firewall in order to be filtered according to the implemented access control policy. This policy consists of filtering rules which examine all the incoming and outgoing TCP/IP packets individually in the aim to allow or deny their transit by the firewall.

The Firewall is the single entry and the exit point of a

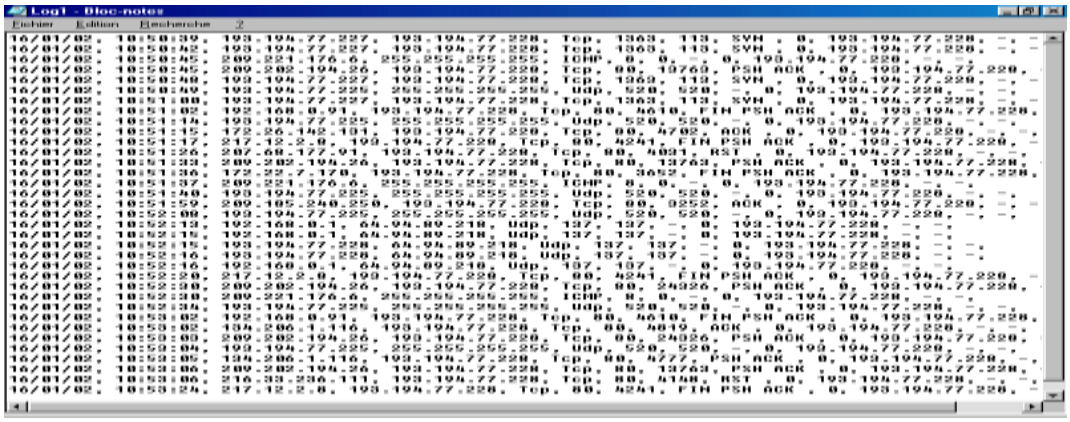

Figure 1. Extract of Microsoft proxy server 2.0 log file

network. So it represents the ideal location for recording the network activities. The firewall log files report all the network incoming and outgoing activities. They can give details about the TCP/IP traffic passing cross the firewall and the malicious activities happening in the network. Then the relevant information contained in firewall log files is an indispensable source of evidence for the investigation and a tool to discover computer crimes. As a consequence firewall forensics was introduced in computer forensics as new axis[5]. We define firewall forensics as the collection and analysis of data contained in firewall log files to identify the network penetration attempts and determine attacks targeting a network protected by a firewall [7].

Figure 1 shows an extract of the log file content of Microsoft proxy server which is a firewall acting as an application gateway. We give significance of the first input of this log file which is:
16/01/02, 10:50:39, 193.194.77.227, 193.194.77.228, TCP, 1363, 113, SYN, 0, 193.194.77.228, -,-

- 16/01/02: is the reception date of the TCP/IP packet.
- 10:50:39: is the time of the TCP/IP packet reception.
- 193.194.77.227: is the source IP address of the system which sends the TCP/IP packet.
- 193.194.77.228: is the destination IP address of the system which will receive the packet.
- TCP: is the protocol used to transmit the TCP/IP packet.
- 1363: is the source port indicating the ongoing application on the system which has sent the packet.
- 113: is the destination port indicating the executing application on the system which will receive the packet.
- SYN: is the value of the TCP flag which indicates the establishment of connection.
- 0: this field indicates the result of the proxy filtering rule. If its value is 0, it indicates the TCP/IP packet is rejected. If it is 1, the TCP/IP packet is accepted.
- 193.194.77.228: is the IP address of the gateway which has received the TCP/IP packet.
- -: is an empty field.

5. The proposed approach

Our objective is to help the security administrator to automatically exploit firewall log files and carry on the firewall forensics process in real time. We decompose the global process of firewall forensics into four main enchained steps which are partially parallel:

1. Collection: this step allows the collection of only the relevant information contained in firewall log files.
2. Inspection: it analyzes the collected information to check whether suspected events exist or not.
3. Investigation: it determines the significance of any suspected event to confirm if the event is malicious or normal behavior.
4. Notification: if the event is malicious, this step will generate a detailed report about the investigation result which will be transmitted to the security administrator.

