An Immune-inspired Adaptive Automated Intrusion Response System Model

Ling-xi Peng, Dong-qing Xie*, Ying Gao, Wen-bin Chen, Fu-fang Li, Wu Wen
Department of Computer and Education software, Guangzhou University,
Guangzhou, 510006, China

Jue Wu*
University of Kentucky, 773 Anderson Tower,
Lexington, KY 40506-0046,USA

Received 12 October 2011
Accepted 15 June 2012

Abstract

An immune-inspired adaptive automated intrusion response system model, named as *MAIM*, is proposed. The descriptions of self, non-self, immunocyte, memory detector, mature detector and immature detector of the network transactions, and the realtime network danger evaluation equations are given. Then, the automated response polices are adaptively performed or adjusted according to the realtime network danger. Thus, *MAIM* not only accurately evaluates the network attacks, but also greatly reduces the response times and response costs.

Keywords: artificial immune; network security; automated intrusion response system; intelligent system

1. Introduction

The role of intrusion detection system (IDS) [1] is to monitor network activity and find attempts of various attacks, aggressive behaviors or attack results. Intrusion response technology is the new generation of technology based on active defense idea, which has very prominent significance on the protection of network security. It is a promising direction of network security technology[2].

Existing research on the automated intrusion response are relatively few. In these studies, Fisch was the earliest to classify the intrusion responses, who presented the classification method according to the detecting time of intrusions and the response of targets. The disadvantage of this method is that the classification is not considerate[3]. Curtis made a more comprehensive classification of intrusion response approach, which takes into account the progress of the attack, attack type, and costs as the basis of response decisions[4]. Wenke L. proposed a response decision method based on response costs, which is a cost-sensitive model[5]. However, the method of quantitative response is too rough. Wu et al. designed and implemented an automated response system called as ADEPTS, which takes a directed graph to model the risk transmission of the system, and then carries out an appropriate response according to specific attack effect[6]. Mu et al. put forward a scheme based on hierarchical task network intrusion response decision-making model, including the response decision-making process and decision-making response time[7]. Cuppens et al. put forward a method based on context and ontology, which defines a set of rules to implement this kind of policy-

*Corresponding author: flyingday@139.com

Published by Atlantis Press

based intrusion responses[8]. Zhang et al. proposed an automatic rollback intrusion response system, which takes responses according to IDS' detected attacks and rolls back until the end of the attack session[9]. Tan et al. proposed an immune-inspired mechanism that allows some level of automation of intrusions response[10]. However, it's just an idea and the implement details including the response polices are not given. Wu et al. presented the method of establishment of the attack groups through the relationship diagram, which judges the attack sequence and reconstructs the collaborative intrusion attacks process, and then responds to attacks so as to minimize response costs[11].

Summary, there are two main problems among the current automated intrusion response researches. The first is the high alarm rate in intrusion detection systems, which leads that automatic intrusion response systems are difficult to transact the dangerous invasions and attacks. Thus, the intrusion response polices are affected. In addition, most of the intrusion response models or methods lack rigorous, comprehensive, and quantitative mathematical model to dynamically adjust the response policies. These two problems limit the application of current automated intrusion response researches.

The problems of computer network security and the problems faced by biological immune system (BIS) show pretty direct and fabulous analogy, since both of them have to maintain stability in a changing environment. The network security techniques based on artificial immune system (AIS) have the features of diversity, self-adaptation and robustness. Thus, they are considered a very promising research direction in network security[12-13].

In order to make the network information system accurately transact the dangerous intrusion attack, and reduce the response costs and response times of intrusion response systems, in this paper, the descriptions of self, non-self, immunocyte, memory detector, mature detector and immature detector of the network transactions are first given, and then an immune-inspired adaptive automated intrusion response system model, named as *MAIM*, is proposed. *MAIM* accurately evaluates the realtime network danger of the network attacks and intrusions, and then adaptively adjusts intrusion response policies according to realtime network danger. In this way, *MAIM* solves the problem that the current automated response system models could not accurately evaluate the dangerous intrusions and attacks. Both the theoretical analysis and experimental results show that *MAIM* provides an effective and intelligent method for automatic intrusion response system.

