

Cloud Immunization and Security for e-Governance Applications

Rama Krushna Das*, Manas Ranjan Patra ** Ajita Kumar Misro**

* National Informatics Centre, Berhampur, Odisha, India

** PG Department of Computer Science, Berhampur University, Odisha, India

Article Info

Article history:

Received Mar 10th, 2013

Revised Apr 15th, 2013

Accepted May 10th, 2013

Keyword:

CloudComputing

Cloud Security

e-Governance

Open Source

CLONA

ABSTRACT

Different e-Governance applications in India are using Cloud for making the services scalable, stretchable and cost effective. Starting from the IT centres being setup at Panchayat level to the State/National data centres use the Cloud to create a common infrastructure that would be accessible by all. The focus is to enable sharing of resources, ensure security and take technology to the smaller towns and villages. But the major concern is to ensure security. This paper proposes a security solution by using architectural framework, open source products and immunization algorithm. Our interest is to use Artificial Immune System (AIS) with Clonal Selection Algorithm (CLONA) for secure transaction of e-Governance services. The proposed Cloud architecture adopts the learning process and follows security optimization techniques. This technique uses spontaneous action-event transactional state of Cloud Immunization and Security (CIS), defined security services such as Authentication, Firewall and Antivirus. With these technique and services the CIS system is empowered to dynamically determine the best clone and the best antibody. Intruder attacks termed as new antigens when approaches the Cloud, then the cloud system's antibody known as threat detector follows Hamming Distance calculation to evaluate the threat termed as "affinity". These affinity alerts and protects the Cloud system through CIS to undertake any kind of future attempt and attacks by the intruders.

*Copyright © 2013 Institute of Advanced Engineering and Science.
All rights reserved.*

Corresponding Author:

Ajita Kumar Misro,
Ph.D. Research Scholar
PG Department of Computer Science,
Berhampur University,
Berhampur-7, Bhanja Bihar, Ganjam, Odisha, INDIA.
Email: misroajit@gmail.com

1. INTRODUCTION

Recent trend of e-Governance application in India is to use cloud computing services widely to benefit from elasticity and dynamic on-demand provisioning, and to reduce costs with the cloud's pay-as-you go billing model. As many cloud based e-governance applications involve multiple collaborating of different departments shared resources, the major concern is security [1]. Most of e-Governance projects and applications suffer with security breaches as a result sensitive data loss occurs. Though cloud computing is cheaper to provide e-Governance services but not free from security issues [2][3]. At this juncture our paper explains some open source based cloud security architecture [4] and trying to improve the security system using Artificial Immune System (AIS)[5]. Section 2 deals with brief view of cloud computing. Section 3 explains AIS and immunization system. Section 4 describes selection mechanism in clonal algorithm. Section 5 is our contribution of this paper which deals with open source based secure cloud architecture and CLONA implication to it, to makes the security system more robust. Section 6 represents the architecture of e-Governance and states how this distributed intra cloud system could be made safe and secure.

2. CLOUD COMPUTING

Cloud computing is one of the new models 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 [6]. Cloud computing is a layered architecture consisting of several layers like Physical, Virtualization, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS) as shown in Figure 1. The layers are connected to each other by User Interfaces (UI), Application Programming Interfaces (API) and middleware.

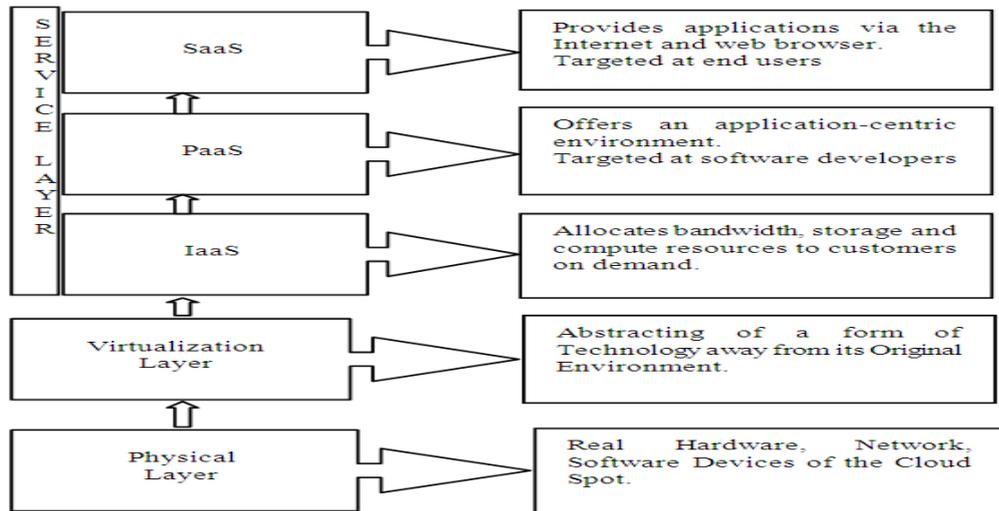


Figure 1. Layered Cloud Architecture

3. AIS AND IMMUNISATION SYSTEM

AIS is a computational systems inspired by the principles and processes of the vertebrate immune system. The field of AIS is mainly concerned with the structure and functions of the immune system into the computational systems. It investigate the application of these systems towards solving computational problems from mathematics, engineering, and information technology. AIS are adaptive systems, inspired by theoretical immunology and observed immune functions, principles and models, which are applied to problem solving. Basically an immune system has some properties i.e. detection, diversity, learning and tolerance.

Detection: Identification takes place in an immune system when the infective fragment and sensory receptor on lymph cell surface is bonded chemically.