There is no standard format for firewall log files. Each firewall generates log files in a proprietor format. So the collection step requires expertise to understand the firewall log files format. The inspection step also requires expertise to discover suspected events in firewall log files content. To determine the significance and the aim of a suspected event, the investigation step involves a qualified knowledge. A multi-agent system [10] [11] will be the most suitable approach to automate the firewall forensics process. We employ cognitive agents. Our motivation is justified by the diversity of expertise required in the three main phases of the firewall forensics process. The agents can collaborate in order to contribute to the forensics process which is a complex problem beyond their individual capacities and knowledge. This collaboration is expressed by exchange of information between the agents. Likewise, partial parallelism is needed between the phases of the firewall forensics complex process.

A multi-agent system is an artificial intelligence approach [10][11]. An agent is a real or virtual entity that has a partial representation of its environment. It acts on itself or on its environment. It can communicate with other agents. A community of agents that coexist and interact with each other in a common environment designates a multi-agent system. They collaborate to resolve a complex problem beyond their capacities and knowledge. This collaboration is expressed by information communication between the agents

We propose a multi-agent system for the firewall forensics process which consists of three cognitive agents:

- The collector agent: it is dedicated for the collection step. Its role is the collection and the processing of the firewall log files content.
- The inspector agent: it is dedicated for the inspection step. It identifies suspected events in the collected firewall log files content. This agent must transmit any suspected event to the investigator agent.
- The investigator agent: it is dedicated for the two main steps: investigation and notification. This agent has to check the suspected event and determine its significance and objective in order to confirm or refute the occurrence of attack. If any attack is confirmed, the investigator agent generates a detailed report and sends it to the security administrator as a security alert.

6. Architectural and functional model

Figure 2 illustrates the global architecture of our proposed system. Considering a private network connected to internet which is protected by a firewall. The firewall logging functionality is activated to daily generate log files in a specific format which is proprietor to the deployed firewall. Our proposed system proceeds by the rotation of the ongoing log file at regular time intervals which results of an instantaneous copy of the ongoing log file. The collector agent reads the instantaneous log file copy. It takes into account only the packets that have been accepted by the firewall. Then it extracts the important fields of every accepted activity and saves them in a data base called activity base. The inspector agent inspects the activity base to identify suspected events and send them to the investigator agent. This latter determines the signification and the objective of the suspected activity. If the suspected activity is confirmed as a malicious activity, the investigator develops a detailed report about this activity and sends it to the security administrator as a security alert. All the reports generated by the investigator are saved in a data base called archives base. Our system includes two interfaces. The user interface allows the interaction between the security administrator and the system. The expert interface allows experts to update the knowledge of the agents.

As follows, we will give a detailed description of our system components and show the agents reasoning and communication.