2. Proposed theoretical models

In our model, the antigen is defined as (Ag, $Ag \subset D$, $D = \{0,1\}^l$) binary strings with fixed l bits, which are extracted from the Internet Protocol (IP) packets transferred in the network. Nonself patterns (*Nonself*) represent IP packets from computer network attacks, while self patterns (*Self*) are normal network service transactions, such that $Self \cup Nonself = Ag$ and $Self \cap Nonself = \Phi$.

We use quadruple <d, age, $count$, s> to describe the set of immune detectors, B, where d represents the detectors' genes, $d \in D$, age represents the detectors' age, $age \in N$, $count$ represents the detectors' match number with the antigens, $count \in N$, s represents the detector's realtime danger, $s \in R$, N is natural number set, R is the set of real numbers. The immune detectors are composed of memory detectors and mature detectors, which is $B=M_b \cup T_b$. We define $T_b = \{x \mid x \in B, x.count < \beta)\}$ and $M_b = \{x \mid x \in B, x.count \geq \beta)\}$ where $\beta \in N$, thus $M_b \cap T_b = \Phi$. The immune detectors do not match any self pattern. The match function in domain D, $Match=\{<x, y> \mid x, y \in D, f_{match}(x, y)=1\}$, where $f_{match}(x, y)$ is the match value between x and y, which can be Hamming distance, Euclidean distance, r-contiguous bits matching function, etc. Let couple <d, age> represent the set of immature immune detectors, I_b. I_b can be randomly generated or from the gene bank. After α number of tolerance cycle, the immature immune detectors which identify the self antigen will be removed, and the left will be evolve into mature immune cells, T_b, and $\alpha \in N$.

MAIM first detects the network intrusions or attack, evaluates the realtime danger of each network attack, and then automatically adjusts intrusion response policies according to the realtime network danger.

2.1. *Immune-inspired realtime network danger evaluation*

At the training stage, for each input set of Ag, Ag is divided into δ (δ is a positive integer) eras. For each era of antigens, a number of antigens will be selected, which will constitute the set of *sAg*. *sAg* will be classified as *Self* and *Nonself* by the detection of *B* detectors set. The whole process is composed of three phases. The first stage is from time 0 to the tolerance termination time α. The initial self set *Self*(0) and immature detectors set $I_b(0)$ are defined in this stage. $I_b(0)$ will grow up into the set of mature detectors. The second stage is from the time α+1 to the birth of memory detectors, which is the self-learning stage. Through clone selection, mature detectors will evolve into memory detectors which are used to detect non-self antigens. Antigens which have been detected as self antigen will be sent to the immature detectors for their tolerance. The last stage is from the birth of memory detectors to the ending of system. All parts of the immune system come into being, and they are used to detect in real network environment: memory detectors detect the antigens; mature detectors detect left ones; immature detectors experience tolerance by the remaining antigens. The whole process is shown in Fig. 1.

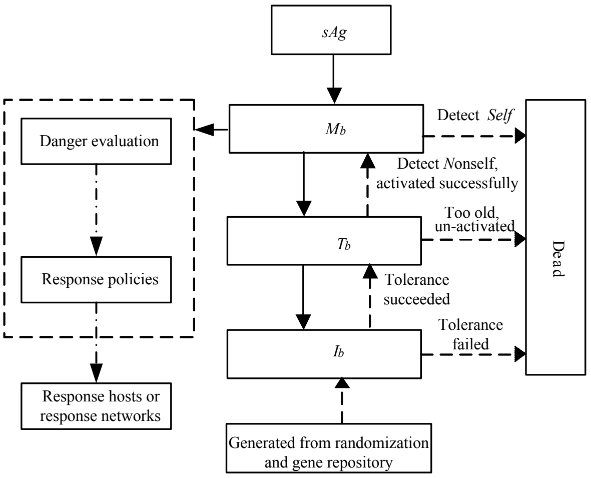
sAg
Danger evaluation
Response policies
Response hosts or response networks
Mb
Tb
Ib
Detect Self
Detect Nonself, activated successfully
Too old, un-activated
Tolerance succeeded
Tolerance failed
Dead
Generated from randomization and gene repository