Diversity: Identification in an immune system is related to non-self bodies of the organism, thus the immune system has a number of sensory receptor, out of which some of the lymph cells will react with the foreign organism.

Learning: An immune system has the capability of detecting and eliminating the foreign organism as soon as possible from the human body. This principle allows the lymphocytes to find out and adjust themselves to specific foreign protein structure. It is done by the B-cells.

Tolerance: The particles which are mark themselves as self bodies are contain in the chromosomal section.

4. SELECTION MECHANISM IN CLONAL

The Clonal selection algorithm is used by AIS to define the basic features of an immune response to an antigenic stimulus [7][8]. Clonal selection mechanism describes the basic features of an immune response to an antigenic stimulus. It establishes the idea that only those cells that recognize the antigen proliferate, thus being selected against those that do not.

The main features of the clonal selection theory are [9][10]:

- The newly cells are replica of their parents (clone) which are submitted to a chromosomal mutation chemical mechanism.
- Evacuation of newly distinguished lymph cell carrying self - reactive sensory receptor.

- Development and differentiation on contact of mature cells with antigens.

When an antibody strongly matches an antigen the corresponding B-cell is stimulated to produce clone of itself that then produce more antibodies. This (hyper) mutation, is quite rapid, often as much as one mutation per cell division (de Castro and Von Zuben's 1999). The hyper mutation process enables the new cells to match the antigen more closely. The B -cells with high affinity are selected to differentiate into memory cells which do not secret antibodies but instead remember the antigen pattern. The B -cells that are not simulated as they do not match any antigens in the human body will eventually die. Once the body has successfully defended against an antigen memory cells remain and circulate in the blood, lymph and tissues for the very long period of time. When the same or similar antigen is encountered in the future, memory cells are simulated and more abundant production of antibodies take places. It allows a very quick response to the antigens.

Clonal selection mechanism is inspired by acquired immunity which explains how B and T lymphocytes improve their response to antigens over time called affinity maturation. It is basically focused on the Darwinian [11] attributes of the theory where selection is inspired by the affinity of antigen-antibody interactions, reproduction is inspired by cell division, and variation is inspired by somatic hyper mutation. Clonal selection algorithms are mostly commonly applied to optimization and pattern recognition domains.

The intense role of immune system is to protect the body from the foreign beings. The immune system has the capable for differentiate among the own constituents of our beings and foreign stuff which can damage us. This foreign stuff is known as antigen. The important role played by the immune system is the antibodies. When an antigen is noticed in our body then those antibodies which can distinguish the antigen will multiply by cloning. This procedure is called is Clonal Selection Method. The clonal selection algorithm can be described as follows:

- Generating a set of candidate keys (K).
- Determining the n best keys (K*) among the set of the candidate from based on their affinity measures.
- Cloning or reproducing the n best individuals(C). The clone size is the increasing function of the affinity with the antigen.
- Submit the population of clone to hyper mutation, where hyper mutation is directly proportional to the affinity of the antibody with the antigen. A matured antibody set is generated (C*).
- Reselecting the improved individuals from the matured clone set (C*). Some of the member of candidate key (K) is replaced by the improved members of cloned set (C*).
- The lower affinity cells have a higher probability of being replaced.

The immune system is to run over how to distinguish the self bodies from the non self bodies. The immune system is used to protect the human bodies from the extraneous stuff which are injurious to the organism. The extraneous stuff may be the bacteria, virus, pollen grains, incompatible blood cells and manmade particles.

The clonal selection theory is a theory postulated by Burnet, Jerne, Talmadge, used to describe the functioning of acquired immunity, specifically a theory to describe the diversity of antibodies used defends the organism from invasion [12]. Antibodies are the particles which are produced by the B- lymphocyte cells that are used to neutralize single antigen. B-lymphocytes or white blood cells produce a single or customized antibody of a particular type. Nowadays clonal selection theory is one of the overtaking measures of empiric demonstration. The mechanism of clonal selection process [13] is shown in figure 2.

This hypothesis determines that the self bodies have a pre-existing pool of individually specific antibodies which can be recognized with all the antigens with some particularity. When the antigen is matched with a specific antibody, a chemical bonding takes place and replication takes place i.e. more cells are generated with same sense organ or sensory receptor. During the development stage mutation in the clone of the cells take place in order to increase the affinity of the antigen.

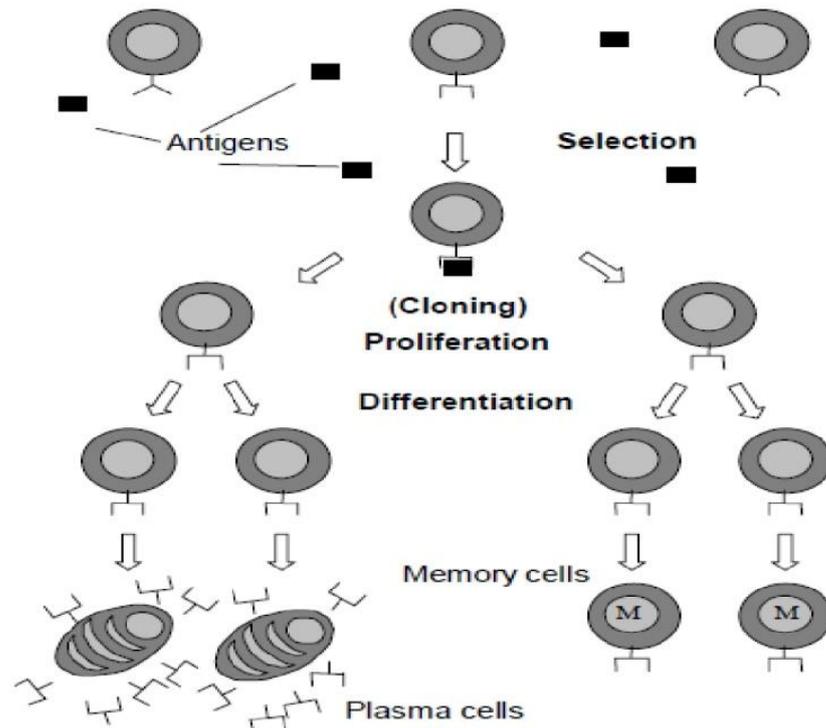


Figure 2. Clonal Selection Principle

5. PROPOSED WORK

In this section we provide the overall architecture of our solution and provide more detail on the central component and thus we discuss how the solutions exactly address the problems above.

