Network Security Management with Intelligent Agents

K. Boudaoud, H. Labiod
Institut EURECOM
B.44P. 193
06904 Sophia-Antipolis France
Phone: (33) 4 93 00 26 38
Fax: (33) 4 93 00 26 27
{boudaoud,labiod}@eurecom.fr

R. Boutaba
ECE Department
University of Toronto
10 King's College Road
Toronto, Ontario, M5S 3G4, Canada
Tel: +1 416 964 3063
Fax: +1 416 978 0804
Email: rboutaba@comm.utoronto.ca

Z. Guessoum
LIP6-OASIS
Case 169, 4 place Jussieu,
75252 PARIS Cedex 05 France
Phone: (33) 1 44 27 87 43
Fax: (33) 1 44 27 70 00
Zahia.Guessoum@lip6.fr

Abstract

Multi-Agent Systems technology can be useful for efficiently designing and maintaining secure networks. Indeed, networks evolve at a rapid pace in terms of the number and type of components and user access queries as well as intrusion possibilities. Features such as autonomy, adaptability and flexibility of the "intelligent" agent paradigm allow managing network evolution in a controlled way. The focus of our work concerns one critical security management issue that is *intrusion detection*. We propose a novel approach called IA-NSM (Intelligent Agents for Network Security Management) for intrusion detection using intelligent agent technology. IA-NSM provides a flexible integration of a multi-agent system in a classical networked environment to enhance its protection level against inherent attacks.

Keywords

Network security management, intrusion detection, network attacks, distributed management, intelligent agents

1. Introduction

The number of users using networks is increasing exponentially every day. As a consequence, many users try to access private networks. The latter are so threatened by malicious attacks. The protection of networks is, therefore, more than useful, it is vital. This problem requires the monitoring of real distributed hosts, the various events and exchanges between these hosts. The complexity of this problem makes the use of multi-agent system necessary. The aim of this paper is to propose a multi-agent system to model the network security management, particularly the network intrusion detection.

The existing solutions for network security management are very complex and costly. What needed is a flexible, adaptable and affordable security solution, which provides greater autonomy. Therefore, it is necessary to review the way in which standard intrusion detection is designed and performed to identify and to alleviate its weakness. In this context, multi-agent systems provide a balance between security requirements, system flexibility and adaptability.

Actually, intelligent agent technology is viewed as one of the fastest growing areas of research and new applications development in telecommunications. The DAI (Distributed Artificial Intelligence) concept [1] consists of a group of individual named agents that have distributed environments. Each agent cooperates and communicates with other agents. Combined knowledge and experience of the agent with the information coming from neighboring agents permits the agent to make the best (optimum in some sense) decision. In this paper, we suggest to improve network security management by integrating DAI approach based on multi-agent system technique in Intrusion Detection Systems (IDS). We propose a new approach based on providing the NSM (Network Security Management) hosts with additional functionalities. These entities become more intelligent, capable of making various decisions with autonomy to detect intrusions and to overcome their bad effects. The introduction of multi-agent system (MAS) in a network seems so promising to embed adaptive features thereby enabling network entities to perform adaptive behavior and becoming “intelligent”. The term “intelligence” is used in the sense that network entities provide reasoning capabilities, exhibit behavior autonomy, adaptability, interaction, communication and co-operation in order to reach some goals. Therefore, we built a new architecture called IA-NSM (Intelligent Agents for Network Security Management). It is used to provide a flexible integration of multi-agent technique in a classical network to enhance its protection level against inherent attacks. To implement our agents, we use an operational multi-agent simulation platform named DIMA [10]. DIMA realizes an integration of a generic agent architecture and a discrete event simulation framework. It provides us with a good tool to study a collection of interacting agents in a dynamic network system.

Our paper is organized as follows. Section 2 provides an overview on network security management and presents a short description of network attacks and Intrusion Detection Systems (IDS). The agent concept and MAS technique are outlined in section 3. In section 4, a distributed architecture integrating a multi-agent system for network security management and particularly for network intrusion detection is described. The use of DIMA platform to realize our framework is also described. Finally, Section 5 provides concluding remarks and some future work.