Fig. 1. The workflow of *MAIM*. The dashed line represents the moving direction of detectors, the solid line represents the moving direction of *Ag*, and the dashdotted line is the flow of response polices.

After the initial training process is completed, the memory detectors evaluate the realtime network danger for the detected intrusion or attacks. Each memory detector corresponds to one kind of attack or intrusion. From time *t*-1 to *t*, for each memory detector, *x*, if it matches the detecting antigen *ag* such as Eq. (1), which indicates that the memory detector has detected intrusion or attack. The realtime danger of memory detectors will be added as Eq. (2). If more antigens have been detected, the realtime danger of memory detectors will be accumulatively calculated, which indicate that the threat of such attacks continues to increase, where $\eta_1(\eta_1>0)$ is the initial realtime danger and $\eta_2(\eta_2>0)$ represents the encouragement factor.

$$f_{\mathrm{match}}<x.d,\ ag.d>=1 \tag{1}$$

$$x.s(t)=\eta_1+\eta_2\bullet x.s(t-1) \tag{2}$$

Conversely, if the memory detector fails to detect the intrusion or attack antigens, the realtime danger will decay as Eq. (3).

$$x.s(t)=x.s(t\text{-}1)\bullet e^{-1} \tag{3}$$

If the memory detector has not detected the intrusion or attack antigens in λ ($\lambda \in N$) number of intervals, then $x.s(t)=x_0.\mathrm{s}(t_0)*\mathrm{e}^{-\lambda}$. The larger value of the λ, the smaller value of realtime network danger, which indicates that the realtime network danger is decaying. When λ->∞, $x.s(t)$->0 which indicates that the menace of the attack has cleaned and the attack alarms will be relieved.

$$r_{\mathrm{k,i}}(t)=\frac{2}{1+e^{-\mu_i\bullet\sum_{x\in A_{\mathrm{k,i}}(t)}x.s}}\text{-}1 \tag{4}$$

Let $r_{\mathrm{k}}(t)$ ($0\le r_{\mathrm{k}}(t)\le 1$, $1\le k\le K$) represent the realtime danger of the *k-th* host at time *t*: if $r_{\mathrm{k}}(t)=1$, which indicates that the system is in extremely danger; if $r_{\mathrm{k}}(t)=0$, which indicates that the system is safe. The larger value of $r_{\mathrm{k}}(t)$, the larger realtime danger that the system is confronted with. Taking into account the rightweight of the host assets and the different types of attacks, we let μ_{i} represent the realtime danger rightweight of the *i*-th type of attack. Let ω_{k} represent the asset rightweight of the host *k*. For the *k-th* host, the realtime danger of *i*-th attack at time *t* is defined by Eq. (4).

For the *k*-th host, the realtime network danger at time *t*, is defined by Eq. (5).

$$r_{\mathrm{k}}(t)=\frac{2}{1+e^{-\sum_{i=1}^{I}\mu_{\mathrm{i}}\bullet\sum_{x\in A_{\mathrm{k,i}}(\mathrm{t})}x.s(t)}}-1 \qquad (5)$$

For the entire network, the realtime danger of the i-th ($1\le i\le I$) attack at time t, $R_{\mathrm{i}}(0\le R_i(t)\le 1)$, can be calculated by following Eq. (6), where ω_{k} and ξ_{n} represent the asset rightweight of *k-th* host and *n-th* sub-network, respectively. The sub-network can also conclude the next level of sub-network, etc.

