Artificial Immune System Based Network Intrusion Detection System for Cloud

Mahek S. Chheda, Sunny K. Gopani, Pragyan Nanda

Department of Computer Engineering,

MCT's Rajiv Gandhi Institute of Technology, Andheri, Mumbai-400 053, India.

sunny.gopani@gmail.com

mahekchheda11@gmail.com

n.pragyan@gmail.com

***Abstract*— Cloud computing is one of the major upcoming technology that has recently taken the attention of both academia and industry. Cloud Computing provides the way to share distributed resources and services that belong to different organizations or sites. Since Cloud Computing share distributed resources via network in the open environment, it is vulnerable to various network attacks like DOS(denial of service), DDOS(distributed denial of service), UDP flooding, TCP-SYN. Thus we have implemented a system for securing a private cloud using a Network Intrusion Detection System(NIDS). This NIDS is based on Artificial Immune System (AIS) which is a distributed, robust, dynamic, diverse and adaptive security mechanism. It captures many features of the human immune system and places them in the context of the problem of protecting a cloud environment from illegal intrusions. Its sole purpose is to differentiate normal connections from abnormal network connections. The experimental results show that this approach 70% of the time successfully detects a variety of denial of service attacks targeted on a eucalyptus based cloud infrastructure and simultaneously generates alerts through an alerting system.**

***Keywords*— Cloud Computing, Artificial Immune System, Network Intrusion Detection System, Denial of Service.**

INTRODUCTION

The National Institute of Standards and Technology's (NIST) Information Technology Laboratory recognizes that cloud computing is an "evolving paradigm"[5]. Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.It has so many advantages such as economy, complex calculations, agility, high scalability, high reliability, easy maintenance.

Virtualization plays a vital role in creating cloud applications which can dynamically consume resources when required for providing services. Commonly used Virtualization technologies are from VMWare, the open source community Xen, Kernal Virtual Machine (KVM). This cloud model promotes availability and is composed of five essential characteristics namely on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service. It supports three service models namely Cloud Infrastructure as a Service (IaaS), Cloud Software as a Service (SaaS) and Cloud Platform as a Service(PaaS). [5]Cloud IaaS provides cloud consumer the capability to provision processing, computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. [5]The Cloud SaaS model provides consumer the capability to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web based email). The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user specific application configuration settings. [5]The Cloud PaaS model provides consumer the capability to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly application hosting environment configurations. There are four cloud deployment models namely public, community, private and hybrid. This paper presents a Network Intrusion Detection System (NIDS) for a private cloud. Today, many organizations are moving their computing services towards the Cloud. This makes their computer processing available much more conveniently to users. However, it also brings new security threats and challenges about safety and reliability. In spite of being attractive, Cloud feature poses various new security threats like Denial of Service attacks (DOS), Distributed Denial of Service attacks (DDOS) based on various protocols like UDP, TCP, ICMP etc. and challenges when it comes to deploying Intrusion Detection System (IDS) in Cloud environments. Most Intrusion

Detection Systems (IDSs) are designed to handle specific types of attacks. It is evident that no single technique can guarantee protection against future attacks. Hence, there is a need for a system which can provide robust protection against a big spectrum of threats.

Recent years, the artificial immune system has the features of dynamic, self-adaptation and diversity [2] that just meet the constraints derived from the characteristics of the grid environment, and mobile agent has many same appealing properties as that of artificial immune system. Negative Selection Algorithm and the concept of computer immunity proposed by Forrest in 1994 [2]. In contrast, the AIS theory adaptively generates new immune cells so that it is able to detect previously unknown and rapidly evolving harmful antigens. However, much theoretical groundwork in immunological computation has been taken up, but there is a lack of perfectly systems based AIS of dynamical immunological surveillance for network security [2]. Based on the correspondence between the artificial immune system antibody in the artificial immune systems an pathogen invasion intensity, this paper is to establish a network risk evaluation model. We built a hierarchical, quantitative measurement indicator system, and a unified evaluation information base and knowledge base. This model will help the network managers evaluate the possibility and the graveness degree of the network dangerous quickly, ease the pressure of recognition, to get targeted immediate defence strategy of the strength and risk level of the current network attacks.