6.1 Collector agent

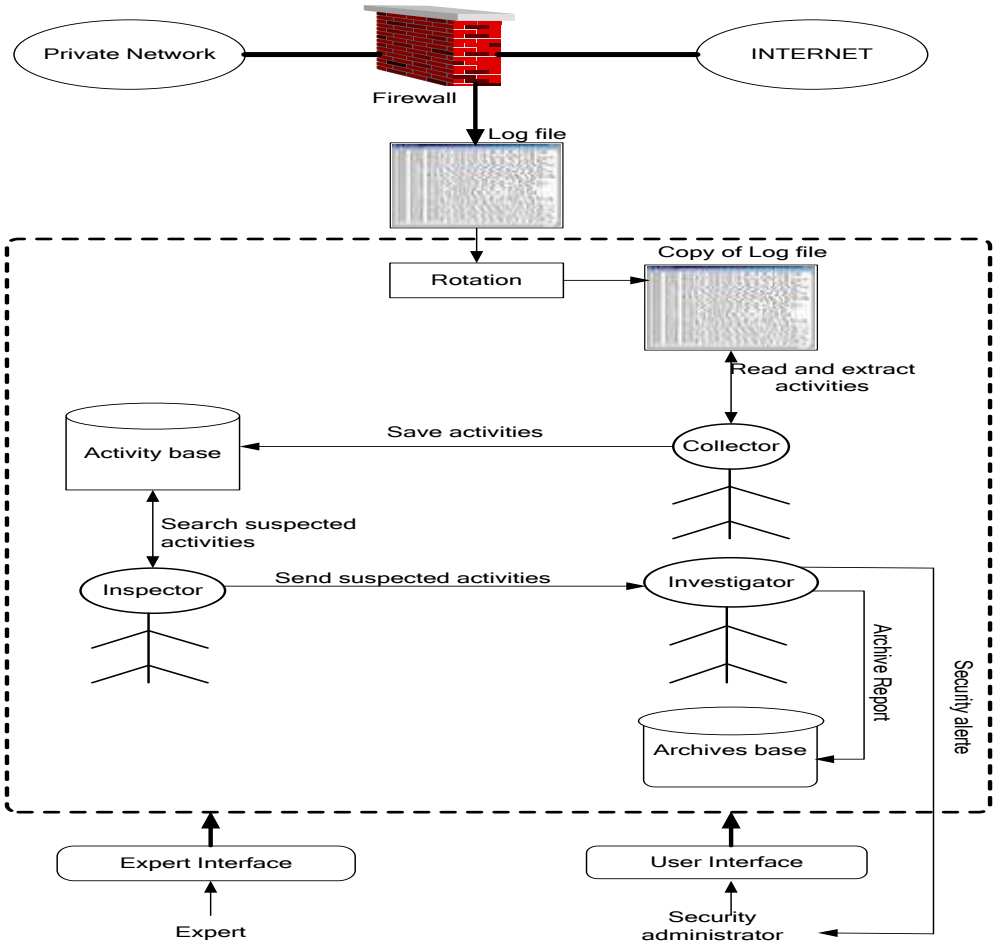

Figure 2. Architecture of the proposed system

The collector is a cognitive agent having knowledge base and inference engine. The knowledge base includes the knowledge related to log files format of the most used firewalls like Firewall-1 and Cisco Pix since there is no standard format for firewall log files. The inference engine represents the brain of the collector agent. It uses the knowledge base to read and process the content of the log file copy resulting from rotation.

Every input in firewall log file content designates an incoming or an outgoing TCP/IP packet passing through the firewall. It includes information about the packet like: date, time, the used protocol like TCP (Transmission Control Protocol) or UDP (User Datagram Protocol) or ICMP (Internet Control Message Protocol), the source IP address, the destination IP address, the source port, the destination port and the result of the firewall filtering rule which has to accept or reject the packet. The collector treats only the log file inputs related to the accepted packets. It extracts the important fields as date, time, protocol, source IP address, destination IP address, source port and destination port. Date and time indicate when the packet has arrived to the firewall. Protocol, source IP address, destination IP address, source port and destination port are the essential elements in a communication. The firewall filters TCP/IP packets while taking consideration of these elements. So the interpretation of any log file input depends on the significance of the essential elements of communication which means the determination of the purpose to be achieved by the communication. We consider the extracted essential communication elements as a record that we called activity. So the collector saves this record in the activity base. The reasoning of the collector agent follows these steps:

1. Take a copy of the firewall log file.
2. Read the input of the firewall log file copy.
3. If the packet is rejected by the firewall, go to step 2.
4. If the packet is accepted by the firewall, extract the essential elements of communication according to the log file format of the deployed firewall.
5. Save the extracted elements (activity) in the activity base.
6. If it is the end of log file copy go to step 1 else go to step 2.

