

# Anti-spam model based on AIS in cloud computing environments

Jin Yang<sup>1, 2\*</sup>, Lingxi Peng<sup>3</sup>, Tang Liu<sup>4</sup>

<sup>1</sup>*School of Information Science & Technology, Southwest Jiaotong University, Chengdu 610031, China*

<sup>2</sup>*Department of Computer Science, LeShan Normal University, LeShan 614000, China*

<sup>3</sup>*Department of Computer and Education software/Guangzhou University, Guangzhou, China*

<sup>4</sup>*College of Fundamental Education/Sichuan Normal University, Chengdu, China*

*Received 1 March 2014, www.tsi.lv*

---

## Abstract

Cloud computing is becoming a hot research topic. However, there is little attention to cloud computing environment work for anti-spam issues. Spam has become a thorny issue facing with many countries. The overflow of spam not only great wastes the network resources, taking up the user's e-mail resources, reducing the network efficiency, affecting the normal use of the Internet, but also violates the user's individual rights. But the traditional spam solutions for anti-spam are mostly static methods, and the means of adaptive and real time analyses the mail are seldom considered. Inspired by the theory of artificial immune systems (AIS), this paper presents an anti-spam system in cloud computing environment. The concepts and formal definitions of immune cells are given, and the hierarchical and distributed management frameworks of the proposed model are built. The results of evaluation indicate that the proposed model has the features of real-time processing and is more efficient than client-server-based solutions, thus providing a promising solution for anti-spam system for heterogeneous cloud environments.

*Keywords:* cloud computing, artificial immune systems, anti-spam system

---

## 1 Introduction

What is cloud computing? Cloud computing is the delivery of computing as a service rather than a product, whereby shared resources, software, and information are provided to computers and other devices as a metered service over a network (typically the Internet) [1]. It has many advantages such as economy, complex calculations, agility, high scalability, high reliability, and easy maintenance. The concept of cloud computing was born in the 1960s from the ideas of American computer scientist J.C.R. Licklider and John McCarthy stated that computing will become a publicly available service in the future. In 1983, Sun Microsystems brought forward a singular vision that "the network is the computer" [2]. On August 9, 2006, the CEO Google, Eric Schmidt, firstly mentioned the concept of Cloud computing on SES San Jose 2006. On January 30, 2008, Google declared "Cloud Computing Research Plan" in Taiwan and will promote the advanced technology in Taiwan's colleges. February 1, 2008, IBM (NYSE: IBM) announced it will establish the first Cloud Computing Centre for software companies in China. On March 5, 2010, Novell and CSA released a supplier neutral plane, named as Trusted Cloud Initiative. May 22, 2009, China's first Cloud Computing Conference held in Beijing China

World Hotel. January 22, 2010, China cloud computing technology and industry alliance (CCCTIA) announced in Beijing. In the cloud computing trend, including computers, communications, Internet, the entire information technology industry is undergoing a comprehensive updating. Now software industry is facing momentous changing, which the software production organization evolves towards the service-oriented, agile, customized direction and the network terminal equipment begins to show the diversified and personalized features [3, 5].

Email spam, known as unsolicited bulk email, junk mail, or unsolicited commercial email, is the practice of sending unwanted email messages, frequently with commercial content, in large quantities to an indiscriminate set of recipients. Spam is becoming a serious problem since it causes huge losses to the organization, such as wasting the bandwidth, adding the user's time to deal with the insignificance mail, enhancing the mail server processing and causing the mail server to crush [6]. Cloud computing also faces the security issues such as the using of virtualization technology to hide viruses, Trojan horses, especially the spam and other malicious software problems. Anti-spam is the application of data investigation and analysis techniques currently mainly by means of blocking and filtering procedures [7]. However, the current techniques