The rest of this paper is organized as follows. Section II discusses Biological Immune System. Section III discusses Artificial Immune System (AIS). Section IV presents AIS algorithm. Section V describes the architecture of our NIDS. In Section VI and VII, we present the key advantages and the experimental results of our proposed scheme.

II. BIOLOGICAL IMMUNE SYSTEM

The biological immune system(BIS) is a robust, complex, adaptive system that defends the body from foreign pathogens. It is able to categorize all cells (or molecules) within the body as self-cells or nonself cells[1]. It does this with the help of a distributed task force that has the intelligence to take action from a local and also a global perspective using its network of chemical messengers for communication. There are two major branches of the immune system. The innate immune system is an unchanging mechanism that detects and destroys certain invading organisms, whilst the adaptive immune system responds to previously unknown foreign cells and builds a response to them that can remain in the body over a long period of time. This remarkable information processing biological system has caught the attention of computer science in recent years.

The key elements of a highly simplified immune system are illustrated in Fig 1[6]. Harmful items are proteins called antigens. In Fig 1 the antigens are coloured red and have sharp corners. The human body also contains many non-harmful antigens called self-antigens, or self-items. These are naturally occurring proteins and in Fig 1 are coloured green and have rounded sides. Antigens are detected by lymphocytes. Each lymphocyte has several antibodies, which can be thought of as detectors. Each antibody is specific to a particular antigen. Typically, because antibody-antigen matching is only approximate, a lymphocyte will not trigger a reaction when a single antibody detects a single antigen. Only after several antibodies detect their corresponding antigens will a lymphocyte become stimulated and trigger some sort of defensive reaction. Notice that no lymphocyte has antibodies that detect a self-item. Real antibodies are generated by the immune system in the thymus, but any antibodies that detect self-items are destroyed before being released into the blood stream, a process called apoptosis.

The BIS consists of a multitude of cells and molecules which interact in a variety of ways to detect and eliminate infectious pathogens (antigens). These interactions are localized because they depend upon chemical bonding-surfaces of immune system cells are covered with receptors, some of which chemically bind to antigens, and some of which bind to other immune system cells or molecules to enable the complex system of signalling that mediates the immune response. Most BIS cells circulate around the body via the blood and lymph systems, forming a dynamic system of distributed detection and response, where there is no centralized control, and little, if any, hierarchical organization. Detection and elimination of antigens is a consequence of trillions of cells interacting through simple, localized rules. A consequence of this is that the BIS is very robust to failure of individual components and attacks on the BIS itself. The problem of detecting antigens is often described as that of distinguishing self from nonself .However, many antigens are not harmful, and an immune response to eliminate them may damage the body. In these cases it would be healthier not to respond, so it would be more accurate to say that the problem faced by the BIS is that of distinguishing between harmful nonself and everything else [2]. We adopt the viewpoint that nonself is synonymous with any pathogen that is harmful to the body, and self is synonymous with harmless substances, including all normally functioning cells of the body.

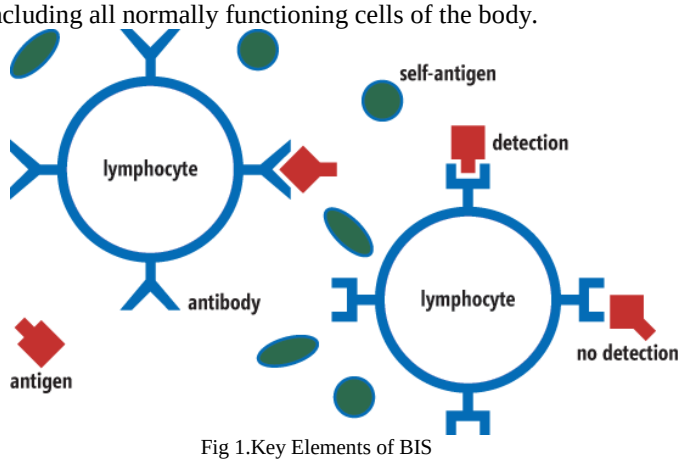

Fig 1.Key Elements of BIS

Once antigens have been detected, the BIS must eliminate them in some manner. Different pathogens have to be eliminated in different ways, and we call the cells of the BIS that accomplish this as detectors. The elimination problem facing the immune system is that of choosing the right detectors for the particular kind of pathogen to be eliminated.