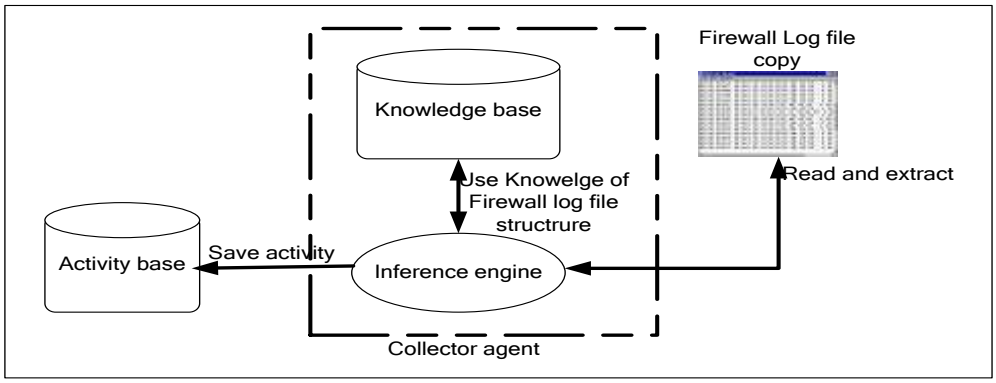

Figure 3. Architecture of the collector agent

6.2 Activity base

Activity number	Activity nature	Date	Time	IP Source	IP Destination	Protocol	Source Port	Destination Port

Figure 4. Activity base structure

Each record in this data base summarizes a TCP/IP packet accepted by the firewall. It includes essential elements of communication already specified which form an activity. We propose this data base to facilitate the inspection of the firewall log file content which is a difficult operation to do on the log file copy. Every activity base record is composed of the following fields: activity number, activity nature and the communication elements which are: date, time, protocol, source IP address, destination IP address, source port and destination port. The Activity number is an integer that acts as the identifier of the activity. It will be incremented at every activity insertion in the activity base. The activity nature field will contain the character string "NOR" if the activity is normal. Else if the activity is suspected and may be malicious, the activity nature field will be "MAL". This field must be filled up by the inspector agent after inspecting the activity.

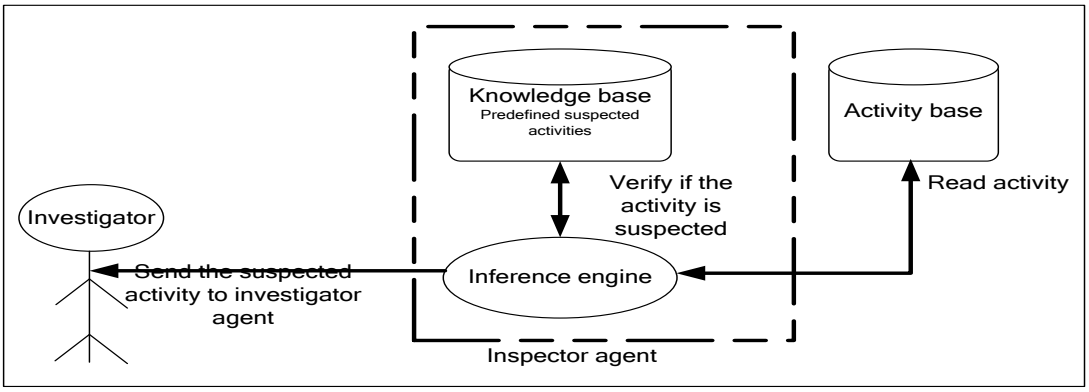

Figure 5. Architecture of the inspector agent

6.3 Inspector agent

It is a cognitive agent that integrates knowledge base and inference engine. The Knowledge base includes the knowledge about all the threats that can involve one or more than one element of the five essential communication elements: source IP address, destination IP address, source port, destination port and protocol. The inspector uses this knowledge to inspect firewall log files content. To create the inspector knowledge base, we use a concise document which is written by Robert Graham titled Firewall Forensics (What am I seeing?) [6]. This document gives the significance of the port numbers, IP addresses and ICMP messages that can be often observed by firewall users in firewall log files content. Then the inspector knowledge base contains what we call the predefined suspected activities related to one or more than one of the five essential communication elements.

The inference engine is the brain of the inspector agent. It exploits the predefined suspected activities to inspect the activity base records. When an activity is inspected as a suspected activity, it will be automatically sent to the investigator agent. The reasoning of the inspector agent respects the following steps:

1. Access to the activity base record in a sequential order.
2. Compare the fields of the activity base record to the predefined suspected activities fields.
3. If the activity is normal, the inspector will mark the field activity nature with "NOR".
4. If the activity is suspected, the inspector will mark the field activity nature with "MAL" and will send the suspected record to the investigator agent.
5. Go to step 1.

