International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 3 (2011) pp. 193-201

A Hybrid Approach for IEEE 802.11 Intrusion Detection Based on AIS, MAS and Naïve Bayes

Moisés Danziger[1], Fernando Buarque de Lima Neto[2]

[1] Computing Engineering Programme – Polytechnic School of Pernambuco,
University of Pernambuco, Recife-PE, Brazil
md@ecomp.poli.com

[2] Computing Engineering Programme – Polytechnic School of Pernambuco,
University of Pernambuco, Recife-PE, Brazil
Senior Member IEEE
md@ecomp.poli.com

***Abstract*: The advancement of network technology has been offering substantial improvements for users in general. Paradoxically, new technologies end up bringing also new types of problems. Not only one can find issues related to architecture failure, mis-configuration and bad adaptation, but also a growing number of problems related to user security. The latter is a problem strongly connected to wireless networks. Obviously, this is facilitated by the means of transport used in those networks (radio waves). Intrusion attempts are common and a strong concern regards detecting them. To this end, due to the fact that it is easy to attack and tough to defend wireless networks, good new approaches would be the ones that could profit from intelligent techniques as they are adaptive and thus may identify attacks that are not necessarily known in advance. In this work we use the Danger Theory (DT) and a Bayesian classifier (naïve Bayes) embedded into a military style multi-agent (MAS) to create a light, dynamic and adaptive detection system to work with the lower layer of wireless networks (WIDS). Experimental results show that the artificial immune aspect of the system is capable of detecting unknown issues and to identify them automatically with considerable few false alarms and low cost for the network traffic.**

***Keywords*:** Intrusion Detection, Artificial Immune Systems, Danger Theory, Multi-agent Systems

I. Introduction

The human immune system (HIS) has been the inspiration for many algorithms, but on security it is possible find even more candidate problems because of the blatant functional similarities. The new generation of artificial immune systems (AIS) may quickly become the new weapon among computational intelligence techniques that could easily be deployed for network security [1] – [3]. Shortly after its initial use, "1st Generation" of AIS proved not to be scalable and not the first choice for real time application [2]. However, as "2nd Generation" arrived – that is algorithms that use danger theory, the pitfalls of prior AIS algorithms were dramatically reduced. For instance, the self-non-self-methods has been replaced by the improved cell signaling processes of DT [4],[5].

Looking at intrusion problems, Aickelin et al. [6] devised the analogy between IDS and DT. For them, DT presents important characteristics to solve the "1st Generation" problems and some authors had presented works on that with good results [7] – [9]. Thus, in our work DT is used as the primary line of detection method embedded in the intelligent agents.

Intelligent agents, the key component of multi-agent systems (MAS), present many suitable characteristics to be used in association with DT [10]. The basic principle of an agent for instance is its perception and interaction with the environment. In a simple observation on reasonably useful IDS, it is fair to imagine the presence of sensors in several specific loci of the network. Hence, certain agents can be used as sensors devices in a comprehensive IDS. In this work six types of agents were created, each one with different activities to perform towards the high goal of intrusion detection.

The model put forward here was developed based on the IEEE 802.11 networks standard operation and this choice was a consequence of the current abundance of problems to be solved with this protocol. To prove the efficacy of our approach the experiments of this paper were carried out for five types of known attacks on wireless networks, namely: (i) interactive packet replay, (ii) fake authentication, (iii) ChopChop [11], (iv) Cafe-Latte [12] and (v) Hirte [13] attack for IEEE 802.11 networks. Our main objective is testing detection, identification and ability to counteract under anomalous events when the proposed system is running in automatic mode. For that we had to development a detection tool able of operating in the link layer of the standard inspecting frames instead of packets. This paper extends the model presented by Danziger et al. [14] where one can find more details on DT and DCA algorithm, both used here.

This paper is organized as follow: in section two MAS and IDS are detailed in an IEEE 802.11 perspective. In section three we detailed the extended version of the proposed model.

The methodology and results of experiments can be found in sections four and five, respectively. Results and future works are included in section six.

II. Multi-agent System and Intrusion Detection System under IEEE 802.11 Network Vision

A. Multi-agent systems

Able to embed distinct computational intelligent techniques, MAS have been a great open field for research with applications in many areas of knowledge. Moreover, a great deal of research involving artificial intelligence (AI), particularly, MAS has produced practical good results in recent years. Multi-agent systems can be defined as a collection of computational entities that have the ability to solve problems in cooperative or individual manner through the exchange of information [15]. It is intuitive that an isolated agent is less likely to solve a distributed task than a collection of them.

Looking at IDS, the MAS can be represented by set of sensory agents or combatant agents [16]. Although this possibility is already available, in this work, we have refined agents solely for distributed aspects of intrusion detection.

B. Individual agents

Agents can be of different types and have some sophisticated features to increase their performance in the environment which it is inserted. However, the basic principle of agent will always be [15]: (i) its perception of the environment, (ii) its ability to interact with other agents and (iii) its ability to initiate and persistently pursuit of its own goals.

Some intelligent engine can enable them with adaptability, which is an important feature in dynamic environments. Intelligence is also quite an interesting asset to distributed systems such as IDS. In this work the AIS based on DT is the intelligent engine embedded in the utilized MAS for detecting attack in conjunction with naïve Bayes for identification of new attacks. For instance, we use the cells abilities (DC and T-cell) embedded in specific agents simulating the same function found in the HIS. Obviously, they work in a highly cooperative mode.