$$R_{\mathrm{i}}(t)=\frac{2}{1+e^{-(\sum_{k=1}^{K}\varpi_{\mathrm{k}}\bullet r_{\mathrm{k,i}}+\sum_{n=1}^{N}\xi_{\mathrm{n}}\bullet \boldsymbol{R}_{\mathrm{n,i}}(\boldsymbol{t}))}}-1 \qquad (6)$$

Similarly, the realtime network danger of network at time t, $R(t)$ ($0\le R(t)\le 1$), can be calculated by Eq. (7).

$$R(t)=\frac{2}{1+e^{-(\sum_{k=1}^{K}\varpi_{\mathrm{k}}\bullet r_{\mathrm{k}}(t)+\sum_{n=1}^{N}\xi_{\mathrm{n}}\bullet R_{\mathrm{n}}(t))}}-1 \qquad (7)$$

2.2. *Self-adaptive automated intrusion response and rollback*

The automated intrusion responses depend on the realtime network danger. When the realtime network danger is less than a given threshold value, the detected irrelevant alarms and false alarms will be ignored. Through realtime network danger evaluation, the detection information is associated with each other. Eq. (8) describes the intrusion responses of the host, which is mainly from two aspects: when the host's realtime network danger, r_{k}, is greater than θ_{k} ($0<\theta_{\mathrm{k}}<1$); or when the i-th attack's realtime network danger, $r_{\mathrm{k,i}}(t)$, is greater than $\delta_{\mathrm{k,i}}$ ($0<\delta_{\mathrm{k,i}}<1$).

$$Response(t)=\begin{cases}1 \ \ iff\ r_{\mathrm{k}}(t)>\theta_{\mathrm{k}}\vee r_{\mathrm{k,i}}(t)>\delta_{\mathrm{k,i}}\\ 0 \ \ otherwise\end{cases} \qquad (8)$$

These two aspects are corresponding to the two specific meaning: the realtime network danger of the host, $r_{\mathrm{k}}(t)$, is larger than a given threshold indicating that all attacks has affected the security of the host; the realtime network danger of i-th kind attack, $r_{\mathrm{k,i}}(t)$, is larger than a given threshold indicating that the host's detected variants of similar attacks are already dangerous.

For the network, the intrusion responses are also mainly from two aspects such as Eq. (9): when the network's realtime network danger, $R(t)$, is greater than θ' ($0<\theta'<1$); or when the i-th attack's realtime network danger, $R_{\mathrm{i}}(t)$, is greater than δ_{i}' ($0<\delta_{\mathrm{i}}'<1$).

$$Response'(t)=\begin{cases}1 \ \ iff\ R(t)>\theta'\vee R_{\mathrm{i}}(t)>\delta_{\mathrm{i}}'\\ 0 \ \ otherwise\end{cases} \qquad (9)$$

Similarly, these two aspects are corresponding to the two specific meaning: the network's realtime network danger, $R(t)$, is larger than a given threshold indicating that all the network attacks has affected the security of the network; the realtime danger of i-th kind attack, $R_{\mathrm{i}}(t)$, is larger than a given threshold indicating that the network's detected variants of similar attacks are already dangerous.

In our proposed model, for each host, when the realtime network danger of host is larger than a give threshold, the host will take intrusion responses policies. Similarly, for the network, when the realtime network danger of network is larger than a give threshold, the network will take intrusion response policies. Since the model implements the automatic response policies according to the holistic realtime danger of the network and host instead of single attacks, *MAIM* can well solve the problem that Wenke L.'s cost-sensitive model [4] is difficult to deal with collaborative attacks.

Curtis summarized and classified all response policies into host-based and network-based response policies. He listed 11 network-based policies[3]. Combining with other 6 response policies from other literates, we propose 6 response policies. The comprehensive automated intrusion response policies and rollback policies are listed as Table 1 shows.

