Agent-Based Artificial Immune Systems (ABAIS) for Intrusion Detections: Inspiration from Danger Theory

Chung-Ming Ou[1], C.R. Ou[2], and Yao-Tien Wang[3]

[1] Department of Information Management, Kainan University, Luchu 338, Taiwan
cou077@mail.knu.edu.tw
[2] Department of Electrical Engineering, Hsiuping Institute of Technology, Taichung 412, Taiwan
crou@mail.hit.edu.tw
[3] Department of Computer Science and Information Engineering, Hungkuang University, Taichung 433, Taiwan
ytwang@sunrise.hk.edu.tw

Abstract. Agent-based artificial immune system (ABAIS) is applied to intrusion detection systems(IDS). The intelligence behind ABIDS is based on the functionality of dendritic cells in human immune systems. Antigens are profiles of system calls while corresponding behaviors are regarded as signals. ABAIS is based on the danger theory while dendritic cells agents (DC agent) are emulated for innate immune subsystem and T-cell agents (TC agent) are for adaptive immune subsystem. This ABIDS is based on the dual detections of DC agent for signals and TC agent for antigen, where each agent coordinates with other to calculate danger value (DV). According to DVs, immune response for malicious behaviors is activated by either computer host or Security Operating Center (SOC).

1 Introduction

The Internet has become a major environment for propagating malicious codes, in particular, through web applications. Internet worms spread through computer networks by searching, attacking and infecting remote computers automatically. The very first worm is the Morris worm in 1988; while Code Red, Nimda and other Internet worms have caused tremendous loss for the computer industry every since. Recently, OWASP issued the Top 10 vulnerability list for Web applications [1]; intrusions, in particular, web-based ones, have become increasing threats for information assets. In order to defend against network attacks, one efficient way is to understand various properties of virus, which include the impact of patching, awareness of other human countermeasures and the impact of network traffic, even the ways how these malicious codes reside in a certain hosts, etc.

A. Håkansson & R. Hartung (Eds.): *Agent & Multi-Agent Syst. in Distrib. Syst.*, SCI 462, pp. 67–94.
DOI: 10.1007/978-3-642-35208-9_4

1.1 Related Work

Forrest et al. [2] proposed an instance of computer immunology to protect the computer systems using the principles of human immune systems. Gu et al. [3] proposed an architecture of combining multiagent systems (MAS) and dendritic cells. Bauer et al. proposed an agent-based model for immune systems [4]. Hamer et al. utilized concepts from artificial immune systems to computer security [5]

Most AIS research are focused on the development of specialized AIS algorithms inspired by theories such as the negative selection theory [6] or the danger theory [7]. Applying AIS algorithm to IDS can be traced back to [9] [10]. For reviews of related works, see [11]. Liu et al. [12] proposed an active defense model for IDS based on immune multiagents (IMA). Dasgupta [13] proposed an immunity-based IDS framework which applied the multi-agent architecture. These schemes are based on negative selection theory.

However, Burgess [14] suggests that negative selection algorithm cannot meet the fast evolution of newly generated computer virus. On the other hand, he put the emphasis of AIS on an autonomous, distributed feedback and healing mechanism. Aickelin et al. [9][15] present the first in-depth discussion on the application of danger theory to intrusion detection. Greensmith et al. [16] employed dendritic cells (DCs) within AIS which coordinated T-cell immune responses. Kim et al. [17] proposed "CARDINAL" which embedded T-cell process within the danger-theory-based AIS. Sarafijanović and Boudec [18] proposed the concept of virtual thymus (VT).

1.2 The Motivation of Emerging AIS and MAS to IDS

Intrusion detection systems, whether utilizing statistical analysis, feature analysis or data mining, have limitations such as self-adaptation, robustness and effective communications. IDSs cannot respond to unknown attacks as they lack self-adaptations. Without robustness, components of IDSs will be isolated each other [12]. On the other hand, if there is no effective communications among components of IDSs, the early warning and response mechanism cannot be established.

AIS can be contributed to the improvements of self-adaptation, moreover, the diversity and memory mechanism of the IDSs. On the other hand, robustness and communications can be improved by mechanisms of multiagent systems. The logical structure of agents with internal reasoning mechanism can effectively solve problems arisen from complicated environment such as complex networks.

The motivation of this research is the following. Can agent-based information security systems learn themselves to effectively determine whether the abnormality is "actually" incurred by some malicious attacks. This paradigm is very similar to the danger theory proposed in the immunology [7]. It leads to the immunity-based multiagent system (IBMAS), which is a promising research topics these days [19].

1.3 Contributions of This Research

This paper combines the advantages of both AIS and MAS to design a better IDS. Several researches related to immunity-based IDS can be further improved by introducing multiagent architectures and utilizing agents' cooperations and communications [20]. This research analyzes some issues of adopting immunity-based IDSs without clarifications of roles of involved systems components. On the other hand, the improvement can be reached by defining role of each agent and the coordinations of these roles.

However, for network infrastructure, there is no entities "on the fly", which means there is no definitive concept for computer hosts "between" two adjacent nodes. This paradigm is quite different from that of human immune system, unless we consider the mobile agents which transit between computer hosts. It is a research area how to identify malicious codes and mobile agents. To be more concentrate is this research, we propose a solution which combines MAS and AIS.

1.4 Approach

The major goal of this paper is to facilitate intelligent agent mechanisms with AIS based on danger theory to improve intrusion detection systems. Such improvements are proposed according to previous researches and architectures such Fu et al. [21] and Kim et al. [22]. Intelligent agents are also required while IDS is considered deploying in the cloud computing environment. For agent-based AIS (ABAIS), these agents embodied the dentritic cell functionality (namely, DC agents) can "detect" danger signal issued by computer hosts being attacked or suspiciously being attacks. While other agents, such as T-cell agent (TC agent), antigen agent (Ag agent) and responding agents (RP agent), communicate one another to improve the efficiency of IDSs. Computer threats generally come from the Internet, which are very similar to those of pathogens to our bodies. The central challenge with computer security is how to discern malicious activities from benign ones. Our intelligent agent-based model majorally consists of the cooperation of DC agents in the innate immune system and TC agents in the adaptive immune system. This dual detective mechanism, where DC agent detects the behavioral information (i.e. signal) caused by an antigen and TC agent detects system call (i.e. antigen), can decrease false positive rate. For the learning process of intelligent system, self-organizing feature map (SOM) network effectively clusters the input (normal) network vectors, while maintaining the topological structure of the input space. ABIDS enables tunable and adaptable threshold values to determine danger signals.

1.5 Structure of This Chapter

The arrangement of this chapter is as follows. In section 2, preliminary knowledge such as AIS, danger theory, intelligent agents system and intrusion detections are introduced. In section 3, ABIDS model inspired by the danger theory is discussed.

Simulations of algorithms related to ABIDS based on different scenarios will be given in section 4. Further analysis of ABIDS will be discussed in section 5.

2 Preliminary Knowledge

2.1 Intrusion Detection Systems (IDS)

Intrusion detection systems (IDS) focus on exploiting attacks, or attempted attacks, on networks and systems in order to take effective measures based on the system security policies, if abnormal patterns or unauthorized access is being suspected. However, there are two potential mistakes by IDS, namely, false positive error (FPE) and false negative error (FNR). For FPE (FNE), a pattern is mistakenly determined as abnormal (normal).

IDSs are used to help protect computer systems. The main goal of IDSs is to detect unauthorized use, misuse and abuse of computer systems by both systems insiders and external intruders [11]. An IDS can be a device or software application that monitors network and/or system activities for malicious activities or policy violations and produces reports to a Security Operating Center (SOC). Kim and Bentley suggested IDSs should satisfy the following seven requirements: robustness, configurability, extendibility, scalability, adaptability, global analysis, and efficiency, for more details see [22][24].

There are two types of IDSs, host-based and network-based intrusion detection system, respectively [23].