---

\* *Corresponding author* e-mail: jinnyang@163.com

classifying a message as either spam or legitimate utilize the methods such as identifying keywords, phrases, sending address etc. Keeping a blacklist of addresses to be blocked, or an appointment list of addresses to be allowed are also used widely. Because spammers can create many false from e-mail addresses, it is difficult to maintain a black list that is always updated with the correct e-mails to block [8]. Message filtering methods is straightforward and does not require any modifications to existing e-mail protocols. But message filtering often rely on humans to create detectors based on the spam they've received. A dedicated spam sender can use the frequently publicly available information about such heuristics and their weightings to evade detection [9]. Neural networks also have been used for the detecting spam [10]. Using data mining method has been described as well. But the methods of adaptive capture the potential sensitive traffic and real time analyses the mail are seldom considered. Therefore, the traditional technology lack self-learning, self-adaptation and the ability of parallel distributed processing, calls for an effective and adaptive analysing system for anti-spam. Artificial immune systems (AIS) is a now receiving more attention and is realized as a new research hotspot of biologically inspired computational intelligence approach after the genetic algorithms, neural networks and evolutionary computation in the research of Intelligent Systems. Burnet proposed clone Selection Theory in 1958 [11]. Negative Selection Algorithm and the concept of computer immunity proposed by Forrest in 1994 [12]. It is known that the artificial immune system has lots of appealing features [13, 14] such as diversity, dynamic, parallel management, self-organization and self-adaptation that has been widely used in the fields such as [15, 16] data mining, network security, pattern recognition, learning and optimization etc. In this paper, we propose a new spam detection technique based on artificial immunity theory in cloud computing environments.

**2 Imperfection of the precise theory of value-based management**

A Cloud Computing environment has many distinctive characteristics such as large-scale, virtual, complex that are different from common network environments. The aim of this paper is to establish an immune-based model for dynamic spam detection in cloud computing environments. The principle of Anti-spam can be summarized as follows. The model is composed of three processes: Agent of Email Character distilling, Agent of Email Surveillance, and Agent of Training.

Agent of Email Character distilling use vector space model and present the received mail in discrete words. Agent of Training generates various immature detectors from gene library to distinguish Self and Non-self.

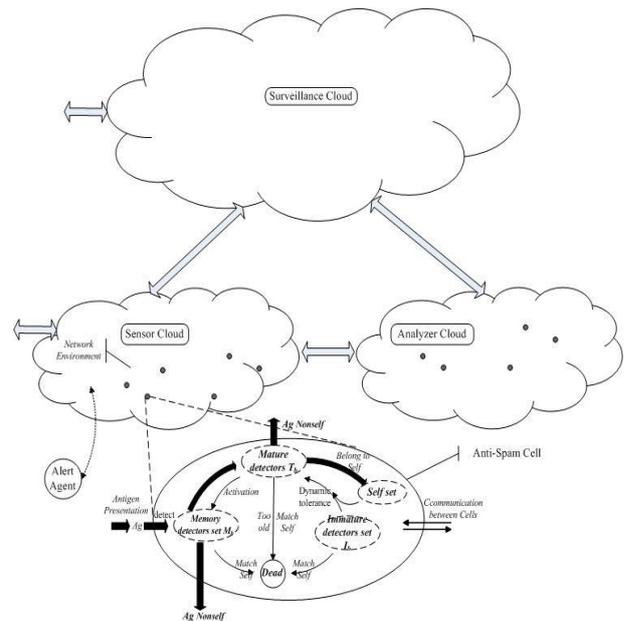


FIGURE 1 The Dynamic Anti-Spam Model

According to immune principle, some of these new immature detectors are false detectors and they will be removed by the negative selection Agent, which matches them to the training mails. If the match strength between an immature detector and one of the training mails is over the pre-defined threshold, this new immature detector is consider as a false detector. Agent of Email Surveillance matches the received mails to the mature detectors. If the match strength between a received mail and one of detectors, the mail will be consider as the spam. The detail training phases are as following.