5.1. Overall Architecture

The overall layered cloud architecture of our solution is illustrated in Figure 3. The terminal at the end user side can be a simple computer running with a minimal version of any open source operating system. The user would be having an access to the security services running in the security server, which in turn having access to a private cloud. The security server is an intermediate machine which is powered with the best intrusion detection system Snort, IPCop Firewall and a security validator and is availed to the user as software as a service. The central idea of presence of open sourced IDS and firewall is to reduce workload and unauthorised access to the cloud with a minimal cost. The cloud is powered with only open source software instead of proprietary alternatives except for few which don't have an open source alternative. The end – user terminal on successful validation at the security server will be connected to the cloud through a secure tunnel. All the operations are performed over the secure connection services.

5.1.1 Operation

The internal operation can be seen in Figure 4. In the proposed architecture all the office terminals would be provided with a unique user-id and password. The office user request for an access to the security service running in the security server with HTTPS or SSL protocol through their web browser. The firewall service running here checks for an authorised IP and then validates the particular terminal by means of the security validator. In case there is any terminal who is trying to login but failed thrice, will be redirected to the intrusion detection service, which in turn will try to deal with the unauthorised access and threat management. Once the user qualifies the tests conducted by the security services then they will get an access to the cloud through a secure tunnel service. Flash disk and other media devices can be mounted from any terminal at any point of time. The media so connected will be restricted to the local terminal itself. No other terminal can access the same device. When a terminal wanted to save a file over the cloud then it will be processed meticulously at the anti-virus services.

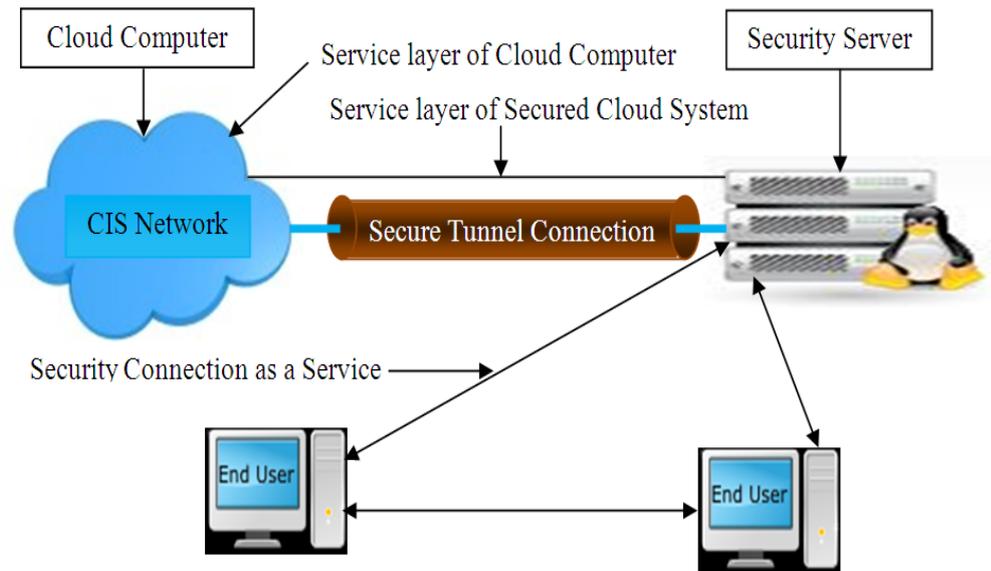


Figure 3. An overview of the proposed architecture

5.2. The end – user Terminal

The end – user terminal is a small computer which will be located at the users for e-Governance operations. The terminals should have their own set of hardware infrastructure. We recommend the offices to use minimum processing power and minimum RAM. The terminals at least must have standard configuration. The terminal can contain a minimal open source operating system like Tiny Core Linux, Damn Small Linux or Puppy Linux.

5.3. The Security Server

A Security Server is a special system that would be working as a security check post for any cloud. The hardware configuration depends upon the number of clients or terminals that would be connecting simultaneously. This server will be connected to the private cloud through a secure tunnel service. This is responsible for providing various security solutions such as firewall, intrusion detection, anti – virus etc. as services to the user. The server would be running with the software such as NetBSD, OpenBSD as an operating system, Snort for intrusion detection system, IPCop as a firewall and Clam for anti-virus.

5.4. The Cloud Computer

A cloud computer is a high end device for general but multipurpose operations. This comprises of hardware and/or software products that are specifically designed for the delivery of services [14]. This is considered to be the most important portion of the entire model. This can be treated as a supercomputer that would be capable enough to take enormous work load. The hardware specification for this depends upon the various factors which are beyond the scope of the paper to be discussed. The software to be used at the cloud computer is given in Table 1.

The cloud delivers all the solutions along with its three services IaaS, PaaS and SaaS which we have already discussed. To have an optimistic computing we propose to use open source software over the server to maintain the robustness of model in terms of availability of softwares and easily affordability.