6.4 Investigator agent

It is a cognitive agent which is endowed with knowledge base and inference engine. The Knowledge base contains the knowledge related to the interpretation of the firewall log files content. For conceiving this knowledge base, we exploit the document written by Robert Graham entitled FAQ: Firewall Forensics (What am I seeing?) which explains the significance of some port numbers and IP addresses [6].

The investigator knowledge base includes 112 production rules.

We give examples of some rules:

- Rule 1: IF {Protocol= TCP and Destination port=0} THEN {Attempt to identify the operating system}.
- Rule 2: IF {Protocol= UDP and Destination port=0} THEN {Attempt to identify the operating system}.

- Rule 3: IF {Protocol=UDP and Source port=68 and Destination address=255.255.255.255 and destination port=67} THEN {Response of a DHCP server to the request of a DHCP client}.
- Rule 4: IF {Protocol= TCP and Destination port=7} THEN {Connection to the TCPMUX service of an IRIX machine}.

Being the brain of the investigator agent, the inference engine exploits the knowledge base to interpret any suspected activity transmitted by the inspector agent. If the suspected activity is a malicious action, the inference engine will generate a report including details about this malicious activity and sends it as a security alert to the security administrator. This report will be stored in a data base called archives base.

This is the reasoning followed by the investigator agent:

1. Receive the suspected activity transmitted by the inspector agent.
2. Research the applied rules in the knowledge base.
3. Execute the selected rules to obtain the interpretation of the suspected activity.
4. If the interpretation indicates a malicious activity then generate a report including the malicious activity and its interpretation.
5. If the interpretation indicates a normal activity, then send a message to the inspector agent to mark the activity as normal "NOR" in the activity base and go to step 1.
6. Send the generated report as a security alert to the security administrator.
7. Save the generated report in the archives base.
8. Go to step 1.

6.5 Archives base

This base gathers all the reports generated by the investigator agent during a year. The structure that we propose for the archives base consists of three linked tables. The first table indexes the month in the year. The second table indexes the day in the month. The third table contains the reports generated which are indexed by day. We have adopted this structure to help the security administrator to interrogate the archives base in a late time.

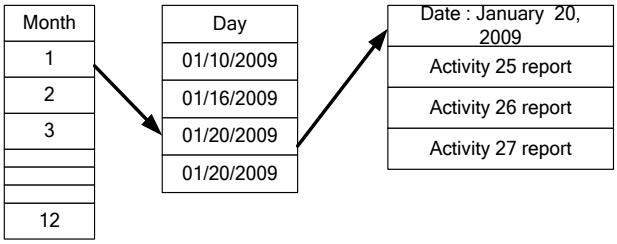

Figure 7. Structure of the archives base

6.6 User Interface

Our system user interface will be used by the security administrator for:

- Introducing any activity composed of at least one of the 5 essential communication elements in order to determine whether its nature is normal or suspected and obtain the its interpretation.
- Interrogating the archives base.

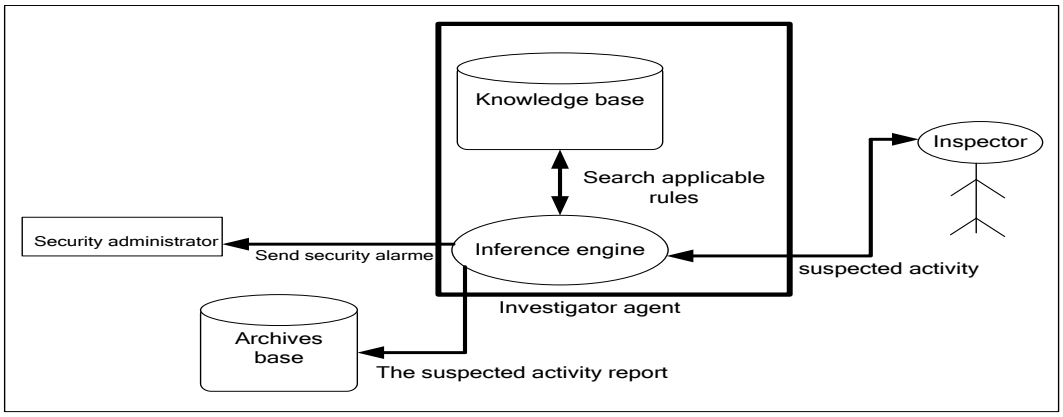

Figure 6. Architecture of the investigator agent

6.7 Expert Interface

This interface allows the experts to manage the knowledge of the agents like:

- Introducing knowledge related to firewalls log file formats.