**2.1 SINUSOIDAL PULSE WIDTH MODULATION**

An immune system can distinguish between self and non-self to detect potentially dangerous. These non-self elements include antibodies and viruses. In a spam immune system, we distinguish legitimate messages from spam. We consider the text of the email include the headers and the body as the antigen of a spam message. In the model, we define antigens (Ag) to be the features of email service and the email information, and given by:  $Ag = \{ag | ag \in D\}$ ,  $D = \{0,1\}^l$ . Antigens are binary strings extracted from the email information received in the network environment. The antigen consists of the gene libraries of emails include sender, sending organization, email service provider, receiving organization, recipient fields, etc.

The structure of an antibody is the same as that of an Antigen. For spam detection, the non-self set (Non-self) represents abnormal information from a malignant email service, while the self set (Self) is normal email service. Set  $Ag$  contains two subsets [17],  $Self \subseteq Ag$  and  $Nonself \subseteq Ag$  such that,

$$Self \cup Nonself = Ag, Self \cap Nonself = \Phi. \tag{1}$$

For the convenience using the fields of an antigen  $x$ , a subscript operator "." is used to extract a specified field of  $x$ , where  $x.fieldname$  = the value of filed fieldname  $x$ . In the model, all the detectors form a Set Detector called  $SD$ .

$$SD = \{ \langle d, age, count \rangle \mid d \in D, age \in N, count \in N \}, \quad (2)$$

where  $d$  is the antibody gene that is used to match an antigen, age is the age of detector  $d$ , count (*affinity*) is the number of detector matched by antibody  $d$ , and  $N$  is the set of nature numbers.  $SD$  contains two subsets: mature and memory, respectively, the set  $M$  and set  $T$ . A mature  $SD$  is a  $SD$  that is tolerant to self but is not activated by antigens. A memory  $SD$  evolves from a mature one that matches enough antigens in its lifecycle. Therefore,  $SD = M \cup T, M \cap T = \phi$ .

$$M = \{ x \mid x \in SD, \forall y \in Self, \langle x.d, y \rangle \notin Match \wedge x.count < \beta \}, \quad (3)$$

$$T = \{ x \mid x \in SD, \forall y \in Self, \langle x.d, y \rangle \notin Match \wedge x.count \geq \beta \}, \quad (4)$$

where  $\beta(>0)$  represents the activation threshold. Match is a match relation defined by:

$$Match = \{ \langle x, y \rangle \mid x, y \in D, f_{match}(x, y) = 1 \}. \quad (5)$$

The affinity function  $f_{match}(x, y)$  may be any kind of Hamming, Manhattan, Euclidean, and  $r$ -continuous matching, etc. In this model, we take  $r$ -continuous matching algorithm to compute the affinity of mature Detectors.

## 2.2 THE DYNAMIC MATURE DETECTOR MODEL

$$M(t) = M(0) = 0, t = 0, \quad (6)$$

$$M(t + \Delta t) = M(t) + M_{new}(\Delta t) + M_{from\_other}(\Delta t) - M_{dead}(\Delta t), \quad \text{if } f_{match}(M(t), Ag(t)) \neq 1, \quad (7)$$

$$M_{clone}(t) = \frac{\partial M_{clone}}{\partial x_{clone}} \cdot \frac{\partial M_{active}}{\partial x_{active}} \cdot \Delta(t-1), \quad (8)$$

if  $f_{match}(M(t), Ag(t)) = 1$ .

$$M.\rho(t + \Delta t) = M.\rho(t) + V_p \cdot \Delta t, \quad (9)$$

$$M.count(t + \Delta t) = M.count(t) + 1,$$

$$M_{new}(\Delta t) = \frac{\partial M_{new}}{\partial x_{new}} \cdot \Delta t = \frac{\partial T_{active}}{\partial x_{active}} \cdot \Delta(t-1), \quad (10)$$

$$M_{dead}(\Delta t) = \frac{\partial M_{death}}{\partial x_{death}} \cdot \Delta t, \quad (11)$$

if  $f_{match}(M(t-1), Self(t-1)) = 1$ ,

$$M_{from\_other}(\Delta t) = \sum_{i=1}^k \left( \frac{\partial M_{from\_other}^i}{\partial x_{from\_other}} \cdot \Delta t \right). \quad (12)$$