Network intrusion detection system (NIDS). It is a platform that identifies intrusions by examining network packets and monitors multiple hosts. Network intrusion detection systems gain access to network packets by connecting to a network hub. Sensors are located at choke points in the network to be monitored; they capture all network packets and analyze the content of individual packets for malicious traffic.

Host-based intrusion detection system (HIDS). It consists of an agent on a host that identifies intrusions by analyzing system calls, application logs, file-system modifications and other host activities and state. Sensors usually consist of a software agent.

2.2 Human Immune System (HIS) and Artificial Immune Systems

Human immune system (HIS) consists of antibodies and lymphocytes, which include varied T-cells and B-cells. HIS uses a large number of highly specific B- and T-cells to recognize antigens. Only B-cells secrete antibodies. Clonal selection theory explains the details of antibody secretion specific to an antigen where T-cells help regulating. The binding between antigen and specific lymphocytes trigger the proliferation from immature lymphocytes to mature one, and the secretion of antibodies. HIS must interact not only with the nonself from the outer world, but also the self from the internal world.

HIS can be categorized as innate and adaptive immune system. The innate immune system is characterized by three roles, namely, host defense in the early stages of infection, induction of the adaptive immune response and determination of the type of adaptive response through antigenic presenting cells (APCs) [25]. On the other hand, the main characteristics of the adaptive immune system are recognitions of pathogens.

Dendritic Cells. Dendritic cell (DC) is a vital link between the innate and adaptive immune system which provides the initial detections of pathogenic invaders. DC is also an APC which captures antigen proteins from the surrounding area and processes it by ingesting and digesting the antigen. DCs are also a part of innate immune system; once activated, they migrate to the lymphoid tissues where they interact with T- and B-cells to initiate the adaptive immune response. Moreover, adaptive immune response is "orchestrated" by DCs.

DCs are the first defense line for HIS which will arrive at the locations where antigens intrude and then swallow the latter to the pieces. These pieces will be attached to APCs and presented to the T-cells. DCs can also be regarded as the commanders for HIS; they can combine the danger and safe signal information to decide if the tissue environment is in distress or is functioning normally. DCs will influence the differentiation of T-cells by releasing particular cytokines. In other words, DCs drive the T-cell to react to the antigen in an appropriate manner.

T-Cell. T-cells coordinate hosts not only by way of Lymph nodes, but also by periphery and Lymph nodes within each host. DCs will influence the differentiation of T-cells by releasing particular cytokines. In other words, DCs drive the T-cell to react to the antigen in an appropriate manner.

Naive T-cells are those have survived the negative and positive selection processes within the thymus, and have migrated to circulation system between the blood and lymphoid organs where they wait antigen presentation by DCs. Naive T-cells reach an activated state when the T-cell receptor (TCR) on their surfaces binds to specific molecules, namely, antigen-peptide-Major Histocompatibility Complex (MHC), on the surfaces of DCs', and costimulatory molecules are sufficiently upregulated on the surface of the DCs to show the degree of danger signals. Activated T-cells proliferate and their clones will differentiate into other cells such as helper T-cells and cytotoxic T-cells (CTL). Differentiation statuses of T-cells play several roles in immunity mechanisms and tolerance in the HIS.

Artificial Immune Systems (AIS), based on HIS, have been applied to anomaly detections [3][12][21][26][27][28]. AISs have been developed according to negative selection algorithm and clonal selection algorithm which are based on the classical self-nonself theory; nonselfs are entities which are not part of human organisms [29]. This so-called self-nonself classification theory had been challenged while failing to explain several immunological phenomena. Some alternative theories have been proposed, for example, the danger theory (DT). DT postulates

that the human immune systems respond to the presence of molecules known as danger signals, which are released as results of unnatural cell deaths.

2.3 Computer Security and Immune System

Computer security is composed of processes to prevent malicious programs such as computer virus, internet worms, etc. to invade computer hosts. An important aspect is the following: a computer security system should protect a host or network of hosts from unauthorized intruders, which is analogous in functionality to the immune system protecting the body from invasions by foreign pathogens. Therefore, it is a straightforward consideration to adopt human immune mechanism to computer security.

For computer security, one important aspect learned from immunology is the following: a computer security system should protect a host or network of hosts from unauthorized intruders, which is analogous in functionality to the immune system protecting the body from invasions by foreign pathogens. For example, anti-virus software has recently adopted some features analogous to the innate immune system, which can detect malicious patterns. However, most commercial products do not yet have the adaptive immune system's ability to address novel threats. According to [30], the central challenge with computer security is to determine the difference between normal and potential harmful activities. IT systems are getting larger and more complex, which invokes the needs of developing automated and adaptive defense systems. One promising solution is to acquire AIS.

In general, HIS maintains both low false positive and false negative errors. The former guarantees our immune system can correctly "recognize" harmful pathogens, while the latter for harmless pathogens. This leads to the motivation of AIS-based intrusion detection systems. For example, MHC stimulates antigen presenting cell (APC) to activate, which helps lymphocyte cells identify antigens. Boukerche et al. [31] proposed a mapping between computer security and HIS.

Fig. 1 illustrates a basic architecture of secure computer network. some SOC issues threat profile for each host computer to determine whether a network packet is malicious. It also collects data from each computer host and monitors network behaviors within its security domain. On the other hand, proxy server is in charge of controlling inward and outward network traffic of each computer host. It can be a firewall, an application gateway, or any security gateway. Fig. 1 also suggests an architecture of distributed IDS, which will be discussed later.

For efficient evaluations of IDS, Kim et al. [28] proposed three conditions for an "intelligent" IDS.

1. Optimize the number of peer hosts polled.
2. Types of system response should be determined by attack severity and certainty
3. For performing adequate magnitudes of responses, both local and peer information needs to be taken into account.

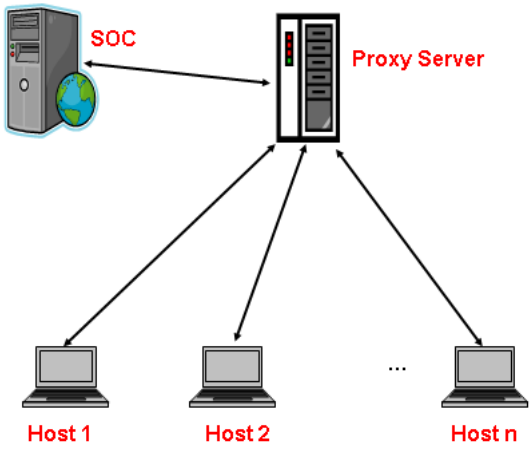

Fig. 1. Architecture of Secure Information System

2.4 Danger Theory

Matzinger [7] proposed the Danger Theory (DT), which has become more popular among immunologists in recent years for the development of peripheral tolerance (tolerance to agents outside of the host). DT states that the immune system will only respond when damage is indicated and is actively suppressed otherwise. It proposes that APCs, (in particular, DCs), have danger signal receptors (DSR) which recognize signals sent out by distressed or damaged cells. These signals inform the immune systems to initiate immune responses. APCs are activated via the danger signals. These activated APCs will be able to provide the necessary signals to the T-cells (more precisely, T-helper cells) which control the adaptive immune response.

Danger signals are generated by ordinary cells of the body that have been injured due to attacks by pathogens. These signals are detected by DCs, which have three modes of operations: immature, semi-mature and mature. In the DC's immature state, it collects antigens along with safe and danger signals from its local environment. DC is able to integrate these signals to decide whether the environment is safe or dangerous. If it is safe, DC becomes semi-matured. Upon presenting antigens to T-cells, DC will cause T-cells to "tolerate". If it is dangerous, DC becomes matured and causes T-cells to become reactive on antigen-presentations.

2.5 Relationship between Multiagent System and AIS

Agent is an entity that has the ability of consciousness, solving problem, self-learning and adapting to the environment, which is very similar to immune systems in functionalities. King et al. [8] proposed an architecture intelligent agents based on AIS. Table 1 summaries such similarities between AIS and MAS.