C. IDS and IEEE 802.11 standard

Nowadays, an IDS is an essential element of security in any system and network. Its importance depends of that will be protected and what is the aggregated value.

The basic principle of IDS operation is the use of sensors to detect problems (or anomalies) in the critical parts of the system or network (sometimes both together). This case remember the main function of Dendritic Cells (DC) [17] in IS and the simply agent in the MAS. Thus, it is possible the use of one agent representing DC and another cell type at IDS-immune based.

Looking at its main activity, any IDS acts based on two models: (i) based on evidence of intrusion and (ii) based on the deviation of behavior. Normally, these two actions are called misuse-based IDS for “i” and anomaly-based IDS for “ii”. Here, we emphasize the second approach.

Beyond of already known problems from wired network standards, new others arise in wireless local networks (WLAN). For instance, a good IDS must be able to work with lower layers of the network architecture rather that working with limited band (i.e. the channel of transmission are radio waves in air travel). Thus, one good tool for detection needs to be able to listen the physical layer)

Two main problems from all (or almost) IDS are known as (i) automation and (ii) adaptation. In the first case, automated tools are desired because of human errors and agility. Look at, suppose for a moment one problem encountered by system, it needs a decision by the administrator to continue, but either is not present or does not see the system call. Thus, depending of problem degree, it can imply serious difficulties for the system, including its full commitment.

Nonetheless, the adaptation follows quite near of automation. Clearly, it is aligned with one well designed tool with ability to detection and counterattack of new anomalous events (unknowns). This issue has reached the majority of IDS and researches about. All these pitfalls were selected to be improving by our work.

1) Basic facts on IEEE 802.11 network and security

The IEEE 802.11 network has grown considerably and their security problems follow suit. The vulnerability of the communication channel, the hardware failure associated with serious difficulty in the cryptography algorithms, the many standard problems between manufacturers, some indifference of network administrator associated with the inexperience of normal users (i.e. mainly home users) has been transformed this network model as an almost perfect ambient to several types of attacks.

In the IEEE 802.11i version, several corrections for security problems were incorporated. But, one difficult remains: the DoS vulnerability [24]. This weakness is caused by the lack of management frames authentication, which allows any IEEE 802.11 network vulnerable to spoofing attack [25]. The last IEEE 802.11 version, named “w” came changes for combat this DoS problems. Unfortunately, the most of networks are carried out with low protection. Surprisingly, we found several enterprises with misconfigured networks during this research using “i” version or previous. Hence, for this work we used the “i” version.

Interestingly, IEEE 802.11 networks deployed with low awareness of vulnerabilities of each version can not be only blaming the administrators. For instance, the Wi-Fi Protect Access protocol (WPA) with the strong Advanced Encryption Standard (AES) algorithm embedded force exchange of the transmission equipment (incompatibility between hardware). Many times, arises a financial problem, managers may have no choice and take risks with the current structure can be a reality. Thus, become very important good detection tools.

Passive and active are two different models of attacks present in IEEE 802.11 networks. Generally, the attacker uses the first attack to conduct a discovered by scan across the networks working in a given area. Most detection tools are not able to detect this attack due to its most striking feature: just listen to the channel.

Thus, more sophisticated tools are necessary. Although some tools make spectrum analysis (i.e. signals analysis) to detect physical presence of the attacker, customarily (because

are expensive tools), some companies employ people to move around the physical space in order to detect the presence of intruders (physically). For the active case, the most studied in the literature, the attacker arises by the frames sent. This is the case related here.

III. Extended Model of Wireless IDS Based on DT, Naïve Bayes and MAS – military style

The extended model presented here detects anomalies using agents within a hierarchical structure of functionalities spread across workstations and servers. The adaptability, induction and inference abilities are drawn from an immune-inspired engine DT-based.

From a biological point of view, many cells and molecules beyond several mechanisms form a great complex system. Despite of various mechanisms, it boils down in two main lines: innate system and adaptive system. For the first, is independent of the foreign antigen. For the second, beyond of antigen-dependent, it has memory and leaning capabilities.

Six types of agents inspired in the military hierarchy were devised, namely: (i) basic agent, (ii) subaltern agent, (iii) intermediary agent, (iv) superior agent, (v) logger agents and (vi) messenger agents. In analogy with HIS, (i) and (ii) belong to the innate system (i.e. those from birth), (iii) and (iv) belong to the adaptive system (evolve over time), and (v) and (vi) are auxiliary agents - signalers. Indeed, (ii) belongs for innate and adaptive system too (see figure 2). The just difference is the moment of creation. For instance, agents will belong to the innate system if instantiated on boot only. Those created during the execution time, will belong to the adaptive system. This is a configuration parameter in our model.

The complete architecture of the system can be seen in Figure 1. The following sub-sections explain each class of agent's function of our model.

A. Basic agent

Although its name may suggest otherwise, this is the most important type of agent in the system (militarily can named "reconnaissance soldier"). Its main function is to detect problems through the processing of several signs based on DT-concepts applied under IEEE 802.11 network adapted DCA algorithm [18]. The detection process is made possible by a packet analyzer that preprocesses the data collected to DCA entrance. As it is conceived, this process can be on-line or off-line; there is only one agent of this type per node.