In the beginning we can use EyeOS which is a disruptive desktop entirely usable from a web browser, which includes a office suite and some collaboration application like moodle, as well as a full framework to develop new web based applications behaving as if they were desktop applications. Because these are free and open source software so there would be no problem by putting these in the proposed cloud and keeping all the data under the cloud control. EyeOS is not only a web desktop with its own valuable applications; it has been designed from the beginning to enable easy development and creation of new applications. EyeOS 2.0 is the perfect development framework for quick and easy creation of rich internet applications. It has been completely developed with open technology and widely accepted standards such as

PHP, MySQL, JavaScript, Qooxdoo, log4php, PHPUnit, OpenOffice and others, enabling the system to function on a common web server without any modifications, and which can easily used by a standard

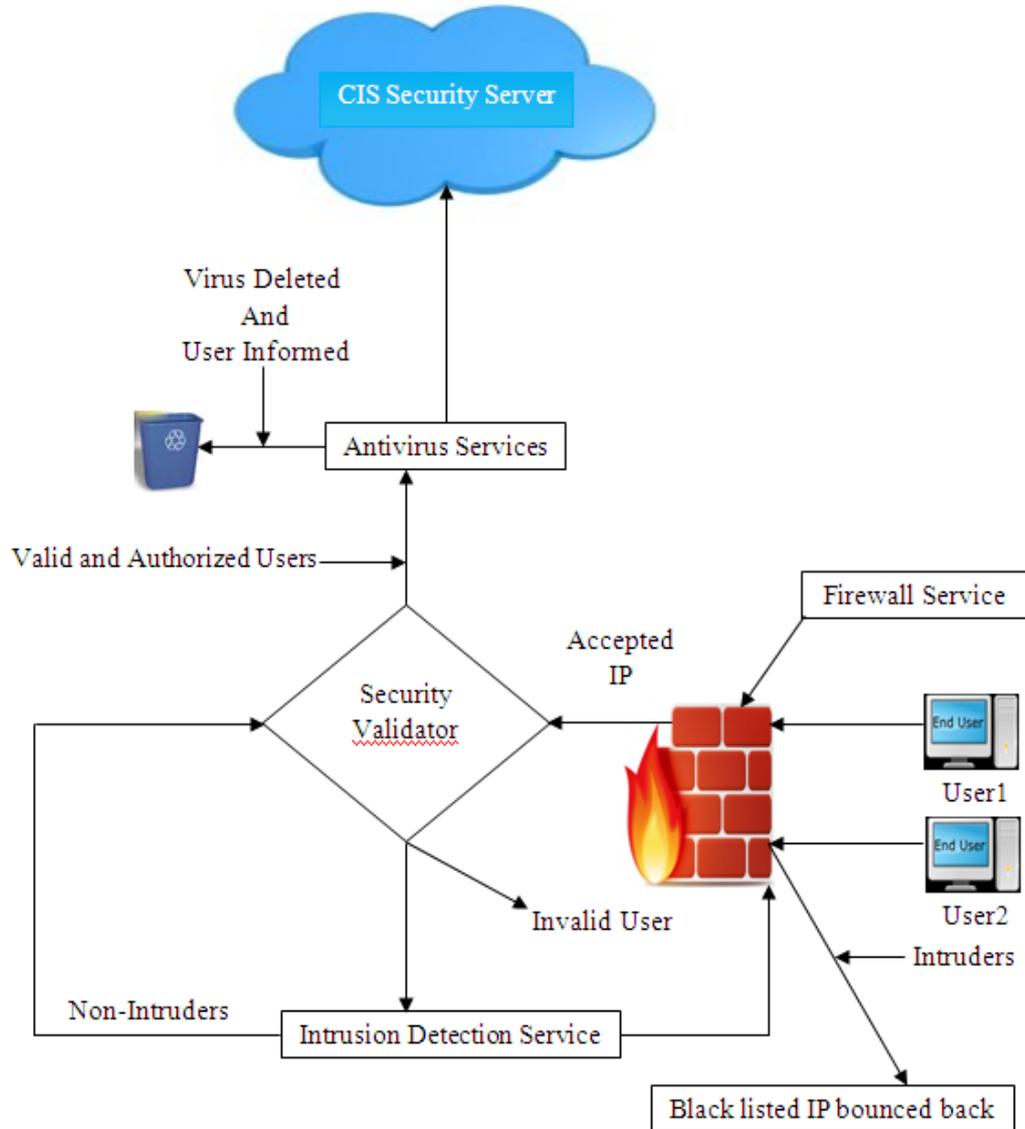


Figure 4. Internal Mechanism

browser without any additional plug-ins installed. However, if open source alternative for a particular purpose like GIS is unavailable, then under that situation the software needs to be purchased in one investment only.

Table 1. Software to be used at the Cloud Computer

Software	Open Source Tools to be used
Operating System	EyeOS , NetBSD , OpenBSD
Virtualization Suite	Sun Virtual Box OSE
Office Suite	OpenOffice
E-Learning Suite	Moodle
Many more...	

5.5. CLONA Implication

Immune response to an antigenic stimulus is the basic deal of clonal selection algorithm used by AIS [15] [16]. The proposed work says how to increase the affinity values of the selected cells using clonal selection algorithm. Selected cells affinity growth process indicates the increased affinity of the selective antigens. The operator known as distance operator used to calculate the affinity values in between the antibody and the antigen.

How the basic cloud security architecture parameters considered in CLONA to create Secure Cloud Immunology (SCI) is given in Figure 5. The symbolic representation of SCI has shown in Figure 6.