Table 1. Comparisons between Multiagent Systems and AIS

System	MAS	AIS
Diversity Generation	Yes	Yes
Self Tolerance	Yes	No
Learning	Yes	No
SNS Act. Threshold	Yes	Yes
Self Maintenance	Yes	Yes
Short-term Memory	Yes	Yes
Long-term Memory	Yes	No

On the other hand, similarities of multiagent system and AIS, based on perspectives from intelligent IDSs [22][24], are given as Table 2. According to this table, AIS-based MAS (ABMAS) and Agent-based AIS (ABAIS) are two (loosely coupled) categories for designing intelligent IDS.

Table 2. Comparisons of AIS and Generic MAS

Property	AIS	MAS
Robustness	Yes	No
Configurability	Yes	Yes
Extendibility	No	Yes
Scalability	No	Yes
Adaptability	Yes	No
Global Analysis	Yes	No
Efficiency	Yes	No

Agent-Based Models (ABM). ABM has inspired significant interest as agent-based language is very similar to that of nature. According to [32], ABM is an appropriate method for studying immunology. As computers became more powerful and less expensive, the ABM became a practical method for studying complex systems such as the immune system. The interaction between various types of agents is a criterion to evaluate a multiagent system. For example, "Autonomous Agents for Intrusion Detection" (AAFID) is the first agent-based IDS proposed by Purdue University [33].

To have the agents learn, we may utilize methodologies derived from AIS such as (immune) response attributes of specificity, diversity, memory and self/non-self recognition. Functionalities of the biological immune system such as content addressable memory and adaptation, are identified for use in intelligent agents.

AIS-Based Multiagent Systems (ABMAS). MAS is suitable for task allocation in heterogeneous computing environment, which become a major characteristics for Internet nowadays, in particular for cloud computing. Adaptiveness is a challenge and also an important feature for multiagent system to interact

with the environment. Three major stages for ABMAS inspired by the clonal selection theory are diversity generation, self-maintenance and memory of nonself. The last two properties define the adaptiveness of the ABMAS. These steps are carried out by agents distributed over the MAS.

Diversity Generation. (Continuous) diversity generation leads to the "adaptation" of ABMAS. Diverse agents with distinct specificity of the receptor and the effector are generated by way of mutations.

Self-Maintenance. Agents are adjusted to be insensitive to known patterns (self) during the developmental phase. Negative selection theory is a central of this phase.

Memory of Nonself. Agents are adjusted to be more sensitive to unknown patterns (nonself) during the working phase.

CARDINAL. CARDINAL (Cooperative Automated Worm Response and Detection Immune Algorithm) is an immunity-based worm detection inspired by T-cell immunity and tolerance [17]. Although it is not an agent-based architecture, CARDINAL proposed an seamless integration between artificial periphery and lymph nodes which emulate the functionalities of human immune systems, in particular, those of the innate and adaptive immune systems. Mechanisms of T-cell immunity and tolerance provide intelligence of performing criterion of intelligent IDS, see Fig. 2.

CARDINAL also adopts concepts from danger theory. Accordingly, DCs present results of danger signal assessments in three different forms to naive T-cells. The costimulatory signal is increased if a DC detects a severe attack; a strong response to this attack is needed. From HIS, the amount of cytokine IL-12 is increased when a DC detects a severe attack requiring a strong response but

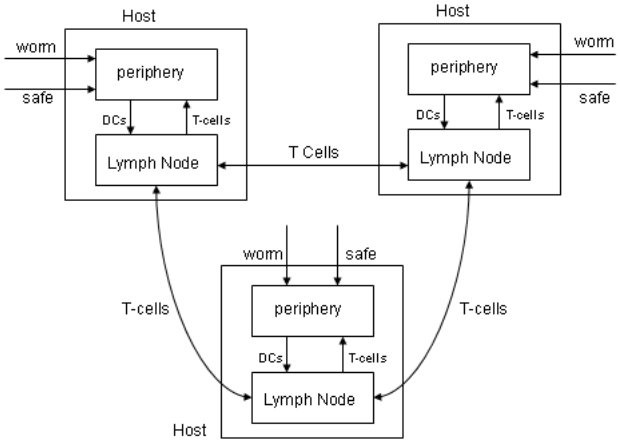

Fig. 2. Architecture of CARDINAL

with a relatively lower certainty. On the other hand, the amount of the cytokine IL-4 is increased when a DC detects a less severe attack which only needs a weak response. CARDINAL does not consider the certainty of attack as a negative effect of a response triggered by a false positive error would be minor. However, it is still an issue worth being studied. One disadvantage of CARDINAL is the assumption that it will target the internet worms with consistent attack signatures. We still think it is a strong hypothesis which should be loosen.

MAAIS. MAAIS (Multiagent-based AIS) was proposed by Fu et al. [21]. The architecture is based on CARDINAL with "concepts" of multiagents. This MAAIS, which consists of two components, namely agents and server processes, provides agent-based anomaly detection functionality. Agents monitor their corresponding hosts; while servers evaluate to select suitable strategies ro respond according to immune response mechanism of immune responses, see Fig. 3. However, [21] did not designate roles of agents specifically. For further improvements, agents may be composed of diverse APCs and various types of artificial T-cells.

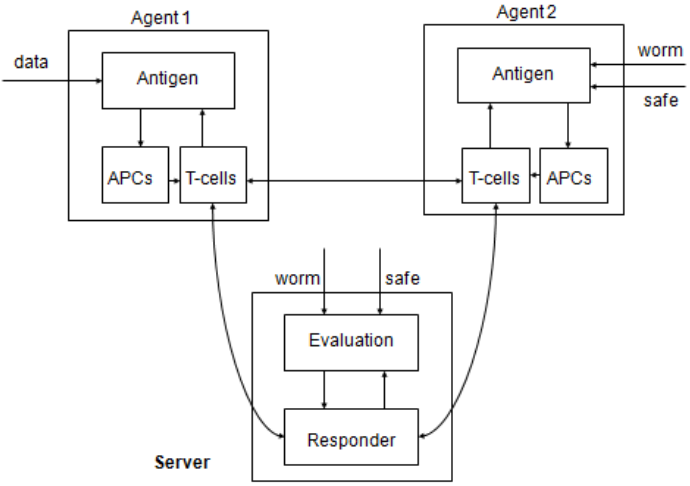

Fig. 3. Architecture of MAAIS

MAAIS is also adopted the popular danger theory from immunology. APCs are sensing danger signals from their hosts rather than identify self and nonself network packets. The intelligence behind this anomaly detection mechanism is abstracted from Dendritic Cell Algorithm (DCA) which will be introduced later.

3 Intrusion Detection Mechanism Based on ABAIS

According to [21][24][31], we propose improved IDS based on MAAIS. Different from traditional self-nonself paradigm for immune systems, our IDS will first detect danger signals emitted by computer hosts. These danger signals are based on some security threat profile, which defines by system calls generated by running processes. According to [27][31], threat profile may be composed of excessive CPU, memory load at the host, bandwidth saturation and high connection number of the host, etc. This threat profile defines what the "event" is. For example, the unknown antigen invoking malicious behaviors such as file deletion, information leakage, etc. In this IDS, several agents are generated which can communicate each other to emulate functionalities extracted from DT-inspired AIS.

3.1 Antigen

For most network detecting systems (for example, those adopting self-nonself paradigm), an antigen is defined as an information vector extracted from network packet. The extracting rule can be very different for each network detecting systems. Moreover, antigens are binary strings extracted from the IP packets, which include IP address, port number, protocol types, etc. However, the antigen defined at our IDS is related to system calls rather than network packets.

A system call is the way how a program requests a service from an operating system's kernel. System calls provide the interface between a process and the operating system. Each process invokes some system calls. The more active the process, the more system calls it makes. Each system call is captured and converted into an antigen attribute [34]. For example, the latter can be represented by CPU usage, memory load at the host, bandwidth saturation and connection number of the host

3.2 Intelligent Agents in AIS

Our ABIDS, which is different from, for example, that of [25], is based on the danger signal rather than self-nonself paradigm. Therefore, we design a MAS with antigen agents, DC agents, T-Cell agents and Responding agents to perform functionalities of IDSs'.