Equation (6) depicts the lifecycle of the mature detector, simulating the Agent that the mature detectors evolve into the next generation. All mature detectors have a fixed lifecycle ( $\lambda$ ). If a mature detector matches enough antigens ( $\geq \beta$ ) in its lifecycle, it will evolve to a memory detector. However, the detector will be eliminated and replaced by new generated mature detector if they do not match enough antigens in their lifecycle.  $M_{new}(t)$  is the generation of new mature  $SD$ .  $M_{dead}(t)$  is the set of  $SD$  that haven't match enough antigens ( $\leq \beta$ ) in lifecycle or classified self antigens as *nonself* at time  $t$ .  $M_{active}(t)$  is the set of the least recently used mature  $SD$  which degrade into memory  $SD$  and be given a new age  $T > 0$  and count  $\beta > 1$ . When the same antigens arrive again, they will be detected immediately by the memory  $SD$ . In the mature detector lifecycle, the inefficient detectors on classifying antigens are killed through the process of clone selection. Therefore, the method can enhance detection efficiency when the abnormal behaviours intrude the email system again.

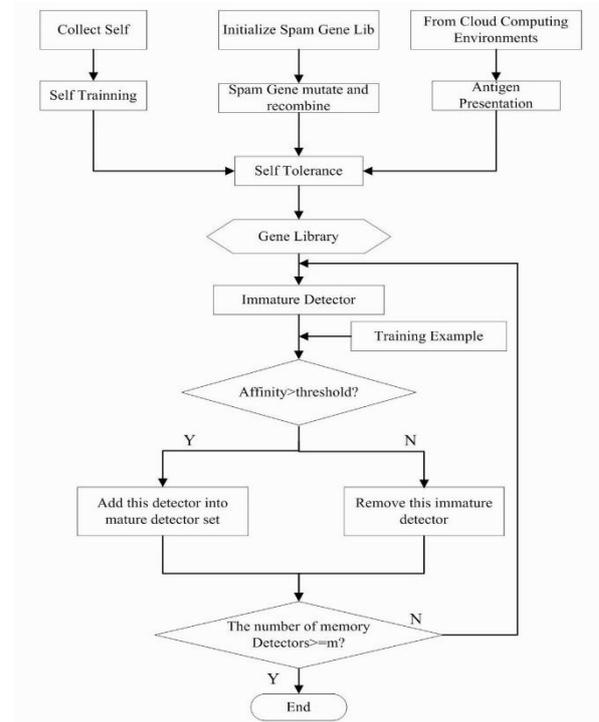


FIGURE 2 The Dynamic Mature Detector Model

As Figure 2 shows, system randomly creates the immature detectors firstly, and then it computes the affinity between the immature detectors and every element of training example. If the affinity of one immature detector is over threshold, it will become a mature detector and will be add into mature detector set.

System repeats this procedure until mature detectors are created.

### 2.3 THE PROCESS OF EMAIL SURVEILLANCE

Our model uses detector state conversion in the dynamic evolution of mature detector and memory detector, erasing and self matching detector. As the Figure 3 shows, the undetected Emails are compared with memory detectors firstly.