B. Subaltern agent

This agent represents the T-cell of HIS and can be viewed as one memory agent. To avoid resource consumptions, for each subtype of IEEE 802.11 frame, just one subaltern agent is created. Nevertheless, if already exists one agent for a given frame, case occurs small changes (in the frame), just are made updates (for better performance). Thus, these agents can evolve over time. Analogously as in HIS, when the T-cells are created for fighting those with better performance are transformed in memory cells. This mechanism helps to reduce the response time in future attacks. For executing correctly its devised function, the internal structure of subaltern agents is formed by an array with the attack known variation alongside the routines of combating the attack. As combating is not the focus of this paper some simple protective actions were implemented (e.g. blocking traffic and shutdown the workstation module).

In the proposed architecture subaltern agents either are created by intermediary agent (after sent to stations in cloned method) or can be created on boot. When in the station, they remain with mobility turned off.

C. Intermediary agent

The main function for this agent is the identification of the unknown problems. For this difficult task, it uses a data base with some already features of known attacks and possible variations of them. Looking at this agent, it has the function of bone marrow and thymus. The first is the birthplace of the IS cell and the last the local of cell maturation. As DT does not use selection negative theory, the maturation process happens into an update process (i.e. when the intermediary agent uses the naïve Bayes for update the structure of one already created subaltern agent). Thus, this process helps to developing the agents.

For discovering new types of attacks or identifying some variation of already known types of attack, it is necessary an able tool for performing the right classification. Although there are quite sophisticated classification techniques available, many are considerate sluggish for real-time execution. So, we decided use a simple and fast naïve Bayes into the engine motor of the intermediary agent. This technique uses statistics looking for compatibles candidates [19] – [21]. Indeed, this choice was related with issue of agility, in spite of known problems of actualization of data base on running.

When a new type of attacks is identified, a new subaltern agent is automatically created with an internal structure that will allow the identification of that attack on the stations rapidly. Although when a simple variation of an attack is detected and already there is one agent, the new information is added into. This choice ensures low growing in the number of elements (objects) and avoids the memory problems.

We stipulated that if after three attempts a new attack cannot be identified, one generic subaltern agent is created and a message is passed to superior agent with all antigens embedded to it.

It is also possible to have more than one intermediary agent; however, one needs to evaluate: (i) the network size, (ii) the cost of communication and (iii) the time of response. If there are two or more agents, then, the knowledge data base (of attacks) must be shared and frequently updated.

D. Superior agent

Usually IDSs send alerts for the network administrator and wait for some response. The process of combating an identified attack is often not automated and can bring serious problems in case the administrator is not active. Otherwise, some automated process can be dangerous in case they are not adaptive and are not controlled at some stage. Furthermore, adaptive assumptions need to be defined about issues of accountability (e.g. when, how and till point the system can act by itself)

Because of this need to control counter-attacks in certain extension, the superior agent was created. This agent has the responsibility of auditing the correct system execution.

According to proposed model, all processing can be done automatically, so the function of superior agent is very important to suppress the possible auto-destruction event. This is analogous to what happen in HIS, where auto-destructive processes named auto-immune (i.e. the body combating it-self) are controlled by a mechanism that control the T-cells population (e.g. subaltern agent).

This agent has a structure that permits to read the logs written by logger agents in each station and at server. Here, the mobility is turned on and guided by a round-robin routine; all superior agents (more than one could be created as well) visit each station in a cyclic manner. To avoid that the agent get stuck as long as visit the stations, one duplicated agent remains in the server. On visiting the stations, it sends a message for cloned agent. Thus, the possible break in audit process is mitigated.

This kind of agent use a data mine process to find information that can be useful to the expected operation of the system. We use special words (e.g. network down, overload on the interface *WlanX*, ping response expected and others). If one problem was found, this agent will decide what to do using routines stored into its decision module. In the system tested here, the only contact of the system and the network administrator is a message reporting the problem found and the counter-attack decision taken.

E. Logger agent

Because of the great quantity of information generated when all the other agents are working and the audit relevance to automation, this agent was created with a unique function: it registers all activities executed by the system in an orderly and rational manner. All stations have one agent of this type, as well as the server. For simplification, a text log file is generated where all records can be read by the superior agent, whenever necessary or each cycle at agent.

One could think that exist a problem with just one agent by station and its endeavour. So, in our experiments the system worked very well. Otherwise, DDoS attack could be a problem. For this case, a solution would be reducing the time of passage of superior agent among stations or some other internal method.

F. Messenger agent

The communication process on every MAS is of fundamental importance. All agents need to exchange many messages and can generate high traffic in the network or among them in a workstation. So, we decided to create an agent to manager this process. All stations and the server are bestowed with a messenger agent. This particular agent reduces the work of others agents, by managing communications more efficiently.

Indeed, the local agents can directly exchanging messages or by messenger agent. This process help to mitigate some fail process or a possible attack against agents.