Antigen Agent (Ag Agent). Antigens are profiles of input data such as IP packet, which includes IP address, port number, protocol type, and network connection, etc. Ag agent simply parses input data into format of antigens then sends them to the DC agents.

Antigen agents, which are installed at computer hosts, represent data item from the nonself dataset. They extract and record selected attributes from these data items. As one Ag agent samples multiple times, each antigen agent randomly selects certain amount of DC agents and sends those DC agents a picked message when a nonself antigen appears.

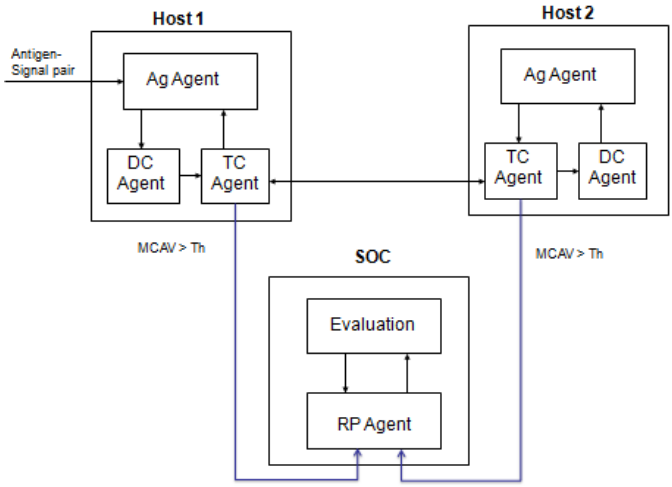

Fig. 4. Architecture of ABIDS

Dendritic Cell Agent (DC Agent). In order to determine whether a malicious behavior is taking place, IDS needs to analyze the input data, i.e., antigens. DC agents are the kernel of the ABAIS which are complex compared to other agents; they are also installed and distributed at each computer hosts. When an Ag agent issues a picked signal, DC agent will evaluate the risk state facing by the host by calculating the danger value (DV) and analyze the signal corresponding to this antigen. Once the DV exceeds some threshold, it will inform the responding installed at the SOC.

Similar to nature DC, each DC agent has three stages, namely immature, semimature and mature. DC agents are started from the immature stages. When a picked signal issued from the Ag agent, DC agent executes data processing function such as the DCA. When a DC agent is at either semimature state or nature state, it returns the mature context to the Ag agent.

T-Cell Agent (TC Agent). TC agents are also installed at each computer hosts. They are activated by the signals from DC agents when the DVs exceed thresholds. Each TC agent has three numerical values associated with it; these represent the accumulated certainties and severities of attack: *T-cell activation threshold, Th1 activation threshold, Th2 activation threshold*. These TC agents will communicate with each other to update these numerical values. There are two perspectives of this agent communications.

– For TC agent issuing warning signal of malicious act to the corresponding antigen, it also informs TC agents "nearby" by exchanging its (three) numerical values.

– For TC agent not issuing warning signal of malicious act for this antigen, it simply updates its numerical values according "nearby" TC agents which have issued warning signals.

This paradigm of TC agents is very similar to that of ensemble neural network (NN), where training NN can be more precise by way of combination of a number of individual networks trained on the same task.

Responding Agent (RP Agent). Ag agents, DC agents and TC agents are coordinating one another to perform immune responses. After DV exceeds threshold value, TC agents will inform RP agent, which is installed at some SOC. Therefore it has all these resources to compare the output category with the original category of each antigen, to calculate the overall true positive or accuracy of the virus detections. It represents that an infected host is detected; RP agents will activate some control measure to such malicious action. Two measures are considered [27]:

1. Reporting to the SOC or security manager, for example patches downloaded, activate relevant anti-virus software on this infected host and removes virus.
2. Disruption of intrusion, discards a suspicious packets, kill the related process, cut-off infected sub-network. These can prevent large-scale spreading of computer viruses, in particular internet worms, which have high spreading rate by their natures.

Table 3 is the comparisons for goals and services fulfilled by these agents.

Table 3. Agents, Goals and Services

Agent	Goals	Services
Ag Agent	Data Parsing	antigen label and signal association
DC Agent	Detect malicious codes or network attacks at the input sources	MCAV estimation, update activation threshold
TC Agent	Categorize nonself antigen	Identify the malicious antigen
RP Agent	Reduce the malicious antigen	Implement the system response

3.3 Threshold Values of Intrusion Behavior

Functionalities of the IDS based on ABAIS are heavily depended on determining whether signals issued by computer hosts are dangerous; that is, IDS should have high positive rate of detecting "real" intrusion behaviors. In details, there is an effective threshold values for signals to be most likely intrusion signs. We adopt the Self-Organizing Feature Map (SOM) network proposed as a basis of threshold value generation.

Let $\{X_i\}_{i=1}^{N}$ be a collection of N testing network vectors consisting of network information such as (attack) starting time, (attack) time duration, protocol ID, source port, destination port, source address, destination address and attack profile, etc. Let F be a fixed SOM with input vectors, whose competitive layer is a matrix. In particular, this matrix is tunable by the SOC if necessary. Then the SOM network is trained by these input vectors X_i; the latter can be classified into several clusters afterwards. These clusters can help establish the baseline of normal network behaviors, namely, the threshold values of intrusion behaviors. When a network vector falls into some deviated classes, intrusion signal is issued by some intelligent agent such as DC agents. The following figure is a graphic representation of threshold values generation.

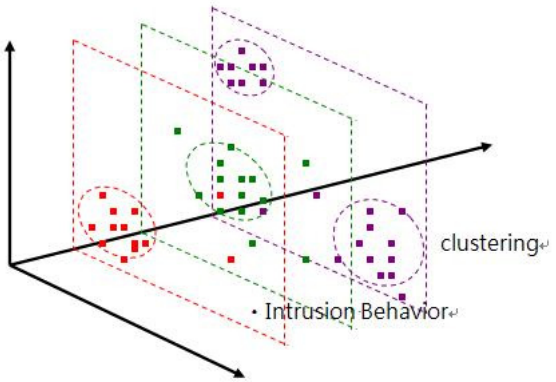

Fig. 5. Intrusion behavior defined by SOM Network

3.4 Agent-Based Dendritic Cell Algorithm (ABDCA)

DCA is an AIS algorithm which is particularly developed for anomaly detection according to the above DCs' functionalities and characteristics [34][35]. It provides information how anomalous a group of antigens is. This is achieved through a generation of an anomaly coefficients, namely, *mature context antigenic value (MCAV)*. It is believed that a DC is better performed by agent technology while considering its adoption to network environment. This antigen-plus-context information is passed on to a class of responder cells, termed T-cells.

DC agents in our ABIDS will evaluate antigens and corresponding signals according to the DCA to determine whether antigens are malicious nor not. The "context" means the classification identifier for an antigen. If an antigen is collected in an environment of danger, the context of this antigen is marked as anomalous and such antigen collected by the DC agent is potentially an intruder. While in the immature state (namely, initial phase), DC agent performs the following three functions:

1. Antigen Sampling: DC agents collect antigens from some external sources (in this case, from computer hosts) and places these antigens in its own antigen storage.
2. Update input signals: DC agent collect values of all input signals present in the signal storage area.
3. Calculate interim output signals: each DC agent calculates its three temporary output signal values from the received input signals, then derives its final output signal.

We observe that multiple DCs present multiple copies of the same antigen type for invoking an immune response; it leads to an error-tolerance immune system as a single DC is far from stimulating a false positive error. The ABDCA is listed as follows.