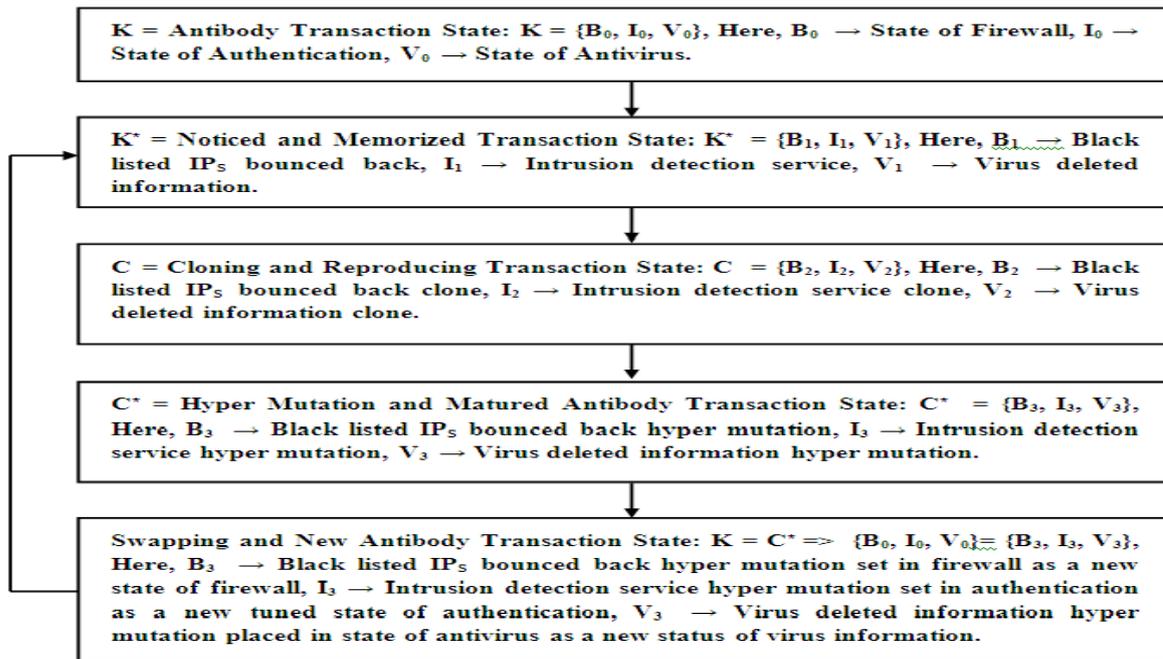


Figure 5. Flow Chart for Cloud Parameters in CLONA – SCI

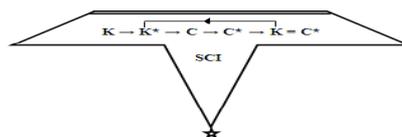


Figure 6. Symbolic Representation of SCI

5.5.1 Hamming distance

The use of Hamming distance [17] is to find out the two binary strings best matching position and follows cyclic shift operation. The maximum hamming distance calculated as follows [18]:

$$HM(x_i^t, x_j^f) = \max\{HD(x_i^t, x_j^f)\}$$

where $HM(x_i^t, x_j^f)$ and $HD(x_i^t, x_j^f)$ are the max hamming and hamming distance between x_i^t and x_j^f respectively.

5.5.2 Shift Continuous Bit Distance

Shift operation operand represented as bit string. In binary matching process shift continuous bit distance [19] [20] is used. Certain positions commonly needed the adjustment. The bit shift operation takes place in both the right and left direction.

5.5.3 Affinity

An antibody has a limited detection space where as antigen has a closer distance to it than the antibody which is based on immune theory. The affinity values determine the danger level of the file. Higher the value of affinity more likely virus are present in the file. Flow chart for finding the affinity value of a file to find out the virus affectedness is shown in Figure 7.