2. Network Security Management

Security management aims to maintain the integrity, confidentiality and availability of systems and services. The increasing number of people, organizations, and enterprises, which install and subscribe to the Internet, makes the security management an important issue. It is therefore necessary to identify the risks of attacks that the networks are exposed to. Applying security management is a two-fold activity: 1) the security architecture is to be deployed to protect networks by detecting attacks; and 2) when attacks are detected the security architecture deals with these attacks in real time by taking security measures.

2.1. Network Attacks

An attack can be defined as any non-standard activity which:

- breaks privacy rules, compromising the information *confidentiality*,
- alters information, compromising the data *integrity*,
- makes a network infrastructure unavailable or unreliable, compromising the *availability* of a resource. In this case, we speak about *denial of service* attacks.

The various attacks can be classified in two categories:

- ***External attacks:*** they are generated from the outside by a hacker who is trying to access to a network to have information, have fun, or trying any kind of denial of service attack.
- ***Internal attacks***: they are generated by internal users. These users abuse of their rights and privileges to do unauthorized activities and to obtain unauthorized access.

Another kind of classification is the following:

- ***Network attacks***: they aim to prevent users from using the network resources or making the network services unavailable. They may also monitor network traffic for analyzing it and for collecting pertinent information.
- ***System attacks***: the purpose of these attacks is to compromise the system, like modifying or deleting critical files (for example, password file).
- ***Web attacks***: an example of these attacks is the modification of site's web page by a hacker in order to discredit it or simply to make it ridiculous.

In this paper, we are interested mainly in ***Network attacks***, such as:

- ***IP spoofing****:* Consists of sending packets with a faked IP source address. Thus, the user can believe that packets were originated from another host, preferably a host that is allowed to establish connections with the attacked host, if the real sender (attacker) itself is not allowed.

- ***TCP SYN flooding**:* the purpose of this attack is to constantly fill the backlog queue of a host, where incoming connection requests are kept, by sending a bulk of SYN requests. The attacking host must spoof the IP address of an unreachable host for the server so the SYN/ACK answers will never be received and ACK messages never generated. The consequences of TCP SYN flooding is that all further requests to this TCP port will be ignored. In some cases, the attacked host may even exhaust memory and crash.
- ***ICMP flooding**:* ICMP packets (usually ping requests but other type of requests are also possible) can be used to flood a network and bring it down. Requests must be sent at a high rate to many host destinations. Concurrent answers generate many collisions on local area networks and fill routers queues.
- ***Doorknob rattling**:* repetitive attempts to log in to several hosts with any user-id/password combination in order to obtain an access to an account.
- ***Traffic Analysis**:* information is leaked to unauthorized entities, through observation of communications traffic patterns.

2.2. Network Intrusion Detection

2.2.1 Intrusion Detection

Securing a network involves protecting it against all possible attacks. But, in practice it is not possible to have a completely secure network. So, the problem is how to detect in real time security violations.

Intrusion detection is a practical approach for enhancing the security of computer and network systems. The goal of IDS is to detect attacks especially in real-time fashion. These systems use one or both approaches of intrusion detection:

- the ***behavior-based intrusion detection*** approach, which discovers intrusive activity by comparing the user or system behavior with a normal behavior profile;
- the ***knowledge-based intrusion detection*** approach, which detects intrusions upon a comparison between parameters of the user's session and known pattern attacks stored in a database.

The ***behavior-based intrusion detection*** approach allows detecting unknown intrusions contrarily to the ***knowledge-based intrusion detection*** approach, which detects well-known intrusions.

According to the kind of data, an IDS uses to detect intrusive activity, we distinguish three types of IDS:

- ***Host-based*** **IDS**, which are designed to monitor a single host. They use their own host Operating System's audit trail as the main source of input for detecting intrusions.