As far as the listing response policies are concerned, the policies with number of 1-4, 12-13, and 17 are moderate; the policies with number of 8-11, 14, 16, and 20 are relatively stringent, and the rest of response policies are between them. The policies with number of 1-4, 12-13, and 21-22 are passive policies, and others are active ones. The policies with number of 8-11 and 14-15 are usually limited by law and other factors; the policies with number of 5-7, 14-16, and 20 can effectively block the attackers.

MAIM carries out automatically responses policies according to realtime network danger of the network and host. The higher of the danger, the more stringent response policies it will take, and vice versa. In the experiments of this paper, the realtime network dangers and their corresponding response policies are listed in Table 2. For all the detected network attacks, *MAIM* will record the secure events. In addition, when the realtime danger is larger than 0.1, *MAIM* will alarm. In practices, the specific response policies can be taken according to the network security.

Table 1. The network-based intrusion response policies

Number	Response policies	Response operation	Rollback operation
1[4]	record secure events	*Log*	*Φ*
2[4]	generate alarm information	*Alert*	*Φ*
3[4]	record additory secure events	*Enable*	*Disable*
4[4]	activate the additive intrusion detection tools	*Add*	*Φ*
5[4]	isolate the attacker's IP	*Lock*	*Unlock*
6[4]	forbidden the service of attacked objects	*Stop*	*Start*
7[4]	isolate the attacked objects	*Shutdown*	*Restart*
8[4]	warn the attackers	*Warn*	*Φ*
9[4]	trace the attackers	*Track*	*Φ*
10[4]	cut off the dangerous connections	*Reset*	*Φ*
11[4]	attack the attackers	*Attack*	*Φ*
12[14]	Honeypot	*Honeypot*	*Φ*
13[15]	Honeynet	*Honeynet*	*Φ*
14[14]	dynamically modify the strategy of firewall	*Link*	*UnLink*
15[14]	absorption and transfer of attack source	*Absorb*	*Release*
16[14]	blacklist	*Blacklist*	*UnBlacklist*
17	forensics	*Forensics*	*Φ*
18	service switch(service excursion)	*Migrate*	*MigrateBack*
19	hot backup	*HotBackup*	*StopHotBackup*
20	cold backup	*ColdBackup*	*StopColdBackup*
21	whole recover	*AllRecovery*	*Φ*
22	differential recover	*DiffRecovery*	*Φ*

Table 2. The realtime network danger and corresponding response policies

Realtime network danger	Number of Response policy
0-0.1	1
0.1-0.2	1,2
0.2-0.3	1,2,3
0.3-0.4	1,2,4,12,13,17
0.4-0.5	1,2,5,16,17,18
0.5-0.6	1,2,6,7,14,17,19
0.6-0.7	1,2,8,15,17,20
0.7-0.9	1,2,9,10,17,22
0.9-1	1,2,11,17,21

MAIM implements the response policies according to the realtime network danger and sends automatic response policies to host or network in the form of message. The response messages are 5-tuple, such as Eq. (10). For the above 22 response policies, the corresponding operations and the corresponding rollback actions are shown in Table 1 where "*Φ*" represents the null operation.

$$\begin{aligned} Message := &< Sender >< Receiver > \\ &< SendTime >< ValidTim >< Action > \end{aligned} \tag{10}$$

After *MAIM* carries out the automatic intrusion responses, if the realtime danger of network and host are becoming more and more larger and have a upward trend, *MAIM* will take more stringent policies in order that the information system gets into more dangerous state, which will assure the security running of the information system.