- Inserting new rules in the knowledge base of the investigator agent.
- Introducing new predefined suspected activities in the inspector knowledge base.

6.8 Communication between agents

The collector agent communicates with the inspector agent by sharing the information existing in the activity base. Then we use the blackboard model as a mean of communication between the collector agent and the inspector agent. When the collector puts down an activity, the inspector inspects it by determining if it is a suspected activity or not. The inspector agent and the investigator do not share a common information zone. So we employ the actor model in order to make them communicate. The two agents will communicate by sending messages. Figure 8 and figure 9 give a view of the models adopted respectively for the communication between the collector agent and the inspector agent and the communication between the inspector and the investigator agent.

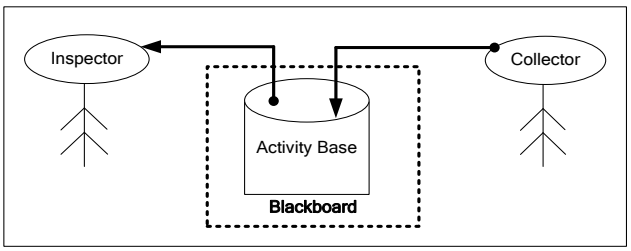

Figure 8. Communication between the collector agent and the inspector agent

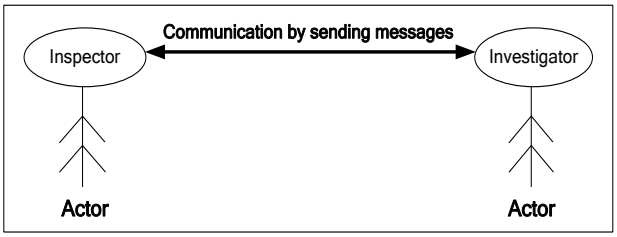

Figure 9. Communication between the inspector agent and the investigator agent

7. Implementation and results

We have implemented the proposed system with Java language because it offers many advantages like the object oriented programming, multitasking application and multiplatform portability. To show the ability of our implemented system in analyzing, inspecting and investigating firewall log files, we give some execution results through this short extract of Microsoft proxy server 2.0 log file which is described in figure 10

```
04/11/08, 03:36:52, 136.199.55.156, 193.194.77.225, ICMP, 8, 0, -, 0, 193.194.77.228, -, -,
04/11/08, 03:36:55, 136.199.55.156, 193.194.77.225, Udp, 520, 520, -, 0, 193.194.77.228, -, -,
04/11/08, 03:36:58, 204.29.239.23, 193.194.77.222, Tcp, 1240, 53, -, 1, 193.194.77.228, -, -,
04/11/08, 03:37:08, 193.194.77.222, 204.29.239.23, Tcp, 53, 1240, -, 1, 193.194.77.228, -, -,
04/11/08, 03:37:10, 130.79.68.209, 193.194.77.227, Tcp, 3125, 23, -, 0, 193.194.77.228, -, -,
04/11/08, 03:37:14, 216.33.236.111, 193.194.77.226, Tcp, 1896, 1, -, 0, 193.194.77.228, -, -,
04/11/08, 03:37:23, 193.194.23.121, 193.194.77.229, Udp, 1132, 22, -, 1, 193.194.77.228, -, -,
04/11/08, 03:37:30, 134.206.1.116, 193.194.77.228, ICMP, 8, 0, -, 0, 193.194.77.228, -, -,
04/11/08, 03:37:43, 134.206.1.116, 193.194.77.228, ICMP, 8, 0, -, 0, 193.194.77.228, -, -,
04/11/08, 03:37:56, 134.206.1.116, 193.194.77.228, ICMP, 8, 0, -, 0, 193.194.77.228, -, -,
04/11/08, 03:38:01, 0.0.0.0, 255.255.255.255, Udp, 67, 68, -, 1, 193.194.77.228, -, -,
04/11/08, 03:38:08, 193.194.77.220, 255.255.255.255, Udp, 68, 67, -, 1, 193.194.77.228, -, -,
04/11/08, 03:38:10, 193.194.78.35, 193.194.77.224, Udp, 1234, 0, -, 1, 193.194.77.228, -, -,
04/11/08, 03:38:15, 193.194.75.190, 193.194.77.225, Tcp, 1526, 11, -, 1, 193.194.77.228, -, -,
04/11/08, 03:38:18, 193.194.75.190, 194.193.77.225, Tcp, 1752, 98, -, 1, 193.194.77.228, -, -,
04/11/08, 03:39:23, 193.194.77.225, 255.255.255.255, Udp, 138, 138, -, 0, 193.194.77.228, -, -,