III. ARTIFICIAL IMMUNE SYSTEM

The biological immune system is highly complicated and appears to be precisely tuned to the problem of detecting and eliminating infections. It provides an efficient way to design an artificial adaptive system. An important and natural application domain for adaptive systems is that of Intrusion detection system (IDS). IDS are broadly classified as host based IDS (HIDS) and network based IDS (NIDS). This paper presents a Network IDS that uses the concept of AIS for detecting a wide spectrum of network attacks targeted on a cloud infrastructure. Differentiating between normal and intrusive activities is the major role played by the IDS. AIS, which is a biologically inspired computing, is currently investigated to solve this problem[2]. It obtains a solution by four stages namely Encoding, Similarity or Affinity measure, Negative selection, mutation.

To preserve generality, we represent incoming packets into self(protected agents) and nonself(antigens/infectious agents) as dynamically changing sets of bit strings. In cells of the human body the profile of expressed proteins (self) changes over time, and likewise, we expect our set of protected strings to vary over time. Similarly, the body is subjected to different kinds of infections, diseases; we can view nonself as a dynamically changing set of strings. The sets of bit strings are nothing but the result of our encoding algorithm. Every incoming packet in our cloud network is intercepted by the IDS server. It is filtered according to the protocols and encoded based on the packet details. In our representation, every packet is encoded into l= 12 bits string of 0s and 1s which unambiguously defines the connection. Each bit is a Boolean value based on the conditions that are placed against the packet details. The various fields that we have considered for encoding purpose are source IP address, destination IP address, source port address, destination port address, time difference between current and previous packet, header length, payload length, hop count, window size, acknowledgement number etc. Natural immune systems consist of many different kinds of cells and molecules—lymphocytes (B-lymphocytes and Tlymphocytes), macrophages, dendritic cells, natural killer cells, mast cells, interleukins, interferon, and many others. In our model, we simplify this by using a detector cell which combines the properties of different kinds of biological immune cells. The detector cell takes up many forms in its lifecycle such as immature antibodies, naïve antibodies and memory cells.

The similarity measure or matching rule is one of the most important design choices in developing an artificial immune system algorithm, and is closely coupled to the encoding scheme. It is derived using a process called binding. Binding of an incoming foreign material encoded into a bit string with a naïve antibody is implemented by using a string matching algorithm where each detector is a string , and detection of a string occurs when there is a match between a and b, according to a matching rule. We use string matching because it is simple and efficient to implement, and easy to analyse and understand. Our matching rule is based on an immunologically plausible rule, called r-contiguous bits [2]. Two strings a and b match under the r-contiguous bits rule if a and b have the same symbols in at least r ontiguous bit positions. The value r is a threshold and determines the specificity of the detector, which is an indication of the number of strings covered by a single detector. For example, if r=l, the matching is completely specific, that is, the detector will detect only a single string (itself; recall that l is the length of the detector bit string).

The detectors are grouped into sets, one set per node controller and one per cloud controller, on the network; each node loosely corresponds to a different location in the body. Because of the broadcast assumption, each detector set is constantly exposed to the current set of connections in the network, which it uses as a dynamic definition of self. On a continual basis, each detector set is used to generate new detector set or antibodies randomly and asynchronously using permutation similar to the human immune system. These newly generated antibodies remain immature for a certain period, during which they have an opportunity to match with any current incoming encoded packets. If a detector matches when it is immature, it is killed (deleted). This process is called negative selection [3], and closely resembles the negative selection of immature T-lymphocytes (thymocytes) in the thymus. Detectors that survive this initial testing phase are promoted to mature antibodies (analogous to mature T-lymphocytes leaving the thymus and mature B-lymphocytes leaving the bone marrow). Each antibody now acts independently and binds with the antigens from next time. Now if the antibody matches a sufficient number of packets, it takes the form of naïve antibody. The transition from immature phase to naïve phase can be called as learning phase. At the end of the learning phase, if the detector fails to match any packet then it is deleted from the database. But if it has matched a sufficient number of nonself packets, it becomes a memory cell with a greatly extended lifetime. Memory cell have a lower threshold thus implementing a “secondary response” that is quicker and more aggressive than naïve detectors to previously seen strings.