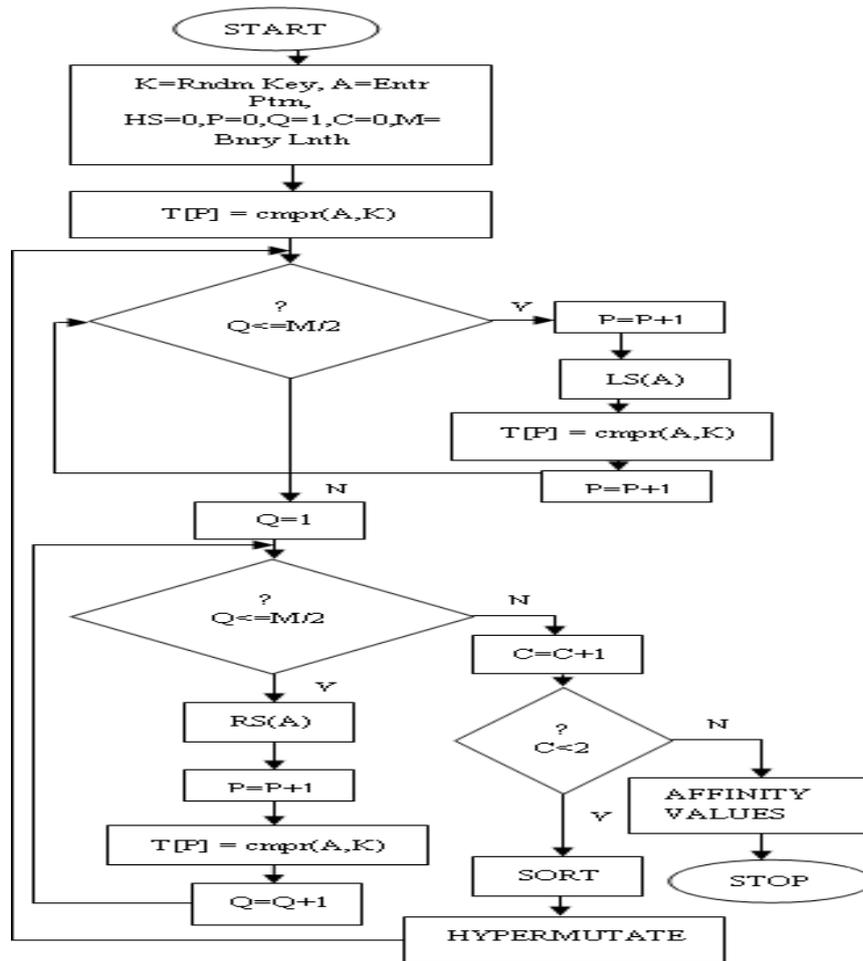


Figure 7. Flow Chart for determining the affinity value of a file

The algorithm involves choosing a selected set of antibodies for cloning and development, then selecting a single best grown antibody for placement in the memory population.

In order to calculate the affinity values between the input antigen values and the randomly generated antibodies we have used the hamming distance and shifting operation. Hamming-max distance can avoid the influence of bits mismatching to enhance the ability of matching. Hamming distance along with shifting operation can get best hamming distance values i.e. max hamming distance. At first the anti-clones values are derived from the antibody values which is randomly generated.

A hamming distance value is found out by comparing the antigen values and the randomly generated antibody, then shifting of the bits takes place in the antigen values. At first left shift operation takes place, shifting a single bit of binary bit in the antigen value and comparing it with the randomly generated antibody, by which we will be getting another hamming distance values.

By repeating the left shift operation $M/2$ times, then right shift operation of the antigen values takes place in the similar way $M/2$ times. After which a number of hamming distance value is found then sorting the distance found by hamming function into descending order. It orders the anti-clone values. Then hyper mutating the anti-clones values i.e. 1's compliment of the anti clones values 0 converted to 1 and 1 converted to 0. Again applying the shift operation on the hyper mutates values and finding the hamming distance. Lastly we get the affinity values of the selected antigen.

5.5.3 Proposed Algorithm

Step 1: Enter the file sample

Step 2: Acquire the antibody Value

Step 3: Derive the Anti-clone value from the antibody

```

Begin
HS=0
Value copied from T[P] to D[P]
For (P = 0; P<L; P++)
    If (T[P] =B[P])
        Increment HS
    End if
End for
Store the value of HS value in DS [0]
Loop
    Until shift < M
    If shift < =M/2
        For (P = 0; P<L; P++)
            LS T[P] and store in C[P]
        End for
    End if
    Else
        For(P=L;P>0;P--)
            Right shift D[P] and store in C[P]
        End for
    For (P=0;P<L;P++)
        If (C[P]==B[P])
            HS = HS +1
        End if
    End for
    FS[shift]=HS
End loop
Find max value FS
Return S_max
End

```

Step 4: Sort in descending order the distance found by hamming function to orders the anti-clone values.

Step 5: compliment the anti-clone value from 1 to 0 or vice versa using the Hyper mutate function.

Step 6: The comparison of the distance of the antibody value and anti clone values changes the antibody values by repeating the 3rd step.

Step 7: Calculated affinity value.

For eg: Let A[P] and B[P] be two binary bits. The number of shifting $M=5$, so the left shifting has done 2 times and right shifting has done 2 times.

A[P] = Antibody

$B[P] = \text{Antigen}$

$A[P]$	0	0	1	1	0	0	1	0
$B[P]$	1	0	0	0	1	1	0	1

Figure 8. Initial state of Antibody and Antigen

Without shifting hamming distance between $A[P]$ and $B[P]$ is 1. Now shifting each bit and then the hamming distance values is calculated. Left shift $B[P]$ and the hamming distance values are:

$B_{L1}[P]$	0	0	0	1	1	0	1	1
$B_{L2}[P]$	0	0	1	1	0	1	1	0

Figure 9. Left shift bit operation

The hamming distance value between $A[P]$ and $BL1[P]$ is 5 and $A[P]$ and $BL2[P]$ is 7. Right shifting $B[P]$ and the hamming distance values are:

$B_{R1}[P]$	1	1	0	0	0	1	1	0
$B_{R2}[P]$	0	1	1	0	0	0	1	1

Figure 10. Right shift bit operation

Hamming Distance value between $A[P]$ and $BR1[P]$ is 3 and $BR2[P]$ is 4. Now hyper mutating the shifting value and then finding the affinity value.

$B_{LH1}[P]$	1	1	1	0	0	1	0	0
$B_{LH2}[P]$	1	1	0	0	1	0	0	1

Figure 11. End Left shift bit operation

Hamming Distance value between $A[P]$ and $BLH1[P]$ is 3 and $A[P]$ and $BLH2[P]$ is 1. The max hamming distance after hyper mutation is 5.