G. Naïve Bayes

It is regarded to be the simplest technique of classification as it operates on a strong assumption of independence between classes [15] (i.e. the probability of one attribute does not affect the probability of the other). Suppose a series of *n* attributes, the naïve Bayes classifier makes *2n!* independent assumptions. Almost all artificial intelligence techniques need to be trained and validated. To avoid high number of error during training it is necessary a good division of data for training, test and validation. This is a low point that can lead towards an overall bad performance. In the training phase, the probabilities of a situation to occur are calculated given a particular attribute, and then this probability is stored. This process is repeated for each attribute; the time taken to calculate the relevant probabilities for each attribute are stored too. This allows us assessing the time necessary to calculate the probability of the given class for each example. For this work, we deployed the "leave-one-out" as method of validation for naïve Bayes.

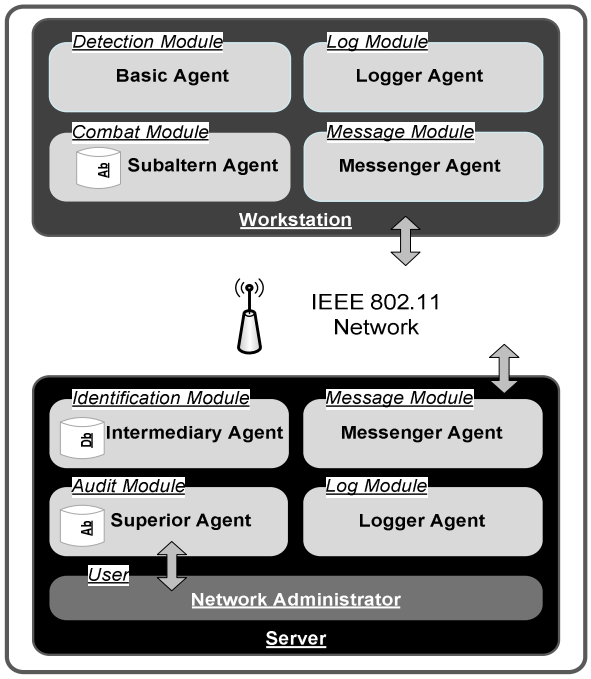

Figure 1. Model of WIDS MAS immune-based

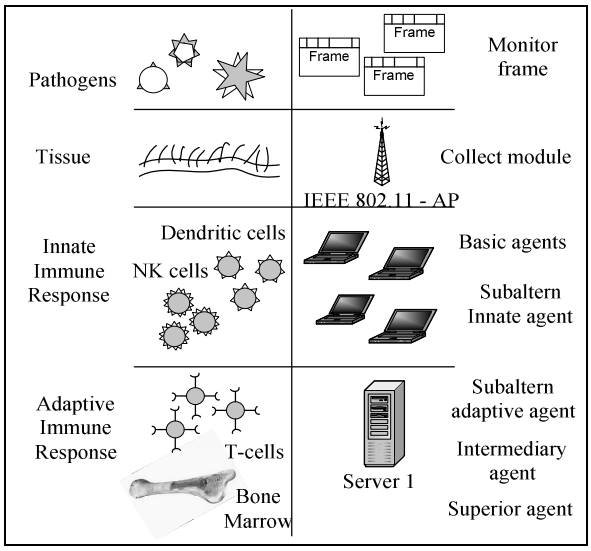

Figure 2. Analogy between HIS, MAS and model proposed

IV. Methodology

A. Attack scenarios

To test our architecture, we devised a network environment containing five workstations and one server, setting up a fairly common client-server environment. Only one AP is used with an antenna of 2.2 dBm (power).

To assess the efficiency of the model we have conducted eight experiments. Because of space, results of only one station were selected for sampling the results of each experiment, however they very much represent the results obtained for the other station. The experiments were divided by type of attack. According to each attack there is one ".cap" database available that is transformed into ".txt" through the Wireshark tool [21].

B. Signals and antigens

Commonly, the representation activity is a problem of bio-inspired systems. Following the same method as in [14], each frame is transformed into an antigen keeping the same structure model standardized by IEEE 802.11 and the signals are reference of antigens. Three signals formed the basic input entry, namely: safe signal (SS), danger signal (DS) and Pathogen Associated Molecular Patterns (PAMP), in other words, a clear attack.

If for each type of attack three signs are the minimum required, in our experiment were used at least fifteen signs. Whenever possible we used two or more signals by category. However, we stress the importance of time during the system running. For instance, the system needs to process all input/output frames in a few seconds otherwise it will not be of any use (as an attack can happen unnoticed). Naturally, more signals to process imply in more effort for processing. All signals (representing attacks) were chosen after tedious observation of frame IEEE 802.11 traffic.

Our defined signals are showed follows.

1) Interactive

SS is the number of data frames per second with destination in mode broadcast and size between 40 and 100 bytes. PAMP is the number of data frames received with unknown source and destination in mode broadcast. Its DS is the number of frames received.

2) Fake authentication

SS is the number of frames ACK received by AP with size different of standard (i.e. more than 10 bytes). For PAMP, the value is equivalent of number of authentication frame received per second, but, there can be exists attack with one frame per second and in discrete form, so, the number is counted if existing this frame in the last 5 seconds. It DS is the number of frames received with different source of known MAC Address.

3) ChopChop

SS is the number of repeated small frames per second and PAMP is the number of frame with different destination per second. It DS is the number of frames sent per second.

4) Cafe-Latte

SS is the number of frame received with size between 69 and 129 bytes. For PAMP, the value is the number of frames received of unknown source or address of network. In the case of DS, it is the number of frames received per second.