- ***Distributed Host-based* IDS**, which are in charge of monitoring several hosts. They perform intrusion detection using Operating System's audit trail or information from multiple monitored hosts. This information is processed on a central site.
- ***Network-based* IDS**, which analyze traffic on a LAN to detect intrusive behavior.

2.2.2 Examples of existing IDS

Several IDS have been proposed such as:

- **NADIR** (Network Anomaly Detection and Intrusion Reporter) was designed for the Los Alamos National Laboratory's Integrated Computing Network (ICN). This network is divided into four partitions; each partition is defined to process at a different security level [2]. Special nodes, called *service nodes,* link these partitions. **NADIR** uses audit records coming from these nodes to perform the analysis of network activity. The main disadvantage of **NADIR** is that it uses a non-standard network protocol. So, it can not be easily ported to a heterogeneous environment.
- **DIDS** (Distributed Intrusion Detection System) is a project developed jointly by UC Davis, Lawrence Livermore National Laboratory, Haystack Laboratory and the US Air Force [2]. It was designed to monitor a local area network (LAN). It is a distributed host-based intrusion detection system. It is constituted by three components:

    - a *DIDS director*: is responsible of analyzing all data received from the two other components and detecting possible attacks;
    - a *LAN monitor*: it monitors all traffic on a LAN segment and reports to the DIDS director unauthorized or suspicious activities on the network ;
    - a series of *Host Monitor*: each of them monitors a single host. It collects audit data and analyze them. The relevant information is then transmitted to the DIDS director.

- **CSM** was developed at Texas A&M University. It was designed to perform intrusion detection in a distributed network environment. More particularly, CSM aims to detect suspicious activities without the use of an established centralized director [3]. In fact, each CSM performs intrusion on its own system and communicates with the other CSM in order to detect cooperatively intrusive activity. This cooperation allows CSMs to handle some kinds of attacks like *Doorknob Rattling* attack in a proactive manner, instead of reactive. CSM is constituted by six components:

    - a *Local Intrusion Detection Component* (IDS): it performs intrusion detection for the local host;

- a *Security Manager* (SECMGR): it co-ordinates the distributed detection intrusion between CSMs;
- an *Intruder Handling* component (IH): its role is to take actions when an intruder is detected ;
- a *Graphical User Interface* (GUI): it permits the security administrators to interact with individual CSMs;
- a *Command Monitor* (CMNDMON): it captures commands executed by users and send them to the IDS ;
- a *TCP Communication* (TCPCOM): it permits TCP communications between CSMs.

2.2.3 Discussion

Looking at these approaches realized to deal with security attacks, some features of these approaches can be derived as main requirements:

- **Distribution of activities**: This important aspect is provided by most existing approaches. It is very important to distribute the control of security management among a number of entities that can monitor the network access at different points.
- **Autonomy**: The CSM and DIDS approaches have shown the necessity to use autonomous entities that constitute the system. They differ in the sense that the final in the DIDS system decision is taken by a centralized manager, whereas in the CSM some decision can be directly taken by the entity.
- **Co-operation**: The CSM has shown also the necessity of security manager co-operation in order to detect security attacks that can not be detected by individual manager. Each CSM detects intrusion and cooperates with the other CSM by exchanging information in order to detect cooperatively intrusive activity.
- **Proactivity**: This feature is found only in CSM, where intrusion detection activities are based on a proactive approach instead of a reactive one. In such case, each subsystem takes autonomously some initiatives driven by its goals. Contrarily, in a reactive approach, a subsystem can only respond to received messages.

Another feature seems necessary but it has not been addressed in the existing systems:

- **Adaptation**: The set of external events is continuously evolving. So, each subsystem must be adaptive to make the whole system capable of sustained performance.