Algorithm 1. Agent-based DCA (ABDCA)

```
input : Antigens and signals
output: Antigen context (0 for safe/1 for danger)
Initialize DC agents (Immature state);
for each DC_agent_i do
   get antigen (Ag);
   store antigen;
   get input_signal;
   calculate output_signal_i;
   if output_signal_i > Activation_Threshold_i then
      Ag_context is assigned as 0;
      State of DC_agent_i="semi-mature";
   else
      Ag_context is assigned as 1;
      state of DC_agent_i="mature";
   end if
end for
update cumulative output_signals;
```

Three temporary output signals are PAMP signal, danger signal and safe signal. According to Greensmith et al. [34], the system should respond with a very high rate of false positives by switching the PAMP and safe signal. The definitions of PAMP, danger and safe signals, by way of parameters of computer hosts such as CPU usage, memory load, network connection (number) and bandwidth saturation, are as follows.

1. PAMP Signal: Network Connection > th_netconnection AND bandwidth Saturation > th_bandsaturation.
2. Danger Signal: CPU Usage > th_cpuload OR Memory Load > th_memoryload.
3. Safe Signal: All parameters < th_parameter.

Where "th_parameter" represents the threshold value of the parameter. Now the computation of output signals by three temporary output values is the following.

$$output\_signal = \frac{W_P \cdot C_P + W_D \cdot C_D + W_S \cdot C_S}{W_P + W_D + W_S} \quad (1)$$

According to empirical experiments [35], the weights for the output signal is $W_P = 2$, $W_D = 1$, $W_S = 1$. C_P, C_D, C_S represent PAMP, danger and safe signal, respectively. The principle for these weights is the following: PAMP signal will decide whether the danger signal is a "really" harmful one; whether output signal is harmful or nor is defined by the Table 4.

Table 4. Definitions of Output Signal

Output Signal	C_P	C_D	C_S
Normality	0	0	1
Harmless Abnormality	0	1	0
Harm Abnormality	1	1	0

Threat Profile. One issue of ABIDS is the "baseline" for determining abnormality of network packets. Such baseline, namely the *threat profile*, provides intelligent determination of attack types of network packets. It can be determined by three factors, namely, attack severity (S), certainty (C) and the length of attack time (T) [21]. There are different aspects of estimating S, C and T. From network detection viewpoints, these factors are functions of Ag agent attribute (CPU_usage, memory_load, bandwidth_saturation, connection_numbers). S, C, and T are normalized, namely, $S, C, T \in [0, 1]$. Threat profile is a vector $\langle W_S, W_C, W_T \rangle$, where W_S, W_C, W_T are weighted factors of S, C, T respectively.

Event-Driven Architecture. If a network packet is not equal to a pattern or signature that the system has already stored, the current pattern-based architecture does not detect it as abnormal. This happens for a virus with mutation capability. In general, IDSs cannot detect such transformations. Such inflexibility leads to the consideration of event-driven architecture for IDSs. One important factor for such improvement is the ***certainty***. In order to improve the false negative rate, we have to discard all detected events whose attack certainties are below a threshold value. The following is a definition of an attack.

Definition 1. *An incident for a network domain is called an "attack", if its antigen attribute is greater than its corresponding threshold value.*

Technologically, incident of attack can be determined by some rule-based methodologies such as the attack graph methodology based on system call sequence. In order to reduce both the false positive and false negative rates, the decent ABIDS should cooperate both the danger theory and attack graph. The paradigm of the latter is out of scope of this research.

Algorithm: Agent-Based IDS. Now according to ABDCA, we propose an algorithm describing agent-based IDS. This algorithm is also illustrated as Fig. 6. ABDCA can provide information not only a network packet but also a group of network packets is anomalous or not. This is achieved by the generation of an anomaly coefficient value, namely MCAV.

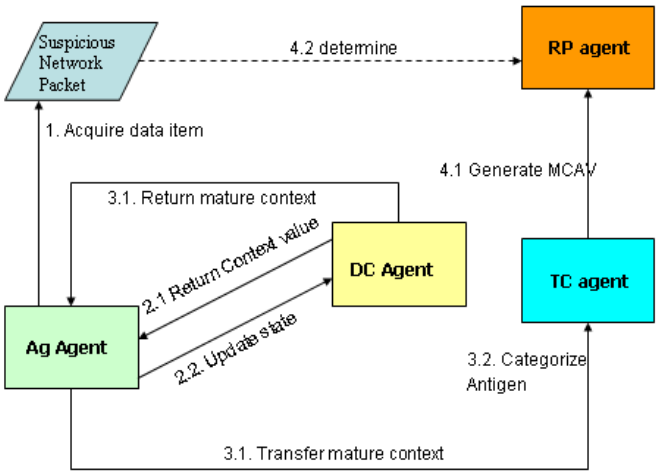

Fig. 6. Diagrammatic Illustration of Agent-based intrusion detection system (index numbers are referred to Algorithm 2

For the step 4 of the Algorithm 2, each nonself antigen gets a binary string of mature contexts from every selected DC agent installed at corresponding computer host. MCAV can be calculated through the number of context "1" divided by the number of all contexts. According to [3], it is similar to a voting system, where the antigen is the candidate and the DC agents are the voters. If the context is "1" ("0"), it means the DC agent determines this antigen is malicious (benign). The MCAV is actually the probability of that this antigen is being malicious.

The reactions of RP agents can be multiple according to the SOC's security policies. For example, passive reaction is initiated by sending some alarm signal such as an e-mail to the administrator and a strategy proactive is defined through mobile agents that implement the characteristics of the reaction. According to [31], for example, an applied reaction is defined for each of the following internet protocols such as DNS, FTP, HTTP, POP3 and SMTP. The issue of RP agents is out of the scope of this research.

3.5 Functionalities of ABIDS

Security Response by SOC. SOC plays the central role of security responding mechanism. Once MCAV exceeds some threshold, RP agent installed in the

Algorithm 2. Agent-based IDS (ABIDS)

Initialization: Some network packets are suspicious as malicious.
Input: Antigen-Signal pair
Output: Antigen Type
1. Data Processing:
1.1 Ag agent extracts an antigen from Antigen-Signal pair.

2. Agent response:
2.1 DC agent returns its context value according to ABDCA; also according to the policy of the SOC if necessary.
2.2 DC agent responds to this antigen by updating its state according to ABDCA.

3. Danger Signal processing:
3.1
if this DC agent returns mature values to Ag agent **then**
Ag agent transfers it to the TC agent.
end if
3.2 TC agent categorizes such antigen according to the threat profile.

4. MCAV generation:
4.1 TC agent generates MCAV, which is sent to the RP agent.
4.2 RP agent determines if the corresponding antigen is malicious or not.

SOC is activated and makes a comprehensive evaluation for the received danger signal. This evaluation is a crucial factor to mitigate the threat of the whole network. If the evaluation is not good enough, the false positive error will produce the damage equal to the one caused by the attack itself. This observation, coincidentally agreed with the paradigm of danger theory, suggests that the comprehensive evaluation should be depending not only on the danger signal, but also the number of DC agents emitting danger signals.

Let CE represent the comprehensive evaluation, AVE is the average value of DV exceeding threshold value T; n is the number of DC agents emitting danger signals, N is the total number of DC agents. The calculations of CE and AVE are as follows; Table 5 is the suggesting weights W_A and W_n.

$$CE = \frac{W_A \cdot AVE + W_n \cdot (n/N)}{W_A + W_n} \tag{2}$$

$$AVE = \frac{1}{n} \sum_{i=1}^{n} MCAV_i \tag{3}$$

$CE, AVE \in [0, 1]$. There will be a critical task for SOC to define its security measures according to CE values. This issue is also out of our scope here.

Agents' Communications and Coordination. One advantage of ABIDS is the communications and coordination between DC and/or TC agents from

Table 5. Weights related to CE

W_A	W_n
2	1
1	1
1	2

corresponding hosts. The basic idea is as follows. TELL and ASK-based communication permit agents share their internal information, enhancing their performances to respond to intruders [21]. On the other hand, useful knowledge obtained by each agent should be stored in a database. These databases could also be shared by each agent to improve their detection efficiencies. Some suitable mechanism of agent communication between agents can contributed to better performance of ABIDS. Table 6 illustrates the agent communications [5].