5) Hirte

SS represent the number of frame with the same sequence number (SN) per second and PAMP is the number of frames received per second.

After the first experiments and the appropriate definition from signals, we discovered that for some attacks definitions are the same for some signal. Thus, it is clear that they will bring forth the same values. This does not threaten the capacity of detection. Indeed, as will be showed in the next sections, it can help the detection as a whole giving a broader perception to cells.

C. System setup

Six models of computer were used for the experiments. All of them use at least 2 GB of memory RAM and a fast dual core processor minimum (one of them was desktop and the others, notebooks). One station is used to perform the attacks with the Ubuntu Linux SO (kernel 2.6.26) and framework Aircrack-ng [22] installed. The architecture was implemented in the JAVA language and the agents were developed using the framework JADE [23].

V. Results and Analysis

The results of the experiments are depicted in the figures below (i.e. 3, 5, 7, 9 and 11). In all of them, we present the output signals of the detection process using DCA. Semi-mature and mature signals are indicated, referring to attack detection with the regular traffic. In the case of semi-mature, the DC processed more security signals (i.e. SS), thus, can represents not attacks. In the mature signal, the DC collected and processed more danger and/or attack signals (i.e. DS and PAMP), thus, can represents attacks. Figures 4, 6, 8, 10 and 12, depict the antigen classification by intermediary agent across naïve Bayes running. This happens as soon as the basic agent detects anomaly and there is no subaltern agent to resolve the problem. Then, it requests help to the server using messages with antigens and signals under collected context. In the figures, in the left vertical axis, 0 represents false (not anomalous) and 1, represents true (anomalous). The right vertical axis shows the frame type of antigen when classified as anomalous. In such case, one specific subaltern agent (i.e. based on frame type) is created. These numbers refer to the tables of the IEEE 802.11 standard. Table 1 shows their meaning.

The table 2 presents other type of relevant experimental result of simulations carried out. For example, it is possible to see the errors rate, the false alarms, the agents created, and the messages shared.

As showed above, the intermediary agent uses a naïve Bayes classifier and the value of its efficiency was 83.9% after the application of "leave-one-out" validation method. The

experiments combined showed good results enough that no false negative alarms were generated (in table 2).

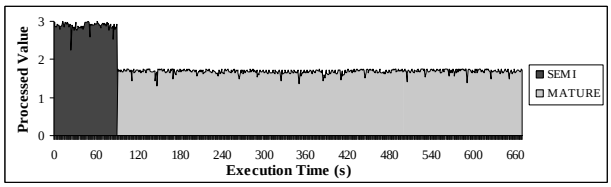

Figure 3. Signals processed for Hirte attack (Exp. 1)

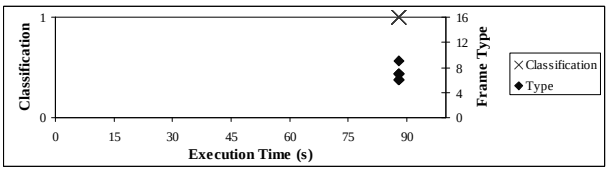

Figure 4. Frames classification for Hirte attack (Exp. 1)

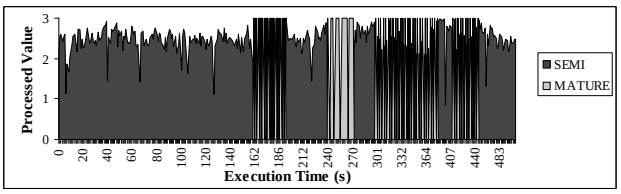

Figure 5. Sample of processed signals during fake authentication (Exp. 2)

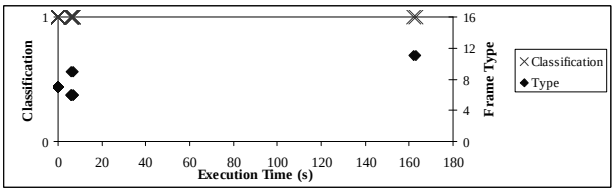

Figure 6. Frames classification for fake authentication (Exp. 2)

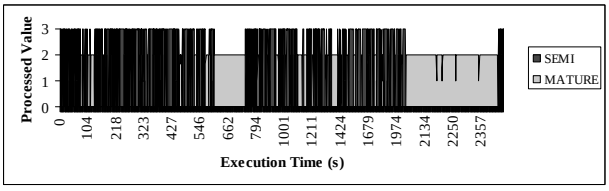

Figure 7. Processed signals for Café-Latte attack (Exp. 3)

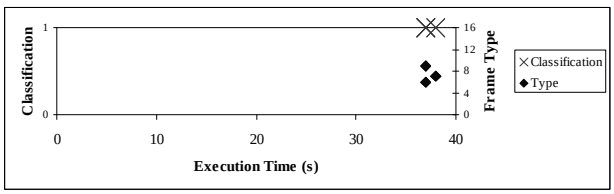

Figure 8. Frames classification for Café-Latte attack (Exp. 3)

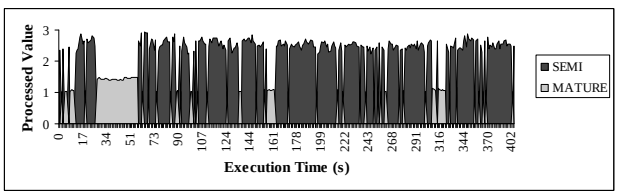

Figure 9. Signals processed for ChopChop attack (Exp. 4)