A detector is initially created by mutating the 12 bit incoming encoded string. It remains immature for a certain period of time. This period is called as tolerization period. If the detector matches any string a single time during tolerization it is killed (deleted). If a detector survives

immaturity, it will exist for a finite lifetime. At the end of that period, it is replaced by a new randomly generated detector string. But if it has exceeded its match threshold, it is promoted to a memory mature antibody. If the threshold is exceeded for a memory cell, it is activated. If an activated memory cell does not receive stimulation it dies. However, if it receives stimulation, it enters the competition to become memory cell. Memory cells have a very small threshold to become activated. Following Fig 2 shows lifecycle of a detector [4].

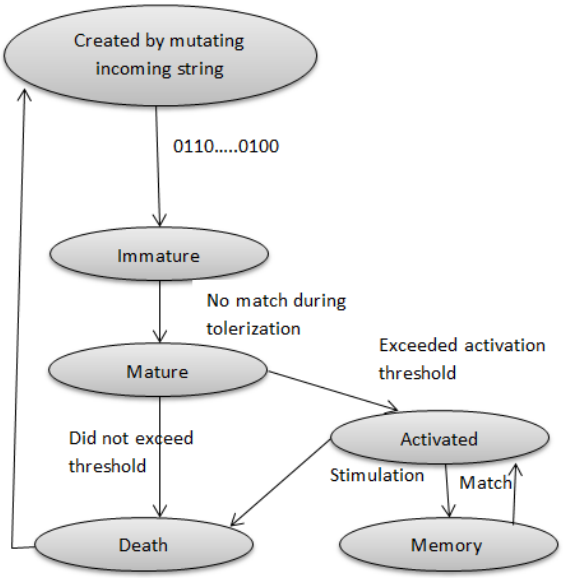

Fig 2. Life cycle of a detector [2]

IV. AIS ALGORITHM

The AIS algorithm is divided into two phases that is used in our NIDS: a) Generation of antibodies, b) Generation of memory cells.

```
Generation_antibodies(received_packet)
p ← received_packet;
while p  do
pro ←  packet_protoco
srcip ← source_ip
destip ← destination_ip
srcport ← source_port
destport ← destination_port
hcount ←hopcount
tdiff ← time_difference
l1 ← header_length
l2 ←payload length
pro ← protocol
snumber ← sequence_number
frag ← is_frament
cell←encode(srcip,destip,srcport,destport,hcount,tdiff,l1,l2,pr
o,snumber,frag)
detector ← mutate(cell)
```

```
//now detector is entered into the database.
//if it matches with any database records, both the entries are
deleted.


Genration_memorycells(received_packet)
  p ← received_packet;
while p do
pro ← packet_protoco
srcip ← source_ip
destip ← destination_ip
srcport ← source_port
destport ← destination_port
hcount ←hopcount
tdiff ← time_difference
l1 ← header_length
l2 ←payload length
pro ← protocol
snumber ← sequence_number
frag ← is_frament

cell←encode(srcip,destip,srcport,destport,hcount,tdiff,l1,l2,pr
o,snumber,frag)
//open connection to memory cell database
while resultset!= 0 do
if cell = database entries
stimulation ← stimulation + 1;
if stimulation > threshold1
alert()
//open connection to antibodies database
while resultset!= 0 do
if 10_bits_cell = database_antibody_entry
stimulation ← stimulation + 1;
if stimulation > threshold2
// enter the antibody into memory cell database table.
break;
```

The first algorithm generates antibodies by negative selection and mutation while the second algorithm generates alerts depending on the 10 continuous bit matching of memory cell to received encoded packet.

V. ARCHITECTURE OF NIDS

Fig 4.1 summarizes the deployment of the NIDS on a cloud infrastructure. It comprises an external network, internetwork, router, cloud architecture, attacker, client, distributed NIDS. Our NIDS is deployed on the cloud controller as well as node controller. It is distributed because all the workstations share the IDS database. Database is updated timely and redundancies are removed if any. The attacker can reside both local network and external network. Every incoming packet is intercepted at the Cloud controller. A DHCP server is used to assign IP addresses to client workstations (cloud consumers). We have used Eucalyptus for building a private cloud. Eucalyptus is infrastructure

software that enables enterprises and government agencies to establish their own cloud computing environments. It provides APIs compatible to the popular Amazon Web Services offerings: EC2, S3 and EBS (Elastic Book Store), thus allowing access to wide variety of cloud tools and option of building hybrid clouds. Eucalyptus is comprised of six components: Cloud Controller, Walrus, Cluster Controller (CC), Storage Controller (SC), Node Controller (NC) and an optional VMware Broker.