3. Overview of Intelligent Agent Technology

Intelligent agent technology is a growing area of research and new application development in telecommunications. Having highlighted the main requirements for security management, the intelligent agent concept seems to be an appropriate approach to fulfill the intrusion detection requirements. Until now, there is no an internationally accepted definition of an intelligent agent concept [4]. Ferber [5] defines an agent as a computational or physical entity situated in an environment (either real or virtual) which is able to act in the environment, to perceive and partially to represent its environment and to communicate with other agents. It is also driven by internal tendencies (goals, beliefs,..) and has an autonomous behavior which is the consequence of its perception, its representation and its interactions with the environment and with the agents. In fact, this new concept is used in different domains and possesses various meanings depending on the context of its application. However, it can be described by a set of properties including:

- **Autonomy**: is the ability of an agent to operate without direct intervention of humans or other agents and to have some kind of control based on its internal state and/or external environment.
- **Socialability**: is the capability of an agent to integrate itself in a large environment populated by a society of agents with which the agent has to exchange messages to achieve purposeful actions. This property is satisfied even when systems have to share their knowledge and mental attitudes (beliefs, goals, desires, etc.).
- **Proactivity**: is the ability of an agent to anticipate situations and change its course of action. It is a relevant property which occurs in network and system management in order to avoid disastrous effects on global performance. Indeed, proactive agents are capable of exhibiting goal-direct behaviors by taking some initiatives [6].
- **Reactivity**: this kind of behavior means that the agent reacts in real-time to changes that occur in its environment.
- **Adaptability**: is the ability of an agent to modify its behavior over time to fulfill its problem-solving goals.
- **Intelligence**: the term “Intelligence” means that the agent is able to exhibit a certain level of intelligence priority, ranging from predefined actions (planning) up to self learning (define new actions).

Moreover, multi-agent systems, as a sub-domain of DAI, are viewed as computational systems in which several autonomous and intelligent agents interact and work together in order to perform a set of tasks and to satisfy a set of goals [1][5]. Three kinds of agents are distinguished in DAI [7] according to their “intelligence” level:

- **Cognitive agents**: A cognitive agent is able to find a solution for a complex problem while communicating with other agents and interacting with its knowledge base. Its main features include a high reasoning capacity, data processing, perception, learning, control, communication and expertise per activity domain.

- **Reactive agents**: A reactive agent reacts quickly for a simple problem that does not require complex reasoning. Thereby, system intelligence emerge from interactions between a great number of this type of agents.
- **Hybrid agents**: An hybrid agent, a mixture of reactive and cognitive agent, owns some reflex (reactive evolution) to resolve repeated problems and thinks (a cognitive attitude) about complex system situations.

In our work, the term intelligence is used in the sense that security network entities and especially NID components should provide reasoning capabilities, exhibit behavior autonomy, adaptability, interaction, communication and co-operation in order to reach some intrusion detection goals.

4. Towards an Intelligent Network Security Management Architecture

This section describes the functional architecture of the proposed multi-agent system, the agent architecture that constitute the MAS and the agent architecture that we have used to develop this multi-agent system.

We have highlighted in section 2, the main requirements for network intrusion detection. So, in our architecture, we propose to add appropriate functionality to make network entities more autonomous by performing local analysis tasks.

The key characteristics of our architecture include autonomy, adaptability, efficiency and distribution to make the network intrusion detection more flexible and less costly in term of maintenance. In our proposed approach, we define a new architecture, called IA-NSM, which supports NSM activities. It is based on a multi-agent system architecture (see Figure 1). It is viewed as a collection of autonomous and intelligent agents located in specific network entities named NSM hosts. These agents cooperate and communicate in order to perform intrusion detection tasks efficiently and achieve consequently better performance.

In fact, by giving more autonomy to agent in the control of the overall intrusion detection, the task of administration becomes easier. Administrators do not have to concern about all the security problems. They interact with the agent from a high level using security policies. Security policies tell the agents what behavior they should exhibit when attacks occur. Hence, communications between agents permit to collect information. This information permits the agents to identify attacks that can not be detected if it is static. Giving more autonomy to the agent permits the system to react in “real time” to attacks and to take necessary actions to avoid severe consequences of the attack.