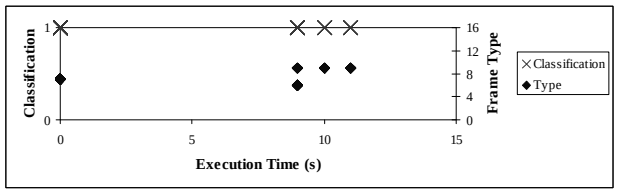

Figure 10. Frames classification for ChopChop attack (Exp. 4)

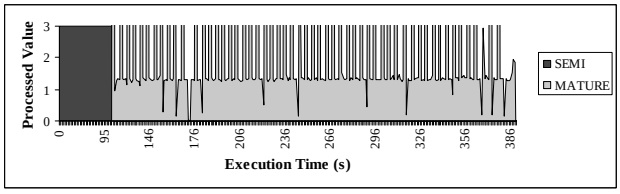

Figure 11. Signals processed for Interactive attack Exp. 5)

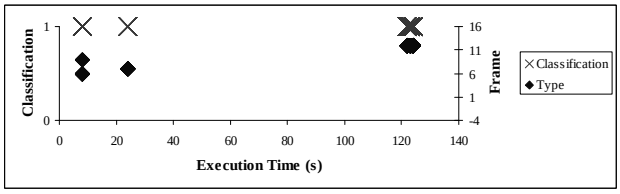

Figure 12. Frames classification for Interactive attack (Exp. 5)

According to results of all figures, the algorithm of detection was effective to detect most of the problems (i.e. all types of attacks tested). In the Table 2, for Hirte (Exp. 1) and fake authentication (Exp. 2) attacks, we obtained zero false negatives. The errors rate is calculated using the Equation 1

$$\frac{n*100}{N}, \qquad (1)$$

where, n is the number of seconds that have been detected attack, N is the total number of seconds that have suffered attacks.

After the attacks some false positive alarms were found with the creation of subaltern agents; this without actual necessity. This is not a major limitation, as it implied only in some small loss of resources (e.g. memory). Some antigens found alone within one second may have not been an attack, but, if in the last or in the next second the same antigen is present, then odds of an attack are high. Unfortunately, one antigen was not identified as an attack for the classifier. After a long time observation, we mitigated this problem using a delay at DCell. When it found one just PAMP signal during one second, the cell sets a flag and store those signal. If in specific range of time another signal (equal) arises, then the cell even with semi-mature larger value receive mature state. Although it can seem confusing, the rationale is that just one frame may not represent an attack too; it can be an isolated event. Thus, the evaluation must be careful.

During the experiments, in spite of not waiting we observed that some antigens were classified as attack and, after observation in database (frames collected), the classification performed by the system was proved correct. Together these antigens, we observed that, some antigens had some anomalous behavior (e.g. antigen of type Null function or

Probe Response that can be seen in figures 4, 6, 8, 10 and 12) and were classified as attack generating a specific subaltern agent. These behaviors can express attack or a likely attack. As commented above, likely attacks can produce values defined for a given signal category (as a reflex of similarities to other types of attacks). -Thus, even without the need to process all types of attacks, the system has demonstrated the ability to identify newcomers. This fact shows very clearly that the system (as a whole) can discovered problems even if they fool the detector and the operator do not define them as false alarm.

The migration threshold used a random number ranging between 0 and 2. We have found that values above 2 generate too many false positives, on the other hand, below 0.5 just the opposite happens, i.e. too many false negatives. So, the better values are between 0.5 and 1.5.

The response time for the intermediary agent vary. This happen because of the distance between station and the AP, mainly. We tested that with various distances. But, three of them were used as standard: (over) 10, 20 and 30 meters. The results were produced in two seconds for first case, two or three seconds for the second case and, four or five for last case. The measurement was made from the output of message to the station till the arrival of a specifically created subaltern agent.

The small amount of messages exchanged between the workstation and the server, shown in Table 2 represents the analogy between the system innate and adaptive system of the HIS. When a station has a problem and must seek the help of intermediary agents (in the server), a new subaltern agent is created or updated, from any existing one. All stations receive the same agent created (i.e. the agent is cloned and does not suffers leverage of no one) and, if there are new instances of the same attack, it is not necessary help of the intermediary. In this case the reaction time is very fast (less than 1 second in tests, see table 2). This process also decreases the amount of messages exchanged among the stations and the server, reducing the impact on the network traffic.

Some cases, in the figures of antigens classifications, one can observe the appearance of one or more equals help request. This happens because there is a time, as long as message travelling between workstation and server. So, if other problems are found soon after that moment, the subaltern agent not was created and/or registered in workstation yet. That is why these new requests were made.

The number of frames passed through the network is calculated using the Equation 2,

$$\frac{V * S}{1500}, \tag{2}$$

where, V is the value found, S is the media size for messages and 1500 is the maximum value of frame.

All experiments were developed for the WEP standard. We tested for WPA too, but, with deauthentication attack (i.e. the WPA protocol does not allow the fake authentication; the first action of the attackers on WEP). Preliminaries results indicate efficacy for this environment.