The CLC queries other components for information about resources, makes high-level scheduling decisions, and makes requests to the Cluster Controllers (CCs).). As the interface to the management platform, the CLC is responsible for exposing and managing the underlying virtualized resources (servers, network, and storage).

Walrus allows users to store persistent data, organized as buckets and objects. It is used to create, delete, and list buckets, or to put, get, and delete objects, or to set access control policies. The Cluster Controller (CC) generally executes on a machine that has network connectivity to both the machines running the Node Controllers (NCs) and to the machine running the CLC. CCs gather information about a set of node machines and schedules virtual machine (VM) execution on specific node controllers. The Node Controller (NC) executes on any machine that hosts VM instances. The NC controls VM activities, including the execution, inspection, and termination of VM instances. It also fetches and maintains a local cache of instance images, and it queries and controls the system software (host OS and the hypervisor) in response to queries and control requests from the CC.

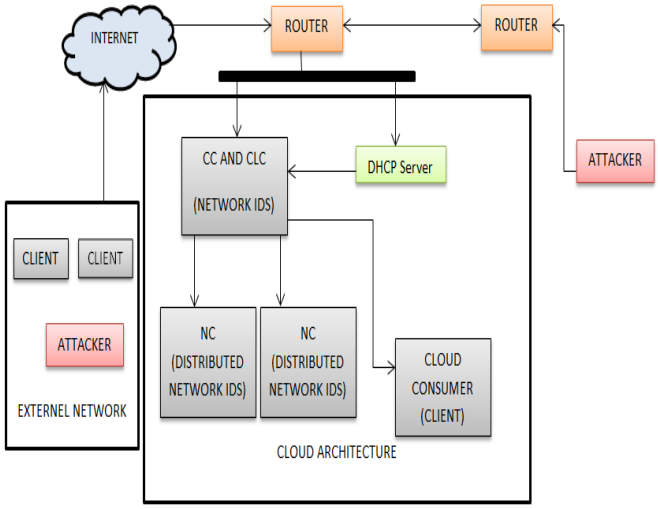

Fig 3. Deployment of NIDS on a private cloud

As shown in Fig. 4 ,when a packet arrives in a network it is encoded in a prescribed format into a string. This string is then compared with the set of antibodies so created to detect any abnormal activity; which is obtained by matching upto r-contiguous bits. If the matching goes above threshold then that antibody is activated and abnormal pattern is detected which can be used for further reference in studying the attack. The concept of memory cell allows us to use signature and anomaly based IDS together.

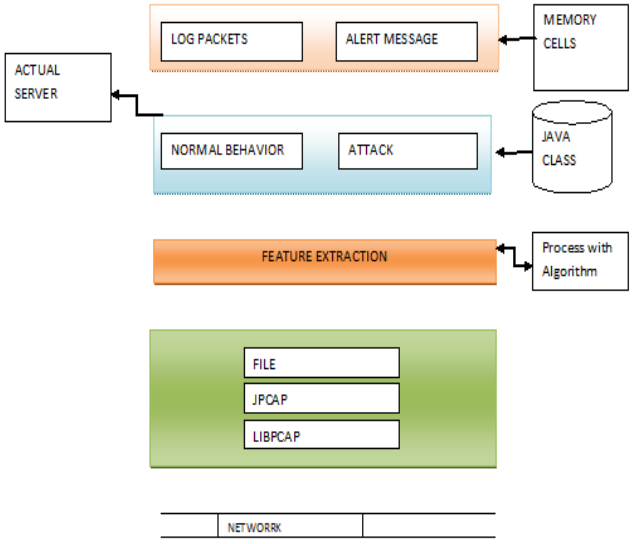

Fig. 4 Architecture of NIDS

VI. EXPERIMENTAL RESULTS

As described in above sections, the algorithm was deployed real-time on a cloud server and to validate the effect of the AIS based system. The real problem faced during the system simulation is selection of appropriate threshold and stimulation value. The problem here is, if the value of threshold is kept low then the probability of false positive increases and if the value is kept high then even the malicious or abnormal packets might go undetected. So, it is necessary to select an appropriate value of threshold and stimulation to balance out the difference and make system work optimally. The table 1 below gives information about false alarm and true positive obtained for different values of activation threshold. FP stands for false positive alarm, NP stands for false negative alarm and D stands for successful detection.