4.1. The IA-NSM Functional Architecture

In our architecture, we propose a hierarchical structure of autonomous agents. In the functional architecture (see Figure 1), we distinguish two: a *Manager Layer* and a *Local Layer*.

- The *Manager Layer* manages the global security of a network. This network can be local or distributed. In this layer we identify three levels of agents: *Security Policy Manager Agent*, *Extranet Manager Agent* and *Intranet Manager Agent*.

    - The ***Security Policy Manager Agent (SPMA)*** manages the security policies specified by the security officer.
    - The ***Extranet Manager Agent (EMA)*** manages the security of the entire distributed network. Its role is to manage and control *Intranet Manager Agents* (IMA). These agents report pertinent analysis to the EMA. The role of the latter is then to perform another analysis on suspicious events in order to confirm or not the detection of an attack. It can also ask for another data processing and delegate then new monitoring tasks to the IMAs. The *Extranet Manager Agent* communicates with the *Security Policy Manager Agent***.** This latter can specify new security policy, new monitoring tasks or new attacks to detect. The *EMA* is also responsible for distributing the set of *Local Agents* to each IMA.
    - The ***Intranet Manager Agent (IMA)*** manages the security of a local network. It controls the *Local Agents* and analyzes the monitored events reported by these agents.

- The *Local Layer* manages the security of a domain, which is constituted by a set of hosts. This layer is composed of a group of *Local agents*, which have specific functions. In fact, the *Manager Layer* specifies to the *Local Layer* the activities that must be monitored. These activities can be classified in *Extranet*, *Intranet* and *Local* activities. According to this classification, we distinguish 3 kinds of *Local Agents*: *Extranet Local Agent*, *Intranet Local Agent* and *Internal Local Agent*.

In this hierarchical multi-agent model, each *manager agent* has the ability to control specified agents and to analyze data, whereas, the b*cal agents* monitor some specified activities.

The *manager layer* interacts with the *local layer* by sending goals, delegating specific monitoring/detection tasks and receiving pertinent reports and alarms.

In each level, agents communicate and exchange their knowledge and analysis for detecting intrusive activities in a co-operative manner.

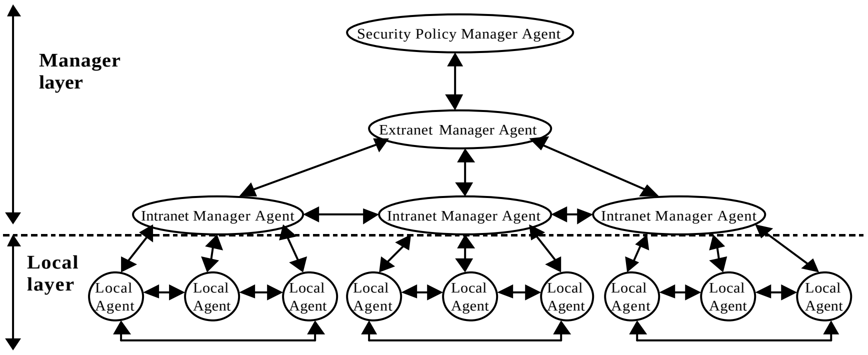

Figure 1: IA-NSM Functional Architecture

For the multi-agent application presented in this paper, the hybrid model is adopted for each agent.

4.2. Hybrid Agent Architecture

In this section, we describe our model agent (see Figure 2).