During the experiments, some problems happened with the network (e.g. shutdown or null connection), as can be seen in the last rows of table 2. The superior agent is responsible for auditing problems within the system, so, every one minute, it was set to visit each station and to read the log file. The goal is to find problems, or to inspect why a particular station is not sending status message to the administrator. An extra security measure included, to avoid the superior agent loss during the visitations, is that the agent sends a clone of itself to perform that task.

Table 1. The frames values and its analogy with the IEEE 802.11

Frame	Num	Frame	Num
Association request	1	Null function (no data)	9
Association response	2	Disassociate	10
Reassociation request	3	Authentication	11
Reassociation response	4	Deauthentication	12
Probe request	5	Power save (PS)-Poll	13
Probe response	6	Request to send	14
Data	7	Clear to send	15
Beacon frame	8	Acknowledgement	16

Table 2. Settings and results of experiments (FP = false positive and FN = false negative)

Description	Exp 1	Exp 2	Exp 3	Exp 4	Exp 5
Frames size	54063	62607	142876	56978	111393
Seconds of attack	426	271	823	314	271
Antigens	42682	8448	66000	8991	3594
Migration threshold	0-2	0-2	0-2	0-2	0-2
DC Population	517	509	2373	402	305
Mature Cell	440	91	861	317	279
Semi-mature cell	76	417	623	5	26
FP detection	14	11	38	4	8
FN detection	0	0	5	3	4
Error rate	3,28%	3,75%	5,21%	2,17 %	4,42%
Subaltern agents created	3	4	3	3	3
FP (agents)	1	2	0	0	0
FN (agents)	0	0	0	0	0
Messages to intermediary	3	2	2	4	3
Frames generated	~200	~133	~133	~266	~200
Time of response	~2s	~2s	~3s	~2s	~2s
Message for superior agent	0	0	0	1	1

VI. Conclusions and Future Works

The use of MAS and AIS-DT as put forward in this paper produced interesting results for intrusion detection systems. The highlights of such combination are: (i) low number of false alarms, especially false negatives, (ii) ability to detect events not known to the system, (iii) simplicity of implementation of automated agents (i.e. without human interference), (iv) low cost of extra traffic upon the network and (v) low response time to anomalous event, mainly suspect of threats. As for the question of scalability it is not a problem anymore since the proposed system capitalized on DT improvements to AIS.

Although quick and having produced no apparent problems in the experiments, the authors find important to highlight that naïve Bayes may present problems in larger networks, especially regarding its data-base update.

The good adaptation proposed here, for the lower layers of IEEE 802.11, is another important result because there are few IDS tackling these layers.

As future work we suggest: (i) test the system in even larger and more complex networks (e.g. networks with two or more APs and hybrid networks), (ii) conceive and test further development of more robust experiments on networks with WPA and WPA2, (iii) application of more signs by category (i.e. modifications in DC algorithm), (iv) application of others alternative techniques for classification of antigens to increase further specificity, (v) incorporation of other type of sensors for analysis of abnormal workstation functioning and hence improve sensitivity, (vi) implementation of more tasks for the superior agent to care and (vii) implementation of counter-attack routines for the subaltern agent, currently out of the scope of this research.

Acknowledgment

The authors thank to Polytechnic School – University of Pernambuco, FACEPE and CNPq (Brazil)

References

[1] J. Greensmith, U. Aickelin, "Dendritic cells for SYN scan detection", *Proceedings of the IEEE Genetic and Evolutionary Computation Conference (GECCO)*, pp. 49-56, 2007.

[2] J. Greensmith, J. Twicross, U.Aickelin, "Dentritic cells for anomaly detection", In *IEEE Congress on Evolutionary Computation (CEC)*, pp. 664-671, 2006.

[3] Y. Al-Hammadi, U. Aickelin, J. Greensmith, "DCA for bot detection", In *IEEE Congress on Evolutionary Computation (CEC)*, pp. 1807- 1816, 2008.

[4] P. Matzinger, "The Danger Model: a renewed sense of self", *Science*, 296, pp. 301-305, 2002.

[5] P. Matzinger, "Tolerance, danger and the extended family", *Annual Reviews in Immunology*, 12, pp. 991-1045, 1994.

[6] U. Aickelin, P. Bentley, S. Cayzer, J. Kim, J. McLeod, "Danger Theory: The link between AIS and IDS", In *Proceedings of the Second International Conference on Artificial Immune Systems* (ICARIS), pp. 147-155, 2005.

[7] M. Lutz, and G. Schuler, "Immature, semi-mature, and fully mature dendritics cells: which signals induce tolerance or immunity?", *Trends in immunology*, 23(9):9911045, 2002.

[8] J. Greensmith, U. Aickelin, G. Tedesco, "Information fusion for anomaly detection with the dendritics cell algorithm", *International Journal of Information Fusion*, 2007.

[9] J. Greensmith and U. Aickelin, S. Cayzer. "Introducing dendritics cells as a novel immune-inspired algorithm for anomaly detection.", In *Proceedings of the Second International Conference on Artificial Immune Systems* (ICARIS), pp. 404-417, 2006.

[10] H. Fu, X. Yuan, K. and N. Wang, "Multi-agents artificial immune system (MAAIS) inspired by danger theory for anomaly detection", In *IEEE International Conference on Computational Intelligence and Security Workshops*, pp. 570-573, 2007.