For the host, Eq. (11) describes the next response of the host, where *Response*(t+Δt) shows that at time t+Δt, the host will respond again. Since the host is up against more serious attack, the system will take more stringent response policies.

$$Response(t+\Delta t)=\begin{cases}1 \;\; iff \; r_{\mathrm{k}}(t+\Delta t)>r_{\mathrm{k}}(t) \\ \quad\quad \vee\, r_{\mathrm{k,i}}(t+\Delta t)>r_{\mathrm{k,i}}(t) \\ 0 \;\; otherwise\end{cases} \quad (11)$$

Similarly, for the network, Eq. (12) describes the next response of the network, where *Response*'(t+Δt) shows that at time t+Δt, the network will respond again. Since the network is up against more serious attacks, the system will take more stringent response policies.

$$Response'(t+\Delta t)=\begin{cases}1 \;\; iff \; R(t+\Delta t)>R(t)\vee \\ \quad\quad R_i(t+\Delta t)>R_i(t) \\ 0 \;\; otherwise\end{cases} \quad (12)$$

When the realtime network danger is lower than a given threshold, *MAIM* will take the automatic response rollback policies[9] in case that the normal network services has been affected.

$$Rollback(t{+}\Delta t')=\begin{cases}1 \;\; iff \; r_{\mathrm{k}}(t)\le\mu_{\mathrm{k}}\wedge r_{\mathrm{k,i}}(t)\le\omega_{\mathrm{k,i}} \\ 0 \;\; otherwise\end{cases} \quad (13)$$

For the host, Eq. (13) describes how the automatic rollback responses of the host are taken, where *Rollback*(t+$\Delta t'$) represents that the host will take rollback response policies at time t+Δt', μ_{k} is the host's realtime danger threshold of rollback, and $\omega_{\mathrm{k,i}}$ is the i-th attack's realtime danger threshold of rollback.

$$Rollback'(t{+}\Delta t')=\begin{cases}1 \;\; iff \; R(t)\le\mu'\wedge R_{\mathrm{i}}(t)\le\omega_{\mathrm{i}}' \\ 0 \;\; otherwise\end{cases} \quad (14)$$

For the network, Eq. (14) describes how the automatic rollback responses of the network are taken, where *Rollback*'(t+$\Delta t'$) represents the network will take rollback response policies at time t+Δt', μ' is the network's realtime network danger threshold of rollback, and ω_{i}' is the i-th attack's realtime danger threshold of rollback.

The automatic response policies will be sent to the executing host or network in form of message as Eq. (10) shows.

Since *MAIM* takes response policies according to the realtime network danger, it has an edge over fast response and good resistance to denial-of-service service; In addition, *MAIM* can be easily linked with the firewall, so its cooperativity is strong. Finally, when realtime network danger decreases, *MAIM* automatically takes rollback policies and so the information resource can be utilized effectively.

3. Simulations

The experiments were carried out in the Laboratory of Computer Network and Information Security at Guangzhou University. The environment is 100M local area network, which was linked to the Internet through a Class C IP address, 202.192.87.*. The Operation System of server is Red Hat Linux 9.0, and the server provided services of WWW, FTP, and Email. The antigen was defined as a fixed length binary string (l=128) composed of the source/destination IP address (64 bits), port number (16 bits), protocol flags (16 bits), and packet contents (32 bits). Hamming matching rule was used to compute the affinity between antigens and immune cells (Distance=80). Since the intrusion detection system-inspired by artificial immune has many desirable attributes including self-learning, adaptability, diversity, which can well detect unknown attacks and approved in many researches[13, 15-17], the emphases of the experiments are intrusion responses.

3.1. *Automated intrusion response experiments*

To verify the effectiveness of intrusion response model, we conducted a simulated *smurf* attack, the response thresholds of the realtime network danger, θ' and δ_{i}', are all set as 0.4. Fig. 2 and Fig. 3 show the comparison of network flow and *CPU* utilization under three scenes including normal cases, no response, and response with *MAIM*.

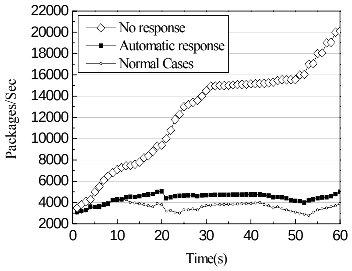

Fig. 2. The network flow comparison of three cases

From Fig. 2, it can be seen that from time 0s to 12s, after the attack, network traffic continued to increase without the intrusion response. In Fig. 3, the corresponding CPU utilization also increased. From 12s to 20s, the network traffic continued to go high as the attacks increased. With the intrusion response, the network traffic and CPU utilization also increased. However, the trend of curves is relatively stable.