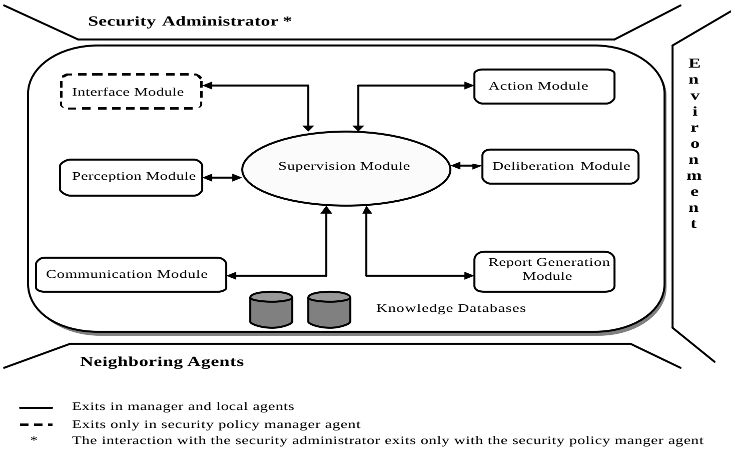

Figure 2: Hybrid agent functional model

Our agents are composed of seven modules: perception, deliberation, communication, action, interface, report; each executing a different task. The supervisor entity coordinates tasks of the different modules.

- A ***perception module***: that gathers all security-relevant events produced in the agent environment.
- A ***communication module***: that allows agents to communicate their analysis, decisions and knowledge.

- An ***action module***: its role is to take appropriate actions when an intruder is detected.
- A ***report generation*:** establishes reports on detected attacks to be sent to the administrator.
- A ***deliberation module***: that enables agent intelligence and autonomy. The hybrid agent should be able to reason and extrapolate by relying on built knowledge and experience in a rational way. Decisions of the agent depend on the security environment status, the neighboring system evolution and its mental attitudes.
- An ***interface module***: interacts with the security administrator receiving administrator requests/specifications, delivering reports, sending alarms when an attack is detected and asking for additional information or confirmation when necessary. For example, the administrator can ask for the current network security status. This module exists only in the SPMA.
- A ***supervision module*** coordinates interactions between the different modules using a finite state automaton.

4.3. Implementation

In this section, we describe the platform that we use to develop our multi-agent system architecture and we give some details on the implementation aspects of the proposed IA-NSM architecture.

4.3.1 DIMA

In attempt to define a modular and generic architecture, which owns the main properties of an agent [6], DIMA proposes the extension of the single behavior of an active object into a set of behaviors [8]. An agent is a pro-active entity. It does not simply act in response to the received messages from the other agents. For example, it interacts with its environment and deliberates to determine the most appropriate action.

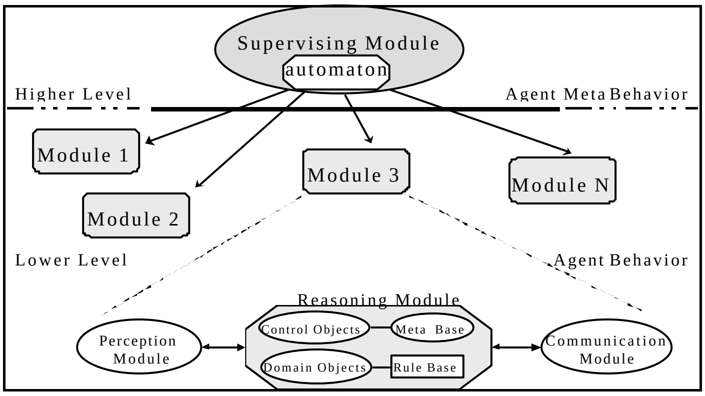

Figure 3: DIMA agent architecture

The DIMA architecture (see Figure 3) relies on two layers:

A first layer made up of interactive modules, which represent the different concurrent agent behaviors such as communicating, reasoning and perceiving. They provide the agents with some properties described in [6] such as autonomy and cooperation. For example, the communication module manages the interaction between the agent and some other agents of the system. Therefore, it is very important to make the agent cooperative.

A second layer made up a supervision behavior representing the agent meta-behavior. This meta-behavior gives each agent the ability to reason about its own behaviors.

4.3.1.1 Examples of Behaviors

To model intrusion detection, agents need to combine cognitive abilities (knowledge-based) to reason about complex situations, and reactive abilities (stimulus-response). So, an agent may have two kinds of behaviors: *reactive* and *cognitive* behaviors.

DIMA proposes three examples of modules: the *perception module* (procedural behavior), the *deliberation module* (knowledge-based behavior) and the *communication module* (which can be either procedural or knowledge-based).