[11] Korek. Chopchop Theory. At. http://www.aircrack-ng.org/doku.php?id=chopchoptheory, (Accessed in January of 2010).

[12] V.Ramachandran, Md S. Ahmad. "Cafe Latte with a Free Topping of Cracked WEP: Retrieving WEP Keys From Road-Warriors", In *9th Toorcon Hacker´s Conference*, 2007.

[13] Aircrack-ng - Hirte Attack. At. http://www.aircrack-ng.org/doku.php?id=hirte,(Accessed in January of 2011).

[14] M. Danziger, M. Lacerda, F. B. de Lima Neto, "Danger Theory and Multi-agents Applied for Addressing the Deny of Service Detection Problem in IEEE 802.11 Networks," *Ninth International Conference on Intelligent Systems Design and Applications (ISDA)*, pp.695-702, 2009.

[15] S. Russel, P. Norvig, *Artificial Intelligence: a modern approach*, Prentice Hall, 1995.

[16] H. Fu, X. Yuan, K. Zhang, X. Zhang, Q. Xie, "Investigating novel immune-inspired multi-agent system for anomaly detection", In *IEEE Asia-Pacific Services Computing Conference (APSCC)*, pp. 466-472, 2007.

[17] T. R. Mosmman, A.M. Livingstone, "Dendritic Cells: the immune information management experts", *Nature Immunology*, pp. 564-566, 2004.

[18] J. Greensmith, "The Dendritic Cell Algorithm", *PhD Thesis*, University Of Nottingham, 2007.

[19]] I. Rish, "An empirical study of the naive Bayes classifier", *IJCAI 2001 In Workshop on Empirical Methods in Artificial Intelligence*. 2001.

[20] Stewart, B. 2002. "Predicting project delivery rates using the Naive-Bayes classifier", *Journal of Software Maintenance 14*, pp. 161-179, 2002.

[21] Wireshark – *A network protocol analyzer*, At http://www.wireshark.org/ (Accessed in January of 2011).

[22] Aircrack-ng – *802.11 WEP and WPA-PSK keys cracking program*, At http://www.aircrack-ng.org/doku.php (Accessed in January of 2011).

[23] JADE – *Java Agent Development Framework*, At http://jade.tilab.com/ (Accessed in January of 2011).

[24] J. Bellardo, and J. Savagi, "802.11 Denial-of-service attacks: real vulnerabilities and practical solutions", At http://www.cs.ucsd.edu/users/savage/papers/UsenixSec03.pdf (Accessed in January of 2011).

[25] F. Guo and T. Chiueh, "Sequence number-based MAC address spoof detection.", In A. Valdes and D. Zamboni editors, *RAID*, volume 3858 of LNCS, Press Springer, pp. 309-329, 2005.

Author Biographies

Moisés Danziger has a degree in Computing by the Universidade Paulista (UNIP) 1999. In 2004 he completed the postgraduate course in management of electronic commerce in the Faculdade Frassinetti do Recife (FAFIRE) and in 2006 a specialization course on computer networks at the Universidade Católica de Pernambuco (UNICAP). In 2010 he obtained his Master degree in Computing Engineering by the Universidade de Pernambuco (UPE). Danziger has worked for several years as Network Administrator, Security System Analyst and Army Officer (in the technical Corps of Informatics). As researcher, he published some papers on Artificial and Computing Intelligence applied for security. He is enrolled in the doctoral program in Electrical Engineering and Computing at the Universidade Estadual de Campinas (UNICAMP) currently. He has strong research interests in security/cryptography applications using Artificial and Computing Intelligence techniques. Danziger also is a member of research group on Computational Intelligence at UPE and of the Research Group of Advanced Cryptography at UNICAMP.

Fernando Buarque de Lima Neto has a degree in Computing by the Universidade Católica de Pernambuco (UNICAP) 1991. In 1992 he studied Business Administration at the Faculdade de Ciências da Administração da Universidade de Pernambuco (FCAP/UPE). He also holds a Master degree in Computer Science by the Universidade Federal de Pernambuco (UFPE) 1998, and in 2002, he presented to the University of London, England (Imperial College London) his PhD Thesis on Artificial Intelligence & Artificial Neural Networks. Back to Brazil in 2003, he joined in the team that helped to develop the Computer Engineering Programme of Escola Politécnica de Pernambuco (POLI), a faculty of the Universidade de Pernambuco (UPE). In 2004 and 2005 he headed the DSC. Before dedicating himself to academia, that is 1995, Buarque worked for several years as System Analyst, Project Leader, Consultant and Army Officer (in the technical Corps of Informatics). Currently Buarque is an Associate Professor at POLI/UPE where he lectures to undergraduate and postgraduates of Computing. As a researcher, he has published many papers on Artificial and Computational Intelligence. At the moment, Prof. Buarque is the leader of some research projects funded by CNPq (the Brazilian agency that funds scientific research) with objectives of applying AI on decision support in Agriculture, Medicine and oil industry. Dr. Buarque also leads a research group on Computational Intelligence at UPE, he is a member of other research groups and scientific societies both in Brazil and abroad. In 2007, Fernando Buarque was elevated to Senior Member of IEEE (United States of America).