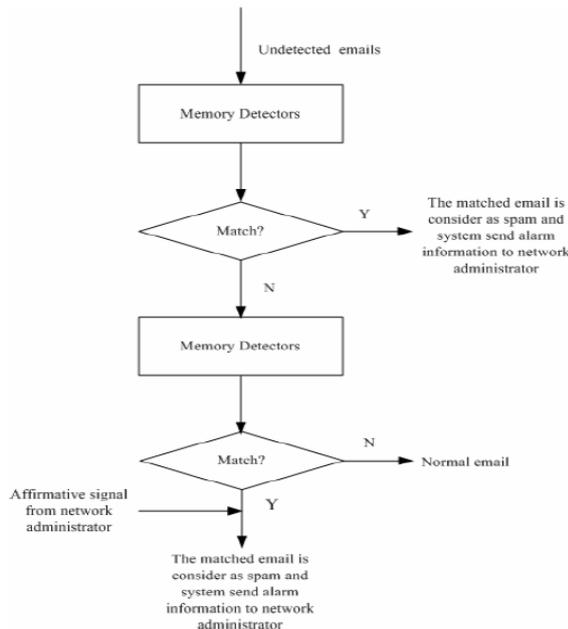


FIGURE 3 The Process of Email Surveillance

If one e-mail match any elements of memory detector set, this Email is classified as spam and send alarming information to user. Then, the remaining Emails which are filtered by memory detectors are compared to mature detectors. Mature detectors must have become stimulated to classify an as junk, and therefore it is assumed the first stimulatory signal has already occurred. Feedback from administrator is then interpreted to provide a co-stimulation signal. If system receives affirmative co-stimulation in fixed period, the matched Email is classified as spam. Or else it is considered as normal Email and delivered to user client in the normal way. During the filtering phase, when a mature detector matches one e-mail, the count field of mature detector will be added. If the value of filed count is over threshold, it will be activated and become a memory detector. Meanwhile, if a memory detector cannot match with any e-mails in fixed period, it will degenerate into a mature detector. When the unsolicited emails and malice intrusions increase, we simulate immune system functions to increase the density of antibody; when they decrease, we simulate immune feedback functions and reduce the density of corresponding antibody, restoring it to normal level.

### 2.4 THE EVALUATION OF THE EMAIL RISK

Owing to the fact that our model relates to enormous factors for evaluation, on purpose of reasonably and entirely measuring the spam email dangerous status, we classify the involved factors as host dangers, area dangers, cells dangers, and special dangers. Afterwards, we subdivide and arrange all the factors which influence the network dangers, in order to let them locate on different layers, forming a structure model with identify matrix.

1) *Construct Identify Matrix*: First of all, we must construct identify matrix which is result that we compared the relative importance of one group of elements on next layer with some past layer element constraint. That is, it shows the relative importance of any pair of factors. In detail, denote  $b_{ij}$  the compared result of the  $i^{th}$  factor and  $j^{th}$  one,  $b_{ij}$  all together form the identify matrix  $B$  :

$$B = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{pmatrix},$$

where:  $b_{ii} = 1$  if  $i = j$  and  $b_{ij} = 1/b_{ji}$  if  $i \neq j$ .

2) *Computing Weights*: Next we obtain the weight of each factor. According to the identify matrix  $B$ , we can get the maximum eigenvalue of the matrix  $\lambda_{max}$ . Here, we can get the maximum  $\lambda_{max}$  according with the following condition:

$$\begin{vmatrix} b_{11} - \lambda & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} - \lambda & \cdots & b_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} - \lambda \end{vmatrix} = 0.$$

Work out the corresponding eigenvector of maximum eigenvalue of  $B$ ,  $X = (x_1, x_2, \dots, x_n)$ , let  $x_i$  to be the weight of factor  $u_i$ , then we can get unitary weights denote  $W_i$ .

$$A = (W_1, W_2, \dots, W_n) = (x_1 / \sum_{i=1}^n x_i, x_2 / \sum_{i=1}^n x_i, \dots, x_n / \sum_{i=1}^n x_i).$$

3) *Test of Consistency*: Because of complexity of evaluation and limit of individual knowledge, the individual identify matrix may not be consistent with the actual one, or the disagreement of any two identify matrixes may result in error of subjective judgment. However, we must test the consistency of the matrix  $B$  as follows:

a) Computing consistency value  $C \cdot I$  :

$$C \cdot I = \frac{\lambda_{\max} - n}{n - 1} \tag{13}$$

b) Computing consistency ratio  $C \cdot R$

$$C \cdot R = \frac{C \cdot I}{R \cdot I} \tag{14}$$

where  $R \cdot I$  is mean consistency value that can be found in the reference and forms, we often consider that if  $C \cdot R$  is smaller than 0.1, the consistency of matrix is acceptable, otherwise we must modify the identify matrix B.