- The ***perception module*** manages the interactions between the agent and its environment. For example an agent percept a list of suspicious events.
- The ***deliberation module*** represents beliefs, intentions and knowledge of the agent. It is responsible 1) for generating adequate responses to the messages transmitted by the *communication module*, or to the changes detected by the *perception module*, and 2) for achieving the agent goal(s). This goal can be the detection of a specific attack.
- The ***communication module*** manages the interactions between the agent and the other agents of its group(s), no matter what machine they are running on. It defines the mailbox of the agent and the way the messages are received and enqueued for later interpretation. An agent may need some others information to refine its analysis. In this case, it asks other agents to give it the necessary information.

These three modules are appropriate to our application domain. In fact, agents related to intrusion detection often own a deliberation behavior, and a communication behavior and/or a perception behavior. Moreover, the use of a DIMA facilitates the integration of new modules such as a learning module.

4.3.2 IA-NSM Architecture Implementation

The hybrid agent model, as described in section 4.2, is selected to be implemented in each level of our IA-NSM architecture. So, a set of such hybrid agents is installed in SPMA, EMA, IMA and LA entities.

As depicted in Figure 2, deliberation, perception and communication modules are performed respectively by the deliberation, perception and communication modules of DIMA. The interface module, action module and report generation module are three new modules, defining three new behaviors, that must be integrated in the DIMA platform, in order to be implemented.

In DIMA, an ATN is associated to each module. In the ATN, we distinguish two main states: an initialization state 'INIT' and a supervision state 'WAIT' that manages the processing of each module.

In this paper, we provide an ATN for an *Extranet Local Agent* dedicated to the detection of both doorknob rattling and IP spoofing attacks.

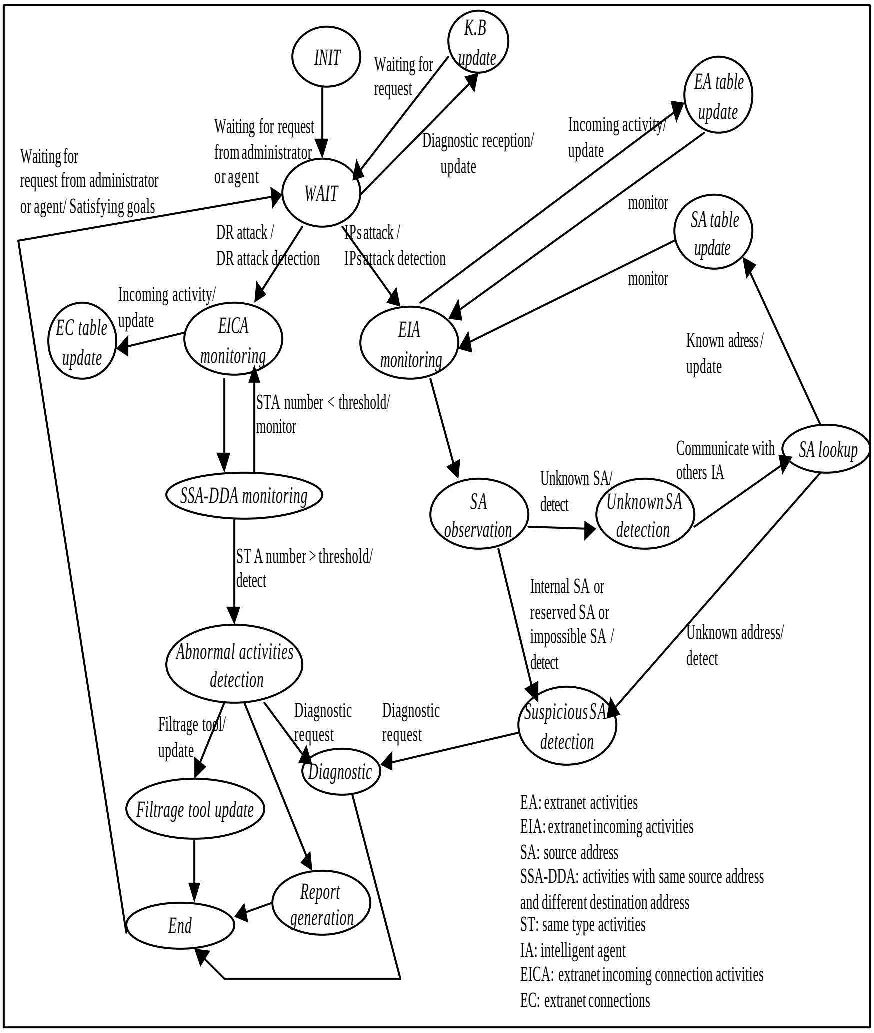
INIT
K.B update
Waiting for request
EA table update
Waiting for request from administrator or agent
Diagnostic reception/ update
Incoming activity/ update
Waiting for request from administrator or agent/ Satisfying goals
WAIT
monitor
SA table update
DR attack / DR attack detection
IPs attack / IPs attack detection
monitor
Incoming activity/ update
EC table update
EICA monitoring
EIA monitoring
Known adress / update
STA number < threshold/ monitor
Communicate with others IA
SA lookup
Unknown SA/ detect
SSA-DDA monitoring
SA observation
Unknown SA detection
ST A number > threshold/ detect
Internal SA or reserved SA or impossible SA / detect
Unknown address/ detect
Abnormal activities detection
Diagnostic request
Diagnostic request
Suspicious SA detection
Filtrage tool/ update
Diagnostic
Filtrage tool update
Report generation
End
EA: extranet activities
EIA: extranet incoming activities
SA: source address
SSA-DDA: activities with same source address and different destination address
ST: same type activities
IA: intelligent agent
EICA: extranet incoming connection activities
EC: extranet connections