Sr no.	Threshold	Stimulation	Result
1	5	50	FP
2	6	60	FP
3	7	80	FP
4	8	100	FP & D
5	9	120	FN & D
6	5	140	FP & D
7	6	160	FP & D
8	7	180	FP & D
9	8	200	D
10	9	250	FN &D

Table 1. Results: UDP flooding attack

.

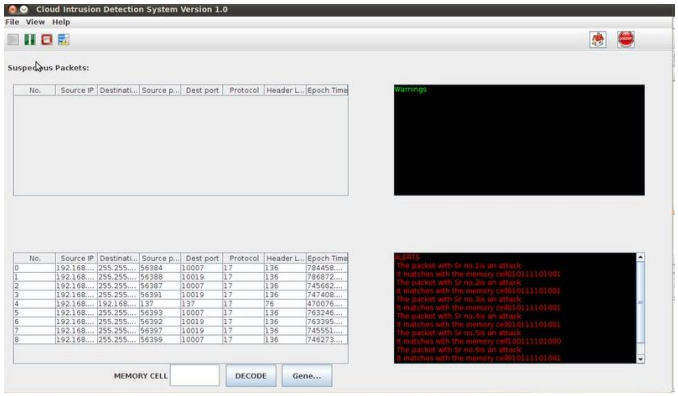

Fig 5 Detection results.

VI. ADVANTAGES

Like the Human Immune System the Artificial Immune system demonstrates following properties in any network scenarios:

Diversity: The AIS model can just detect any kind of diverse and anomalous network traffic which makes up for a better detection system.

Distributed Computation: The different components of cloud infrastructure work together and maintain a common memory-cell to provide direct detection and prevention of attacks.

Dynamic Learning: This probably the most important property of any intrusion detection system. The antibodies set so generated are designed to bind with antigens. This set is capable of binding with any antigen hence; the new anomalous activity can be easily detected.

Self-monitoring: If some antigens are detected commonly then these antibodies are straight away added as memory cells so next time if those antigens are detected instantly and user is alerted.

CONCLUSIONS

The self-adaptability, distributed character and quantization of antibody concentration that biological immune system bears are just the effective method to solve the technological problems of network security situation awareness, thus this article applies the characteristics of biological immune mechanism in the field of network security situation awareness research, and establishes immune network security situation awareness system to make real-time and quantitative analysis on network security condition and its changing trend in a cloud computing network.. This paper combines the risk evaluation methods with application security engineering principles, and can change current passive defence situation using traditional network security approaches, and is helpful to establish new generation proactive defence theories and realization techniques. At the same time, the work is of not only theoretic values to design proactive defence systems which have intrusion tolerant ability and survivability in any complex network circumstances, but also very significant to protect network infrastructure. The experimental results show that the proposed model has the features of real-time processing that provide a good solution for network surveillance and can solve cloud's many security problems.

REFERENCES

[1] U. Aickelin and D. Dasgupta. Artificial Immune Systems Tutorial, Kluwer, 2005

[2] S A Hofmeyr, and S Forrest, Architecture for an artificial Immune system, Evolutionary Computation, vol. 8, pp. 443-473, 2000

[3] Kim, J. and Bentley, P. (2000), "Negative Selection within an Artificial Immune System for Network Intrusion Detection", the 14th Annual Fall Symposium of the Korean Information Processing Society, Seoul, Korea.

[4] S A Hofmeyr, and S Forrest, Immunity by design- an artificial Immune system, 1999

[5] Gtsi, Cloud Computing: Building a Framework for Successful Transition.

[6] James McCaffrey, Test Run:Artificial Immune Systems for Intrusion Detection, MSDN Magazine, Issues, 2013.

[7] Karen Scarfonem Peter Mell, Guide to Intrusion Detection and Prevention System, NIST Publication 800-94.

[8] www.eucalyptus.com.

[9] www.wikipediacom.