4) *Computing the General Weight Order*: The general weight order means that the weight order comparing the elements in the present layer and the highest layer. We have got each order of element in rule layer to the object layer and the values are  $W_1, W_2, \dots, W_n$ , respectively, we also know that order that design layer to the rule layer and the values are  $W_1^j, W_2^j, \dots, W_n^j$ , then the general order is

$$V = W^j W = \begin{pmatrix} W_1^j & W_1^j & \dots & W_1^j \\ W_2^j & W_2^j & \dots & W_2^j \\ \vdots & \vdots & \ddots & \vdots \\ W_n^j & W_n^j & \dots & W_n^j \end{pmatrix} \begin{pmatrix} W_1 \\ W_2 \\ \vdots \\ W_n \end{pmatrix} = \begin{pmatrix} V_1 \\ V_2 \\ \vdots \\ V_n \end{pmatrix} \tag{15}$$

2.5 EVALUATING THE DANGER LEVEL

The entire network of spam danger level should fully reflect the value of each of the host facing attacks. Let  $n_{ij}(t)$  be the numbers of  $i^{th}$  spam detect attacking at time t. Let  $\beta_i (0 \leq \beta_i \leq 1)$  be the importance coefficient of  $i^{th}$  computer in the network and  $\alpha_j (0 \leq \alpha_j \leq 1)$  be the danger coefficient of the  $j^{th}$  kind of attack in the network. Therefore, we can get the spam danger  $R(t)$  situation and evaluate security at real time.

$$R_j(t) = \frac{2}{1 + e^{-\alpha_j \sum_i \beta_i n_{ij}(t)}} - 1 \tag{16}$$

The conclusion can be shown that the higher value  $R(t)$  reaches the more dangerous the network is.

3 Experimental results and analysis

Experiments of simulation were carried out in our Laboratory. The main aim of the experiment was to test the feasibility of the application for anti-spam based on AIS to implement spam detecting. And we developed some series experiments. Here are the coefficients for the model as the Table 1 showing.

TABLE 1 Coefficients for the model

Parameter	Value
r-contiguous bits matching rule	8
The size of initial self set n	40
The Initial Scale of Detectors	100
Match Threshold $\beta$	40~60
Activable Threshold $\lambda$	50~150
Clone Scale	20
Mutation Scale	19
Life Cycle of Mature Detectors	120s

We prepared the Ling-Spam datasets for analysis and experiments. A mixture of 481 spam messages and 2412 messages sent via the Linguist list, a moderated list about the profession and science of linguistics. Attachments, HTML tags, and duplicate spam messages received on the same day are not included. The whole experiment is divided into two phase: training phase and application phase. The main different between the two phases is that the former does not use filtering module and just generates detectors for system. We partitioned the emails randomly into ten parts and choose one part randomly as a training example, then remaining nine parts are used for test and we can get 9 group recall and precision ratios. The average value of these 9 group values is considered as the model's recall and precision ratio.

Traditional spam filters system and technology almost adopted static measure, however, lack self-adaptation and the ability of parallel distributed processing. In this paper, we have presented a model of spam detection based on the theory of artificial immune system, and we have also illustrated the advantages of this model than traditional models. The concepts and formal definitions of immune cells are given. And we have quantitatively depicted the dynamic evolutions of self, antigens, immune-tolerance, and the immune memory. Additionally, the model utilized a distributed and multi-hierarchy framework to provide an effective solution for the spam. Finally, the experimental results show that the proposed model is a good solution for anti-spam system.

Acknowledgments

This work is supported by the China Postdoctoral Science Foundation (No.2012T50783, No.2011M501419), and Sichuan Provincial Department of Science and Technology Project (No.2014JY0036), and Scientific Research Fund of Sichuan Provincial Education Department (No.13TD0014), and Leshan Normal University of Achievements Transformation Project (No.Z1322), and Science and Technology Key Research Project of Leshan (No.12GZD014).