Table 6. Agents' Communications

Communication	**Initiator**	**Receiver**	**Description**
eRaiseWarning	Ag Agent	(Selected) DC agents	Notify of a possible malicious attack
eContextReturned	DC Agent	TC Agent	sending MACV
eMaliciousOrNot	DC agent	TC agent	Identify the malicious antigen
eEvidenceResponse	TC Agent	RP agent	Implement the system response according to TC agent's information

4 Simulation

4.1 Malicious Behaviors Determined by ABDCA

According to essence of the danger theory, the advantage of ABIDS is the following: it can determine some nonself-antigenic behavior which is at the verge of normality and abnormality. For example, for those network behaviors with short attacking time. In this section, we simulate several critical cases for this ABIDS. The threshold of S, C and T are the following $S_{th} = 0.50$, $C_{th} = 0.50$ and $T_{th} = 0.5$. The number of computer hosts within this network is 2000. Each host has its fixed vector of weights. Simulation principle is to determine those critical behaviors, which include one extremely high factor with other low factors, or some factors are closing to the corresponding threshold values.

Nonself Antigen with High Severity, Low Certainty and Short Attacking Time. While an antigen induces a computer host with behaviors of high severity but low certainty and short attacking time, it is difficult sometimes to determine if this is a malicious attack. In this case, we consider the threat profile TP=$[0.9, 0.01, 0.01]$ for the ABDCA algorithm. We expect the simulation should be "stable" for reasonable many hosts as Fig. 7. MCAV is around 0.03 and 0.05, which shows that this nonself antigen is normal.

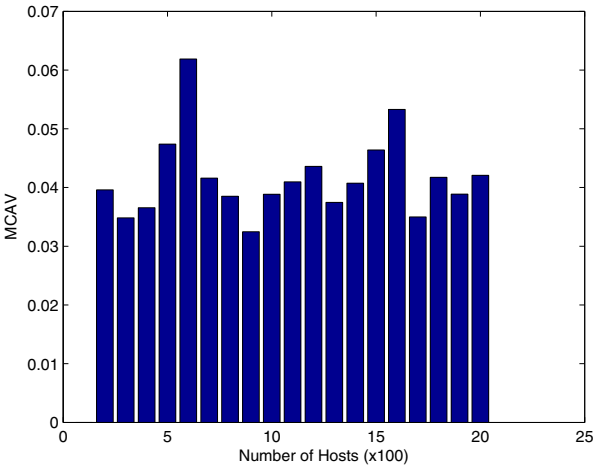

Fig. 7. Average MCAV for an Antigen with High Severity solely

Nonself Antigen with Edge Behavior. While an antigen induces a computer host with edge behavior such that severity, certainty and attacking time are all closing to their threshold values, respectively. It is also difficult to determine if this is a malicious attack or not. In this case, we consider TP=$[0.65, 0.65, 0.65]$. The simulation also show a stable result (see Fig. 8); average MCAV is equal to 1, which indicates that this nonself antigen is harmfully abnormal.

Nonself Antigen with Short Time Attack. In this simulation, an antigen causes computer hosts with both high severity and certainty, but relative medium attacking time. It is difficult to determine if this is a malicious attack or not. In this case, we consider TP=$[0.9, 0.9, 0.4]$. The simulation show a stable result (see Fig. 9); average MCAV is around 0.5, which indicates that this nonself antigen can be harmfully or harmful abnormal depending on the SOC's security profile.

Antigen with Long Time Attack. Another interesting network behavior is the following: the connection time is relatively long but with medium severity and certainty. The following simulation is for antigen profile $TP = [0.3, 0.6, 0.8]$.

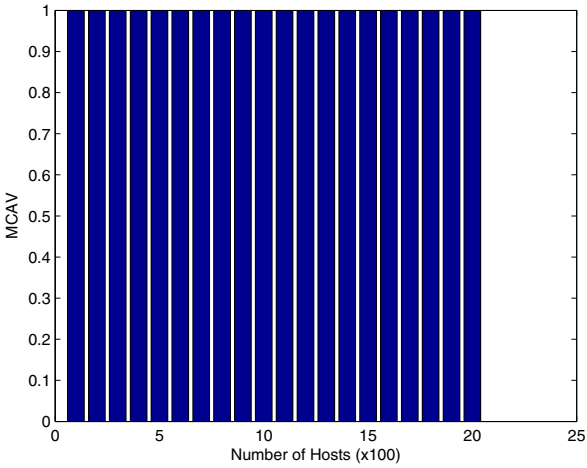

Fig. 8. Average MCAV for an Antigen with Critical Threshold Behavior

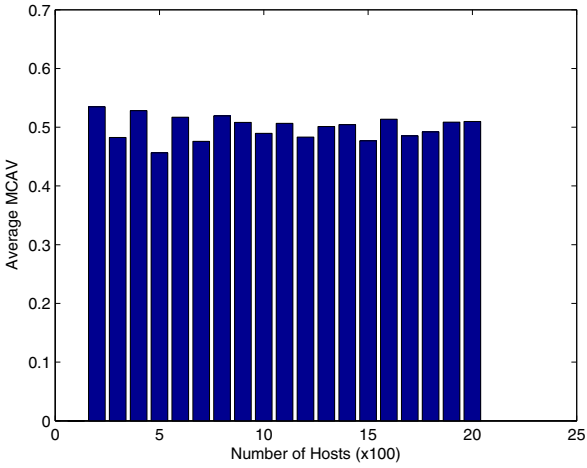

Fig. 9. Average MCAV for an Antigen with High Severity and certainty, but relatively short Attacking Time

According to Fig. 10, the average MCAV is stably around 0.4 and 0.5. This nonself antigen can be harmfully or harmful abnormal depending on the SOC's security profile.

4.2 Simulation with Costimulation Signals

In this subsection, we will concentrate on the actual host behaviors rather than threat profile. These host behaviors (HB) include CPU usage, memory load,

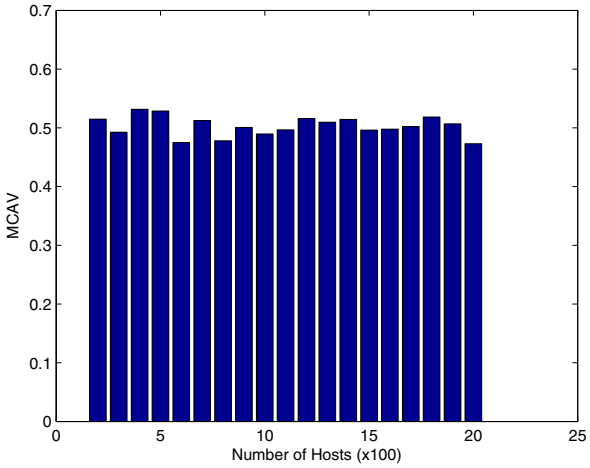

Fig. 10. Average MCAV for an Antigen with Relatively High Attacking Time

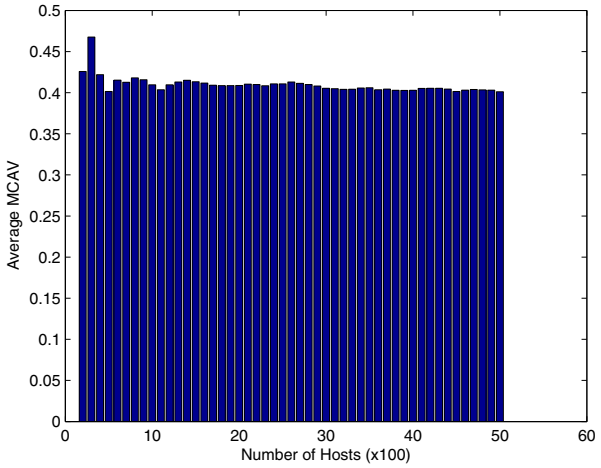

Fig. 11. Average MCAV for an Antigen causing host behavior $HB = [0.95, 0.8, 0.8, 0.3]$

network connection (numbers) and bandwidth saturation. We first consider the following host behavior: $HB = [0.95, 0.8, 0.8, 0.3]$. This is the case where bandwidth is relatively small. According to Fig. 11, the average MCAV is stably around 0.4. On the contrary, we also simulate the similar case only for large bandwidth saturation: $HB = [0.95, 0.8, 0.9, 0.9]$. According to Fig. 12, the average MCAV is stably around 1, which shows this nonself antigen is definitely a malicious one.