04/11/08, 03:39:37, 193.194.78.35, 193.194.77.228, Tcp, 1768, 80, SYN, 0, 193.194.77.228, -, -,
04/11/08, 03:39:53, 193.194.68.20, 193.194.77.226, Tcp, 143, 143, -, 0, 193.194.77.228, -, -,
04/11/08, 03:39:53, 193.194.68.20, 193.194.77.226, Tcp, 110, 110, -, 0, 193.194.77.228, -, -,
04/11/08, 03:39:53, 193.194.68.20, 193.194.77.226, Tcp, 25, 25, -, 1, 193.194.77.228, -, -,
04/11/08, 03:39:58, 80.89.196.27, 255.255.255.255, Tcp, 4998, 80, SYN, 0, 193.194.77.228, -, -,
04/11/08, 03:40:11, 193.194.242.145, 193.194.77.230, Tcp, 3240, 1243, -, 1, 193.194.77.228, -, -,
04/11/08, 03:40:33, 64.94.89.218, 193.194.77.228, ICMP, 8, 0, -, 0, 193.194.77.228, -, -,
04/11/08, 03:40:46, 169.254.1.22, 193.194.77.222, Udp, 161, 161, -, 1, 193.194.77.228, -, -,
04/11/08, 03:41:10, 193.194.77.225, 255.255.255.255, Udp, 520, 520, -, 0, 193.194.77.228, -, -,
```

Figure 10. A short extract of Microsoft Proxy Server 2.0 log file

The collector agent reads the log file inputs. It extracts only the important fields related to the packets which have been accepted by the firewall and stores them as records in the activity base. The inspector agent inspects the activity base records. If the record presents a normal activity, it fixes its activity nature field with "NOR". If the record is suspected as malicious activity, the inspector agent sets its activity nature field as "MAL" and sends this record to the investigator agent. Figure 11 gives a snapshot of the activity base content.

Activity number	Activity nature	Date	Time	IP Source	IP Destination	Protocol	Source Port	Destination Port
01	NOR	04/11/08	03:36:58	204.29.239.23	193.194.77.222	Tcp	1240	53
02	NOR	04/11/08	03:37:08	193.194.77.222	204.29.239.23	Tcp	53	1240
03	MAL	04/11/08	03:37:23	193.194.23.121	193.194.77.229	Udp	1132	22
04	MAL	04/11/08	03:38:01	0.0.0.0	255.255.255.255	Udp	67	68
05	NOR	04/11/08	03:38:08	193.194.77.220	255.255.255.255	Udp	68	67
06	MAL	04/11/08	03:38:10	193.194.78.35	193.194.77.224	Udp	1240	0
07	MAL	04/11/08	03:38:15	193.194.75.190	193.194.77.225	Tcp	1526	11
08	MAL	04/11/08	03:38:18	193.194.75.190	194.193.77.225	Tcp	1752	98
09	MAL	04/11/08	03:39:53	193.194.68.20	193.194.77.226	Tcp	25	25
10	MAL	04/11/08	03:40:11	193.194.242.145	193.194.77.230	Tcp	3240	1243
11	MAL	04/11/08	03:40:46	169.254.1.22	193.194.77.222	Udp	161	161

Figure 11. The activity base records

The investigator agent uses its rule base to undertake reasoning about the malicious records. It gives an interpretation of each record in the aim to confirm or refute the inspector decision. Table 1 displays the results of the investigator reasoning which demonstrate that all the records are malicious activities except activity number 04 which is a normal activity. In general the IP source address 0.0.0.0 is a tampered address but according to the investigator reasoning when this address is used with IP destination address 255.255.255.255 and Udp protocol and respectively the source and destination ports 67 and 68 it indicates a request sent by a DHCP client to a DHCP server. When a DHCP client starts, it has not IP address. It uses 0.0.0.0. as source IP address to send a request to the network on the port 68.