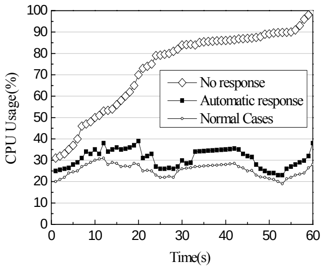

Fig. 3. The CPU utilization comparison of three cases

Compared with Fig. 2 and Fig. 3, network traffic with intrusion responses was much smaller than that without intrusion responses, and so does the CPU utilization. However, they are slightly larger than under normal scenario, which is because the intrusion responses increase the network traffic and the response transactions also slightly increase CPU utilization.

Experimental results show that *MAIM* effectively reduces the network traffic and CPU utilization through the automated intrusion responses.

3.2. *The comparative experiments*

In order to prove that *MAIM* effectively reduces the response costs and times, we compare *MAIM* with RARS[9]. RARS has well reduced the response costs compared with automatic intrusion response model.

Fig. 4 illustrates the realtime network danger curve, the attack intensity curve, and the points where the responses were sent out. From 8:30 to 9:00, the host suffered the portscan attacks. The attacks are to scan the open ports of the host. The rightweight of this attack is very low, which is just 0.0001. The *R*(*t*) was only 0.098. However, the realtime network danger of the portscan attack went to 0.825, which was over 0.4, so *MAIM* responded at point *A*. From 11 to 11:30, the host was attacked by sshtrojan attack. The attacker tried to trick the system administrator into installing a trojan version of the SSH program, which is a dangerous attack. This version of trojan allows the attacker to login into the victim. Although there was only two aggressive packets, *MAIM* detected this kind of attack, *R(t)* went to 0.488, and then the response was sent out at point *B*. From 16:30 to 17, the host was continuously attacked by apache2 and *smurf* attacks, which would exhaust the resources of the host machine. The *R*(*t*) went to 0.7, so the response was sent out at point *C*.

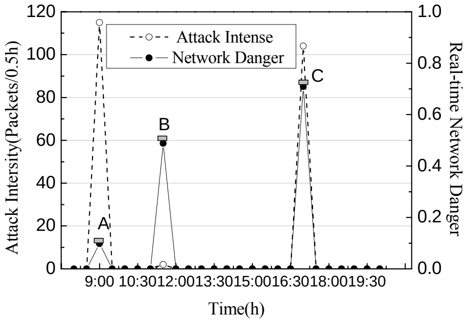

Fig. 4. The attack curve and the damage degree curve

The response costs of intrusion response system include rollback cost *RRCost*, operation cost *OCost*, response cost *RCost*, and damage cost *Dcost*. Let *TCost(R)* represent the accumulative costs of intrusion response system *R*, which is the summary of all costs. *E* is the set of intrusion event, thus,

$$TCost(R)=\sum_{e\in E}(RRCost(e)+OCost(e)+RCost(e)+DCost(e)) \tag{15}$$

Table 3 gives the comparison results of experiments. Compared *MAIM* with RARS, *MAIM* accurately calculated the realtime network danger, carried out the intrusion response policies according the realtime network danger, automatically adjusted the response strategy and implemented the rollback polices. In this way, *MAIM* reduces not only response times but also response costs, so which has good adaptability.