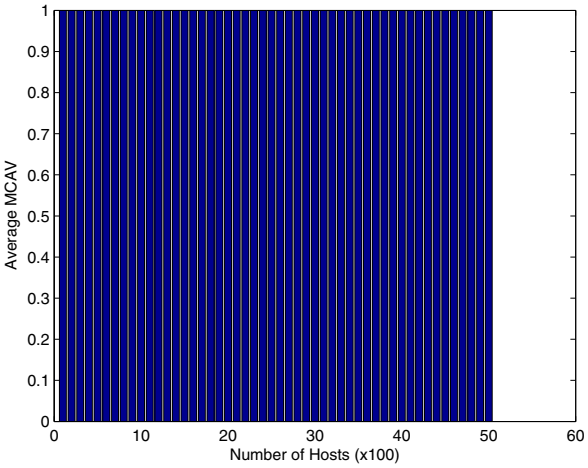

Fig. 12. Average MCAV for an Antigen causing host behavior HB=$[0.95, 0.8, 0.9, 0.9]$

4.3 Simulation of TC Agents' Coordination

If a TC agent determines its corresponding host is under malicious attack, it will inform other TC agent "nearby". Such communication is direct and does not pass through proxy server or SOC. The TC activation threshold T will be updated according to the following equation.

$$T(t) = T(t-1) + \alpha E(t-1)(E(t-1) - T(t-1)), t = 1, 2, \ldots, t_b \qquad (4)$$

where E is the excitation level of TC, and t_b is the attacking time by this antigen. Fig. 13 is a simulation result where attacks incurs around time 150.

Interaction between Two TC Agents. We consider two computer hosts with TC interactions. TC1 and TC2 are informing each other for confirmed attack from the corresponding antigen. According to Fig. 14, even two TC agents will lead to "more" complex threshold behaviors.

Interaction among Three TC Agents. We consider three computer hosts with TC agents interactions. TC1, TC2 and TC3 are informing each other for confirmed attack from the corresponding antigen. According to Fig. 15, three TC agents will lead to "more" complex threshold behaviors.

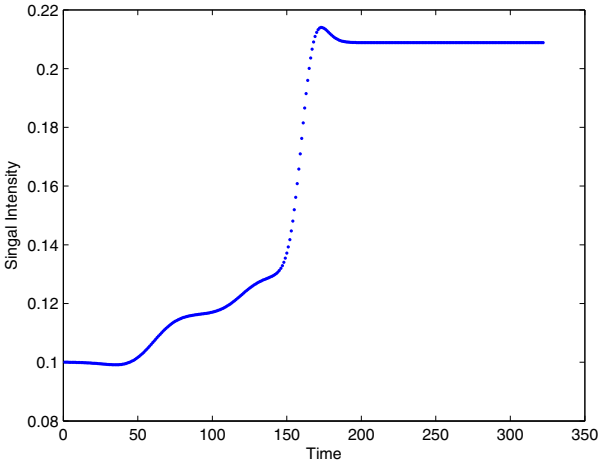

Fig. 13. Activation Threshold of a TC agent within the attacking cycle

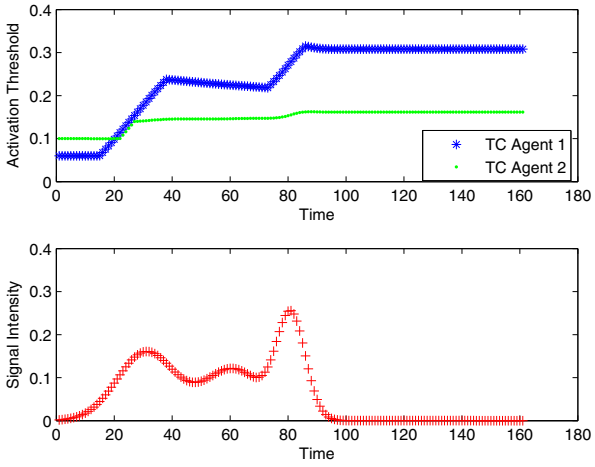

Fig. 14. Activation Thresholds of Two TC agents within the attacking cycle

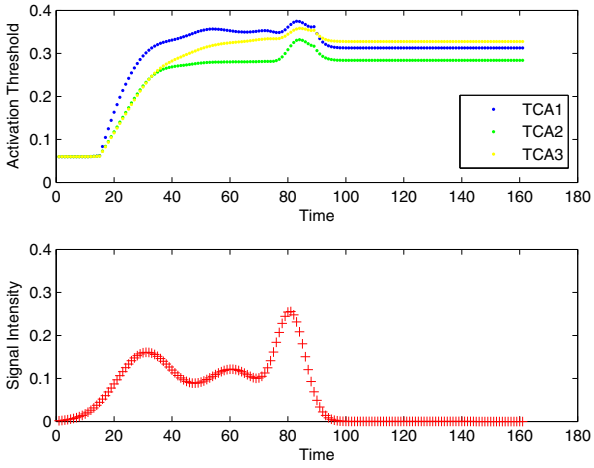

Fig. 15. Activation Thresholds of Three TC agents within the attacking cycle

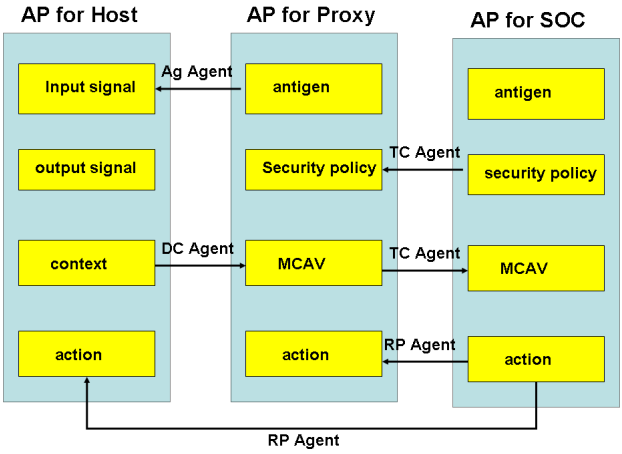

Fig. 16. Agent Interaction for ABIDS

5 Analysis of ABIDS

5.1 Agent Interactions

One advantage of this ABIDS is the feasibility of being adapted into large-scale computer networks such as cloud computing environment, as TC agents will collect information from other computer hosts to update threat profile. Fig. 16 illustrates the agent interactions between SOC, computer hosts and proxy server

on their agent platforms. This can be regarded as a security mechanism based on cloud computing.

5.2 Advantages of Adopting AIS to IDS

Table 7 is the comparisons for IDS adopting AIS mechanism such as danger theory. We realize that AIS improves the adaptability of IDS such as self tolerance and learning mechanism. In particular, AIS mechanism can contribute to the updates automation of Self-nonself activation threshold, which is a major concern for current network security.

Table 7. Comparisons between IDS with and without AIS

System	IDS /w AIS	IDS /wo AIS
Diversity Generation	NO NEED	NEEDED
Self Tolerance	YES	NO
Learning Mechanism	GOOD	MEDIUM
Automated updates	**YES**	**NO**
SNS Act.Threshold	**YES**	**NO**
Self Maintenance	YES	YES
Memory of Nonself	YES	YES

6 Conclusions

We propose an agent-based IDS. The intelligence behind such system is based on the danger theory of human immune systems. In particular computations of danger values with dynamic activation thresholds will reduce the false positive rate of danger signals issued by computer hosts. Three agents, namely, Ag agent, DC agent and TC agents are coordinated to exchange information of intrusion detections. The evaluations of three factors S, C, and T are pragmatic issues.