$B_{RH1}[P]$	0	0	1	1	1	0	0	1
$B_{RH2}[t]$	1	0	0	1	1	1	0	0

Figure 12. End Right shift bit operation and hyper mutation

6. PROPOSED ARCHITECTURE

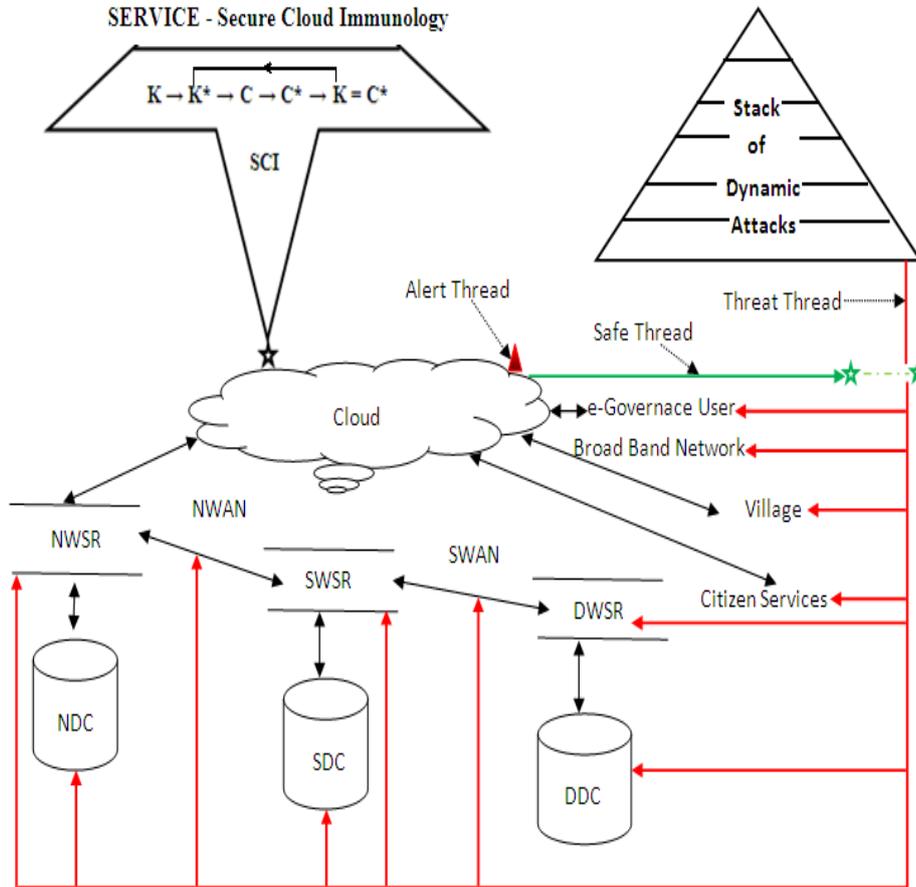


Figure 13. Layout of e-Governance Secure Cloud Architecture

The outer layout of simplified e-Governance secure cloud architecture is shown in Figure 13. It consists of interconnected data centers, web servers, and broad band networks. It spread over the distributed geographical region such as federal, provincial and local area. The different identified interacting, collaborating and shared setup points of the e-Governance cloud system in India are National Data Centre (NDC), State Data Centre (SDC), District Data Centre (DDC), National Level Web Service Repository (NWSR), State Level Web Service Repository (SWSR), District Level Web Service Repository (DWSR), National Wide Area Network (NWAN) and State Wide Area Network (SWAN).

e-Governance cloud system though not free from varied stack of dynamic attacks, either that may be from malware, DoS, DDoS, Shady Rate or from any virus. Here the service SCI through its internal logic provides a secure system. The secure system state is nothing but event based iterative consequence of respective threads. Following three threads hold by the cloud system to get the fellow filling of secure cloud in e-Governance services such as: (i) Threat thread (ii) Alert thread (iii) Safe thread.

Threat Thread: It is represented in figure as a red colored line. It may affect any part of the designed cloud system and at any of the end that may be data center, networks, servers and user end. Though this geographically distributed system extended from National, state, district, block and to remote end users. The threat thread may activate any point of these multilevel to shock and hang the cloud system resources. **Alert Thread:** It is represented in figure as a red colored lamp. It is the sequential consequence of threat thread. It is the kind of thread which generates an alert inside the cloud system when any part of the cloud system comes under any kind of invaders attack.

Safe Thread: It is represented in figure as a green colored line. It brings the SCI logic as a sequential consequence of alert thread. It protects the whole cloud system and brings the state of cloud system into secure safe until another new consequence of threat and alert followed.

7. CONCLUSION

E-governance services in India normally use private cloud, all the cloud services execution is protected under a firewall as a first line of defense. The second line of defense is through identification and authentication of e-Governance cloud users. Our proposed work is using the clonal selection algorithm as the third line of defense for e-Governance applications using cloud. Cloud computing service delivery architecture is loosely coupled and interacting with each other through API and user interfaces for final delivery of services. Generally the intruders attacks try to pass through all these layers and interconnections irrespective of different kind of security. The idea here is to treat all layers and interconnections as cells. Firewalls and user authentications are treated as antibodies, which already has a capability to protect the cloud system. But whenever the attack rate is beyond the firewall, authentication and identification that mean, it makes through the antibody protection, then the third line of defense comes into the picture, to protect the cloud system just as a system call. The dynamic security stack is maintained under this cloud environment to protect the cloud system from any kind of unethical hidden attack by the intruders. The attack could be of any kind as Malware Attack, DoS Attack, Shady Rate Attack, Ethical hacking etc.

Inspired by Clonal Selection Process of AIS we have proposed an affinity maturation process in immune response, which we have presented as an algorithm. The value provided by the function being optimized for a given input may be treated as the affinity to the antigen. Hyper mutation is applied to the anti clone values derived before sorting of the distance found by shifting. Shift operation is again applied to the hyper mutate values and then final affinity values are found out. The purpose of these distance algorithms is to determine an affinity vector. This affinity vector acts a measure of the risk level. A large affinity vector value means that the file is likely at risk, whereas a small affinity vector value means the file is with less risk. The average affinity for a file is calculated to be its danger level. Thus improvement in the affinity vector values increases the probability for the selective antigens.