Table 1. Investigator reasoning results

Activity number	Investigator reasoning results
03	Request for remote access and control of the system.
04	Request sent by a DHCP client to a DHCP server.
06	Attempt to identify the operating system.
07	Request to list the active processes on a Unix machine.
08	Connection to linuxconf of a Linux machine.
09	Attempt to scan the SMTP service by Sscan.
10	Remote access to the Trojan horse Sub-7.
11	The IP source address is tampered.

8. Conclusion and future work

Our proposed system represents an intelligent tool which has the following strong points:

- Managing and exploiting the voluminous and mysterious firewall log files content.
- Identifying suspected activities in the mass of information contained in firewall log files.
- Interpreting and notifying any confirmed malicious activity.
- Summarizing all the TCP/IP packets passing through the firewall in the activity base. This data base can help the security administrator to study the network activity and make statistics about the nature of traffic passing through the firewall.
- Archiving detailed reports about all malicious activities in the archives base. This data base is well structured and it can be easily interrogated in an offline mode by the security administrator.

Our proposed multi-agent system can accomplish the firewall forensics process automatically in real time thanks to the expertise instituted in the cognitive agents. The system results are useful for the security administrator to take the best decisions and achieve successfully the investigation step. As perspective, we envisage:

- Extending the knowledge base of the collector agent by the knowledge corresponding to the format structure of other current main firewalls like Juniper, Fortinet and Sonic Wall.
- Enriching the Knowledge of the inspector agent and the investigator agent in the aim to expand the coverage of malicious activities and improve our system analysis results.
- Exploit the archives base to study the behavior of attackers and define their motivations, purposes and intrusion methods in the aim to create attackers profiles. This information is useful in security incident response.

References

1. Bensefia, H.: Fichiers Logs : Preuves Judiciaires et Composant Vital pour Forensics. Review of Scientific and Technical Information (RIST), Vol. 15, n°01-02, pp. 77-94, (2005)
2. Carrier, B., Spafford, E. H, Getting physical with the digital investigation process, International Journal of digital evidence, Vol. 2, Issue. 2, (2003)
3. Yasinsac, A., Manzano, Y.: Policies to Enhance Computer and Network Forensics. Workshop on information assurance and security, United States Military Academy, West Point, pp. 289-295, NY (2001)
4. Sommer, P.: Digital Footprints: Assessing Computer Evidence, Criminal Law Review, Special Edition, pp. 61-78, (1998)
5. Casey, E.: Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet. Book review, Academic Press, San Diego, California, (2000)
6. FAQ: Firewall Forensics (What am I seeing?), http://www.capnet.state.tx.us/firewall-seen.html, last visit October (2010)
7. Bensefia, H.: La conception d'une base de connaissances pour l'investigation dans Firewall Forensics. Master thesis. Centre of Research in Technical and Scientific Information, Algeria (2002)
8. Lodin,W., Schuba, L.: Firewalls fend off invasions from the net. IEEE spectrum. Vol. 35, Issue. 2, (1998)
9. Chown, T., Read, J., DeRoure, D.: The Use of Firewalls in an Academic Environment. JTAP-631, Department of Electronics and Computer Science. University of Southampton, (2000)
10. Ferber, J.: Introduction aux systèmes multiagents. InterEditions, (2005)
11. Boissier, O., Guessoum, Z.: Systèmes Multi-agents Défis Scientifiques et Nouveaux usages. Hermès, (2004)
12. Murray, C. P.: Network Forensics. University of Minnesota, Morris,(2000)
13. Sommer, P.: Downloads, Logs and Captures: Evidence from cyberspace. Computer Journal of Financial Crime, pp. 138-152, (1997)