## References

- [1] [http://en.wikipedia.org/wiki/Cloud\\_Computing](http://en.wikipedia.org/wiki/Cloud_Computing)
- [2] <http://www.mysql.com/news-and-events/sun-to-acquire-mysql.html>
- [3] Garg S K, Versteeg S, Buyya R 2013 A framework for ranking of cloud computing services *Future Generation Computer Systems* **29**(4) 1012-23
- [4] Patel A, Taghavi M, Bakhtiyari K 2013 An intrusion detection and prevention system in cloud computing: A systematic review *Journal of Network and Computer Applications* **36**(1) 25-41
- [5] Luo J-Z, Wu W-J, Yang M 2011 Mobile Internet: Terminal Devices, Networks and Services *Chinese Journal of Computers* **34**(11) 2029-51
- [6] Mezmaz M, Melab N, Kessaci Y 2011 A parallel bi-objective hybrid metaheuristic for energy-aware scheduling for cloud computing systems *Journal of Parallel and Distributed Computing* **71**(11) 1497-508
- [7] Subashini S, Kavitha V 2011 A survey on security issues in service delivery models of cloud computing *Journal of Network and Computer Applications* **34**(1) 1-118
- [8] Kshetri N 2013 Privacy and security issues in cloud computing: The role of institutions and institutional evolution *Telecommunications Policy* **37**(4) 372-86
- [9] Rong C, Nguyen S T, Jaatun M G 2013 *Computers & Electrical Engineering* **39**(1) 47-54
- [10] Villa O, Petrini F 2008 Accelerating real-time string searching with multicore processors *Computer* **41**(4) 42-4
- [11] Burnet F M The 1959 Clonal Selection Theory of Acquired Immunity *Cambridge Cambridge University Press*
- [12] Kepler T B, Perelson A S 1993 Somatic hypermutation in B cells: An optimal control treatment *Journal of Theoretical Biology* 37-64
- [13] Forrest S, Perelson A S, Allen L, Cherukuri R 1994 Self-Nonself Discrimination in a Computer *Proceedings of IEEE Symposium on Re-search in Security and Privacy Oakland*
- [14] Kim J, Bentley P 1999 The Artificial Immune Model for Network Intrusion Detection *the 7<sup>th</sup> European Congress on Intelligent Techniques and Soft Computing*
- [15] Artin-Herran G, Rubel O 2008 Zaccour G Competing for consumer's attention *Automatica* **44** 361-70 (in Chinese)
- [16] Hanke M 2008 On the effects of stock spam e-mails *Journal of Financial Markets* **11** 57-83
- [17] Li T 2007 An Introduction to Computer Network Security. 1<sup>st</sup> edition *Publishing House of Electronics Industry Beijing*

Authors	
	<p><b>Jin Yang, born on June 9, 1980, Sichuan, China</b></p> <p><b>Current position, grades:</b> associate professor at LeShan Normal University, PhD.</p> <p><b>University studies:</b> Ph.D in computer science at Sichuan University, Sichuan in 2007.</p> <p><b>Scientific interest:</b> network security, artificial immune, knowledge discovery, expert systems.</p> <p><b>Publications:</b> 20.</p>
	<p><b>Lingxi Peng, born on July 23, 1978, Guangzhou, China</b></p> <p><b>Current position, grades:</b> professor at department of computer and education software, PhD.</p> <p><b>University studies:</b> Ph.D in computer science from Sichuan University, Sichuan in 2008.</p> <p><b>Scientific interest:</b> Network security, artificial immune, knowledge discovery, expert systems.</p> <p><b>Publications:</b> 20.</p>
	<p><b>Tang Liu, born on January 20, 1979, Chengdu Sichuan, China</b></p> <p><b>Current position, grades:</b> associate professor in Sichuan Normal University, PhD student at the College of Computer Science, Sichuan University, Chengdu, China.</p> <p><b>University studies:</b> M.S. degree at college of computer science, Sichuan University, China, in 2009.</p> <p><b>Scientific interest:</b> wireless sensor networks.</p> <p><b>Publications:</b> 15.</p>