The said research work may further be enhanced in several ways by analyzing and segregating all cloud layer architecture issues into a 3×3 cubic matrix. In next step it may require to track and trap the issues of each cell of the cubic matrix. Using the saddle point determination method of theory of games we may able to trace each cell. On and above we can think about vaccination technique as well as different other AIS algorithms to improve the security mechanism.

REFERENCES

- [1] Gerges S, Khattab S, Hassan H, Omara FA. Scalable Multi-Tenant Authorization in Highly-Collaborative Cloud Applications. *International Journal of Cloud Computing and Services Science (IJ-CLOSER)*. 2013; 2(2): 106-115.
- [2] Tariq MI. Towards Information Security Metrics Framework for Cloud Computing. *International Journal of Cloud Computing and Services Science (IJ-CLOSER)*. 2012;1(4): 209-217.
- [3] Srinivasan MK, Sarukesi K, Rodrigues P, Sai MM, Revathy P. *State-of-the-art cloud computing security taxonomies: a classification of security challenges in the present cloud computing environment*. 1st International Conference on Advances in Computing, Communications and Informatics (ICACCI). Chennai, India. 2012; 470-476.
- [4] Das RK, Brahma M, Misro AK. *Cloud Computing for Economic Optimization in e-Governance: A Case Study*. 8th International Conference on E-Governance (ICEG). Ahmedabad, India. 2011; 1: 59-71.
- [5] Draman NA, Wilson C, Ling S. Bio-inspired Audio Content-Based Retrieval Framework (B-ACRF). *World Academy of Science, Engineering and Technology*. 2009; 29(5) 785-790.
- [6] Mell P and Grance T. The NIST Definition of Cloud Computing. *National Institute of Standards and Technology*, 2009.
- [7] Kim J and Bentley PJ. *Negative Selection and Niching by an Artificial Immune System for Network Intrusion Detection*. Genetic and Evolutionary Computation Conference (GECCO). 1999; 19– 25.
- [8] Kim J and Bentley PJ. *An Evaluation of Negative Selection in an Artificial Immune System for Network Intrusion Detection*. Genetic and Evolutionary Computation Conference (GECCO). 2001; 1330–1337.
- [9] Burnet FM. Clonal Selection and After. *Theoretical Immunology*. 1978; 63-85.
- [10] Burnet FM. The Clonal Selection Theory of Acquired Immunity. *Australian Journal of Science*. 1959; 67-69.
- [11] Holland JH. *Adaptation in Natural and Artificial Systems*. MIT Press, 5th Ed, 1998.
- [12] Paul WE. Recognition and Response. *Scientific America magazine*. 1991.
- [13] Decastro LN, Vonzuben FJ. *The Clonal Selection Algorithm with Engineering Application*. Genetic and Evolutionary Computation Conference (GECCO), Workshop on Artificial Immune Systems and their Application. 2000; 36-37.
- [14] Cellary W, Strykowski S. *E-Government on Cloud Computing and Service – Oriented Architecture*. (ICEGOV). 2009; 5-10.
- [15] Aickelin U, Greensmith J, Twycross J. *Immune System Approaches to Intrusion Detection—a Review*. 3rd International Conference on Artificial Immune Systems (ICARIS). 2004; 316–329.
- [16] Forrest S, Perelson AS, Allen L, Cherukuri R. *Self-nonsel self discrimination in a computer*. IEEE Symposium on Research in Security and Privacy, Los Alamitos. 1994; 16-17.
- [17] Kephart JO. *A biologically inspired immune system for computers*. Artificial Life IV, MIT Press, Cambridge, MA. 1994; 6–9.

- [18] Chao R and Tan Y. A Virus Detection System Based On Artificial Immune System. *Computational Intelligence and Security*. 2009; 6-10.
- [19] Haeseleer PD. A change-detection algorithm inspired by the immune system: Theory, algorithms and techniques. *Technical Report CS95–The university of New Mexico*. Albuquerque, NM, 1995.
- [20] Helman P, Forrest S. An efficient algorithm for generating random antibody strings. *Technical Report CS-94-07, The University of New Mexico*. Albuquerque, NM, 1994.

BIOGRAPHY OF AUTHORS



Rama Krushna Das is Technical Director (Scientist-E) working with National Informatics Centre (NIC), Department of Electronics and Information Technology, Government of India. His research and professional career spans about twenty five years of coding, research and capacity building in computing, e-governance and related subjects. His expertise is primarily in the domains of Electronic Governance, Implementation Architectures and Strategy, and Software Technology. He is presently involved in development and implementation of different E-Governance projects of NIC. He has published several peer-reviewed papers as journal articles, book chapters, and contributions to conference proceedings. His research interests include e-governance, software engineering, Service Oriented Architecture and Cloud computing. He is a life member of Computer Society of India (CSI) and a professional member of the Association for Computing Machinery (ACM).



Dr. ManasRanjanPatra holds a Ph.D. Degree in Computer Science from the Central University of Hyderabad, India. Currently he is an Associate Professor in the Post Graduate Department of Computer Science, Berhampur University, India. He has about 24 years of experience in teaching and research in different areas of Computer Science. He had visiting assignment to International Institute for Software Technology, Macao as a United Nations Fellow and for sometime worked as assistant professor in the Institute for Development and Research in Banking Technology, Hyderabad. He has about 90 publications to his credit. His research interests include Service Oriented Computing, Software Engineering, Applications of Data mining and e-Governance. He has presented papers, chaired technical sessions and served in the technical committees of many International conferences.



Ajita Kumar Misrois is currently a Ph.D. Research Scholar of PG Department of Computer Science, Berhampur University, Odisha, India. He has achieved his M.Tech.-Computer Science degree from Berhampur University in 2010. His research interest includes Cloud Computing, Software Engineering and Applications of e-Governance.