Figure 5: An example of ATN of an Extranet Local Agent

5. Conclusion

In this paper, we underlined the network intrusion detection requirements. We presented some existing systems and illustrated their limitations. Mainly, the flexibility, autonomy, adaptability and distribution were the principal features to be addressed to build a suitable architecture that fulfills these requirements. Thus, the introduction of a multi-agent system was proposed as a mean of modeling and implementing adaptive decision. The multi-agent system makes intrusion detection more flexible. In fact, the autonomy given to the agents reduces considerably the implication of the security officer in security management and makes its administration tasks easier. A new architecture using intelligent agents is outlined. Moreover, a functional structure for hybrid agent is presented.

For further work, we intend to implement the presented work with the DIMA platform. Moreover, we will specify more precisely mental attitudes in terms of beliefs, goals and motivations used by the deliberation module of the agent to perform detection of network attacks.

References

[1] L. Gasser, "An overview of DAI", Kluwer Academic Publisher, Boston, 1992.

[2] L.T. Heberlein, B.Mukherjee, and K.N.Levitt, "Network Intrusion Detection", IEEE Network Journal, pp. 26-41, May/June, 1994.

[3] Maj. Gregory B. White, Eric A. Fisch, and Udo W. Pooch, "Cooperating Security Managers: A Peer-Based Intrusion Detection System", IEEE Network journal, pp. 20-23, January/February, 1996.

[4] H. Nwana and M. Wooldridge, "Software Agent Technologies", BT Tech. Journal, vol. 14, no 4, pp. 68-78, 1996.

[5] J. Ferber, " Les Systèmes Multi-Agents", InterEditions, 1995.

[6] M. Wooldridge and N. R. Jennings, "Intelligent Agents : Theory and Practice". Knowledge Engineering Review, vol. 10, no 2, pp. 115-152, 1995.

[7] H. Labiod, "Error Control in Wireless ATM networks", Ph.D thesis, University of Versailles, France, 1998.

[8] Z. Guessoum, "An Operational Environment of Conception and Realization of Multi-Agent Systems", Ph.D thesis, University of Paris VI, France, 1996.