This research is concentrated on the adaptation of DCA to agent-based intrusion detection mechanism in first place. Our experiments are concerned with the feasibility of such technology. According to [36], DCA can effectively reduce both the false positive rate and the false negative rate while applied to detection of port scan-based attacks. Further consideration of importing real network packets for discussing FPR and FNR will be our future goal.

References

1. The Ten Most Critical Web Application Security Vulnerabilities, 2007 update, 2002-2007 OWASP Foundation
2. Forrest, S., Hofmeyr, S.A., Somayaji, A., Longstaff, T.A.: A sense of self for Unix processes. In: Proceedinges of the 1996 IEEE Symposium on Research in Security and Privacy, pp. 120–128. IEEE Computer Society Press (1996)

3. Gu, F., Aickelin, U., Greensmith, J.: An Agent-based classification model. In: 9th European Agent Systems Summer School (EASSS 2007), Durham, UK (2007), http://arxiv.org/ftp/arxiv/papers/0910/0910.2874.pdf
4. Bauer, A., Beauchemin, C., Perelson, A.: Agent-based modeling of host-pathogen systems: the successes and challenges. Information Sciences 179, 1379–1389 (2009)
5. Harmer, P., Williams, P., Gunsch, G., Lamont, G.: An Artificial Immune System Architecture for Computer Security Applications. IEEE Transactions on Evolutionary Computations 65(3), 252–280 (2002)
6. Burnet, F.: The clonal selection theory of acquired immunity. Cambridge Univerisity Press (1995)
7. Matzinger, P.: Tolarance, Danger and the Extended Family. Annual Review in Immunology 12, 991–1045 (1994)
8. King, R., Russ, S., Lambert, A., Reese, D.: An Artificial Immune System Model for Intelligent Agents. Future Generation Computer Systems 17(4), 335–343 (2001)
9. Aickelin, U., Bentley, P., Cayzer, S., Kim, J., McLeod, J.: Danger Theory: The Link between AIS and IDS? In: Timmis, J., Bentley, P.J., Hart, E. (eds.) ICARIS 2003. LNCS, vol. 2787, pp. 147–155. Springer, Heidelberg (2003)
10. Kephart, J.: A Biologically Inspired Immune System for Computers. In: Artificial Life IV Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems, pp. 130–139. MIT Press
11. Kim, J., Bentley, P., Aickelin, U., Greensmith, J., Tedesco, G., Twycross, J.: Immune system approaches to intrusion detection: a review. Natural Computing 6(4), 413–466 (2007)
12. Liu, S., Li, T., Wang, D., Zhao, K., Gong, X., Hu, X., Xu, C., Liang, G.: Immune Multi-agent Active Defense Model for Network Intrusion. In: Wang, T.-D., Li, X., Chen, S.-H., Wang, X., Abbass, H.A., Iba, H., Chen, G.-L., Yao, X. (eds.) SEAL 2006. LNCS, vol. 4247, pp. 104–111. Springer, Heidelberg (2006)
13. Dasgupta, D.: Immunity-based intrusion detection systems: a general framework. In: Proceeding of the 22nd National Information Systems Security Conference, NISSC (October 1999)
14. Burgess, M.: Computer Immunology. In: Proceedings of the Twelfth Systems Administration Conference (LISA 1998), Boston, Mass, December 6-11 (1998)
15. Aickelin, U., Cayzer, S.: The danger theory and its application to ais. In: Proceedings of the First International Conference on Artificial Immune System (ICARIS 2002), pp. 141–148 (2002)
16. Greensmith, J., Aickelin, J., Cayzer, S.: Detecting danger, The Dendritic Cell Algorithm. Robust Intelligent Systems 12, 89–112 (2008)
17. Kim, J., Wilson, W.O., Aickelin, U., McLeod, J.: Cooperative Automated Worm Response and Detection ImmuNe ALgorithm(CARDINAL) Inspired by T-Cell Immunity and Tolerance. In: Jacob, C., Pilat, M.L., Bentley, P.J., Timmis, J.I. (eds.) ICARIS 2005. LNCS, vol. 3627, pp. 168–181. Springer, Heidelberg (2005)
18. Sarafijanović, S., Boudec, J.Y.: An Artificial Immune SYstem for Misbehavior Detection in Mobile Ad-Hoc Networks with Virtual Thymus, Clustering, Danger Signal and Memory Detectors. International Journal of Unconventional Computing 1, 221–254 (2005)
19. Ishida, Y.: Immunity-based Systems. A Design Perspective. Springer (2004)
20. Weiss, G.: Multiagent Systems, A Modern Approach to Distributed Artificial Intelligence. MIT Press (1999)
21. Fu, H., Yuan, X., Wang, N.: Multi-agents artificial immune system (MAAIS) inspired by danger theory for anomaly detection. In: International Conference on Computational Intelligence and Security Workshops, pp. 570–573 (2007)

22. Kim, J., Bentley, P.: The Artificial Immune Model for network intrusion detection. In: Proceeding of European Congress on Intelligent Techniques and Soft Computing (EUFIT 1999), Aachen, Germany (September 1999)
23. Scarfone, K., Mell, P.: Guide to Intrusion Detection and Prevention Systems (IDPS). Computer Security Resource Center (National Institute of Standards and Technology) (800-94) (February 2007)
24. Kim, J.: Integrating artificial immune algorithms for intrusion detection, Ph.D thesis, University College London (2002)
25. Yeom, K.-W., Park, J.-H.: An artificial immune system model for multi agents based resource discovery in distributed environments. In: Proceedings of the First International Conference on Innovative Computing, Information and Control, pp. 234–239 (2006)
26. Aickelin, U., Bentley, P., Cayzer, S., Kim, J., McLeod, J.: Danger Theory: The Link between AIS and IDS? In: Timmis, J., Bentley, P.J., Hart, E. (eds.) ICARIS 2003. LNCS, vol. 2787, pp. 147–155. Springer, Heidelberg (2003)
27. Zhang, J., Liang, Y.: Integrating Innate and Adaptive Immunity for Worm Detection. In: Second International Workshop on Knowledge Discovery and Data Mining (WKDD 2009), pp. 693–696 (2009)
28. Kim, J., Bentley, P., Aickelin, U., Greensmith, J., Tedesco, G., Twycross, J.: Immune system approaches to intrusion detection-A review. Nature Computing 6, 413–466 (2007)
29. Hofmeyr, S., Forrest, S.: Immunity by Design. In: Proc. of the Genetic and Evolutionary Computation Conference (GECCO), pp. 1289–1296 (1999)
30. Yeom, K.-W.: Immune-inspired Algorithm for Anomaly Detection. In: Nedjah, N., Abraham, A., de Macedo Mourelle, L. (eds.) Computational Intelligence in Information Assurance and Security. SCI, vol. 57, pp. 129–154. Springer, Heidelberg (2007)
31. Boukerche, A., Machado, R., Juca, K., Sobral, J., Motarem, M.: An Agent based and Biological Inspired Real-time Intrusion Detection and Security Model for Computer Network Operations. Computer Communications 20, 2649–2660 (2007)
32. Forrest, S., Beauchemin, C.: Computer immunology. Immunological Reviews 216(1), 176–197 (2007)
33. Spafford, E., Zamboni, D.: Intrusion detection using autonomous agents. Computer Networks 34(4), 547–570 (2000)
34. Greensmith, J., Aickelin, U., Tedesco, G.: Information fusion for anomaly detection with the Dendritic Cell Algorithm. Information Fusion 11, 21–34 (2010)
35. Greensmith, J., Feyereisl, J., Aickelin, U.: The DCA: SOMe Comparison. Evolutionary Intelligence 1(2), 85–112 (2008)
36. Greensmith, J., Aickelin, U., Cayzer, S.: Introducing Dendritic Cells as a Novel Immune-Inspired Algorithm for Anomaly Detection. In: Jacob, C., Pilat, M.L., Bentley, P.J., Timmis, J.I. (eds.) ICARIS 2005. LNCS, vol. 3627, pp. 153–167. Springer, Heidelberg (2005)