Table 3. The comparision of response number and costs

	n	*OCost*	*RRCost*	*RCost*	*DCost*	*TCost*
MAIM	3	100	100	40	85	325
RARS	221	100	22100	2175	3450	27825

4. Conclusion

In this paper, we apply immune-inspired computation intelligence to the research of the automated intrusion response system, and a new adaptive automated intrusion response system model, referred as *MAIM*, is proposed. *MAIM* can automatically adjust the response strategies according to the accurate realtime network danger, and only transact the dangerous network intrusions and attacks. In this way, *MAIM* not only ensures the performance of the system, but also greatly reduces the comprehensive costs of automated intrusion response systems, which greatly improves the self-adaptability of information systems.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant No. 61100150, Natural Science Foundation of Guangdong Province of China under Grant No. S2011040004528 and S2011040003843.

References

1. K. Ilgun, R. A Kemmerer, and P. A Porras, State transition analysis, A rule-based intrusion detection approach. *IEEE T. Softw. Eng.* **21**(3) (1995) 181–199.
2. X.T. Duan, C.F. Jia and C.B. Liu, Intrusion detection method based on hierarchical hidden Markov model and variable-length semantic pattern, *J. Commu.* **31**(3) (2010), 109-114.
3. E A. Fisch, *Intrusion Damage Control and Assessment: A Taxonomy and Implementation of Automated Responses to Intrusive Behavior* (Texas A&M University, College Station, TX., 1996).
4. A. C. Curtsi, A Methodology for Using Intelligent Agents to provide Automated Intrusion Response, in *Proc. of the IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop* (New York, USA, 2000), pp. 110-116.
5. L. Wenke, M. Matthew, J. S. Salvatore, F. Wei, and Z. Erez, Toward Cost-Sensitive Modeling for Intrusion Detection and Response, *J. Comp. Secur.* **10**(1-2) (2002) 5-22.
6. Y.S. Wu, B. Foo, Y. C. Mao, B. Saurabh and S. Eugene, Automated adaptive intrusion containment in systems of interacting services, *Comput. Networks*. **51**(5) (2007) 1334-1360.
7. C. P. Mu and Y. Li, An intrusion response decision-making model based on hierarchical task network planning, *Expert Syst. Appl.* **37**(3) (2010) 2465-2472.
8. N. Cuppens-Boulahia, F. Cuppens, F. Autrel, and H. Debar, An ontology-based approach to react to network attacks. *Int. J. Inf. Comp. Secur.* **3**(3) (2009) 280-305.
9. J. Zhang and J. Gong, Rollbackable Automated Intrusion Response System, *Acta Electr. Sin*. **32**(5) (2004) 769-771.
10. G.Z. Tan, Sam N. Njuki and R. M. Rimiru, Towards Automated Intrusion Response: A PAMP-Based Approach, *Int. J. Artif. Intell. Expert Syst.* **2**(2) (2011) 23-35.
11. Y. R. Wu and S. F. Liu, A Response Method for Cooperative Intrusions Based on the Attack Group Model, *Acta Electr. Sin.* **37**(11) (2009) 2416-2419.
12. T. Li, Dynamic Detection for Computer Virus based on Immune System, *Sci. China*, *Ser. F*, *Inf. Sci.* **51**(10) (2008) 1475-1486.
13. T. Li, An immunity based network security risk estimation, *Sci. China*, *Ser. F*, *Inf. Sci*. **48**(5) (2005) 798-816.
14. F. Zhang, *Policy Tree Based Proactive Defense Model for Network Securit*y (University of Electronic Science and Technology, Chengdu, 2004).
15. L. Wang, Z.G. Qin, Technology and Application of Production Honeynet, *Comput. Appl*. **24**(3) (2004) 43-45.
16. W. Chen, X.J. Liu, T. Li., Y.Q. Shi, X.F. Zheng and H. Zhao, A Negative Selection Algorithm Based on Hierarchical Clustering of Self Set and its Application in Anomaly Detection, *Int. J. Comput. Int. Sys*. **4**(4) (2011) 410–419.
17. X.C Zhao, G.L. Liu, H.Q. Liu, G.S. Zhao and S.Z. Niu, A new clonal selection immune algorithm with perturbation guiding search and non-uniform hypermutation, *Int. J. Comput. Int. Sys*. **3**(S1) (2010) 1–17.