https://doi.org/10.51483/IJCCR.1.1.2021.81-89 ISSN: 2790-1386

SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Cryptocurrency Research

Publisher's Home Page: https: / / www.svedbergopen.com /

Research Paper Open Access

Cloud Computing Mechanism for Security

Mohammed I. Alghamdi[1*]

[1]Department of Computer Science, Al-Baha University, Al-Baha City, Kingdom of Saudi Arabia. E-mail: mialmushilah@bu.edu.sa

Article Info

Volume 1, Issue 1, December 2021
Received : 21 September 2021
Accepted : 16 November 2021
Published : 05 December 2021
doi: 10.51483/IJCCR.1.1.2021.81-89

Abstract

Recently, cloud computing—which describes the use of a collection of distributed services, applications, information, and infrastructure comprised of pools of computers, networks, information, and storage resources—has grown from being a promising business concept to one of the fast-growing segments of the IT industry. It extends Information Technology's (IT) existing capabilities. But as more and more information on individuals and companies are placed in the cloud, concerns are beginning to grow about just how safe an environment it is. Despite of all the hype surrounding the cloud, enterprise customers are still reluctant to deploy their business in the cloud. However, security has always been seen as the biggest barrier to putting applications in the cloud. Encryption, authentication, and other security issues are real threats when it comes to adopting cloud computing. When we are talking about data security, issues like data breach liability and data privacy are bound to arise. This paper extends the work that has earlier been established (Stephanie *et al.*, 1997; Somayaji and Forrest, 1998; Forrest *et al.*, 2002; Top Threats to Cloud Computing, 2010). Immune inspired approach for securing mobile ad hoc networks is specified there. Although it is clearly indicated there that the research scope is the wireless networks in general and hybrid mobile ad hoc networks in particular, applying the protocol in cloud environment might have effective impact.

Keywords: *Security, Cloud computing, Networks, IT industry*

1. Introduction

Cloud computing generally refers to grid computing, utility computing, software as a service, storage in the cloud and virtualization (Sergio *et al.*, 2000). Improving end-user productivity, improving reliability, increasing security, gaining accessing to more sophisticated applications, and saving energy; these are some of cloud computing benefits. In contrast, a lack of interoperability, application compatibility, difficulty in meeting compliance regulations, and inadequate security are some of some challenges that cloud computing poses. Cloud infrastructure can reside within the company's data enters (as internal clouds or on-premises solutions) or on external cloud computing resources (off-premise solutions available through service providers). It encompasses any subscription-based or pay-peruse service that extends existing IT capabilities. Many challenges are facing cloud computing, the consistent need for backed-up services, one client is allowed to use multiple applications in creating its own, allowing multiple clients to use same

* *Corresponding author: Mohammed I. Alghamdi, Department of Computer Science, Al-Baha University, Al-Baha City, Kingdom of Saudi Arabia. E-mail: mialmushilah@bu.edu.sa*

resource without knowing it which may cause a conflict of interest among those clients, many applications may run on one machine or/and various machines may run one program, Data synchronization, partitioning, distribution, and security, different cloud providers have variety ways to store the vendors' data which may affect the data integrity.

As approved and stated in de Paula *et al.* (2004), there are seven issues that need to be addressed before switching to cloud computing model: privileged user access, regulatory compliance, data location, data segregation, recovery, and long-term viability.

2. Cloud Computing Security

Enterprises across sectors are eager to adopt cloud computing but security is needed both to accelerate cloud adoption on a wide scale and to respond to regulatory drivers (Jeffry *et al.*, 2008).

Cloud computing is shaping the future of IT, but the absence of a compliance environment is having dramatic impact on cloud computing growth. Organizations using cloud computing as a service infrastructure, critically like to examine the security and confidentiality issues for their business-critical insensitive applications. Yet, guaranteeing the security of corporate data in the''cloud'' is difficult, if not impossible, as they provide different services like SaaS, PaaS, and IaaS. Each service has its own security issues (Sarafijanovic and Le Boudec, 2004).

In cloud SaaS model, it has been stated that most commonly cited reason why enterprises are not interested in SaaS. Consequently, addressing enterprise security concerns has emerged as the biggest challenge for the adoption of SaaS applications in the cloud (Le Boudec and Sarafijanovic, 2003). IaaS only provides basic security (perimeter firewall, load balancing, etc.) and applications moving into the cloud will need higher levels of security provided at the host. On other hand, although PaaS has many advantages to the developers to build their applications, these advantages could be helpful to the hackers to leverage the PaaS cloud infrastructure.

3. Related Works

New technologies usually bring new risks besides the new opportunities. Clouds provide a possibility to re-architect older applications and infrastructure to meet or exceed modern security requirements (Sarafijanovic and Le Boudec, 2005). With the different options for adopting cloud (services, infrastructure, platforms, internal clouds, external clouds, etc.) a single security method cannot handle different types of threats. Moreover, the different ways to store the data implies more challenges for cloud security. The use of thin clients that run with the minimum number of resources and with no stored user data is one of the good ways to maintain data security (Nauman Mazhar and Muddassar Farooq, 2007). The approach protects the passwords and seems solid, however using some protocols for performing some particular tasks has long been rendered as insecure. According to a recent International Distribution and Consulting Inc. (IDCI) survey, 74% of IT executives and CIO's cited security as the top challenge preventing their adoption of the cloud services model (Hofmeyr *et al.*, 1996). There are several research works happening in the area of cloud security. Several groups and Organization is interested in developing security solutions and standards for the cloud. Some solutions have presented a method for strengthen the access policies to the cloud and divide the users' view of one application from the backend information storage. Virtualization, multiple processors, or network adaptors are the means to implement that approach (Jimmy *et al.*, 2007).

In Hofmeyr and Forrest (2000) LDSE with an auditing monitoring control has been developed by IPI lab, University of Southern California. Although the research is not yet implemented, its aim is to establish a new method for assuring image integrity while being transferred from end to end. Few thousands random bits are selected; lossless compression for the 10% of the selected bits is then performed. A random pixel LDSE is then used to embed the image digital signature in US image while LDSERS method is used to embed the digital signature in an MR. a hash value is then computed encrypted and embedded as a digital signature. On another hand, to record and examine the access activities to private data, an audit trial method has been proposed by HIPAA Security Standards (Health Insurance Portability and Accountability Act (HIPAA)). The system as stated is able to audit the image data flow and generate the required trials of the image data. Four layers are involved in the system: Action, notification, Audit, and record layers. The project validation and evaluation are still under process. The best security solution for web applications is to develop a development framework that has tough security architecture. Tsai *et al.* (2009), put forth a four-tier framework for web-based development that though seems interesting, only implies a security facet in the process (Zheng and Jian, 2006). It has been stated in the abovementioned project that the Internet ware presents many new issues not faced by traditional software development, the new issues mainly from the open, dynamic, and distributed nature of the Internet environment. While many new techniques, such as, ontology, SOC, environment modeling, social ranking, cloud computing, and adaptive control mechanisms, are being developed to address these issues, these new techniques still need to be evaluated in the Internet environment to demonstrate their feasibility and

effectiveness. One of the main issues to be addressed in these new techniques is the security that represents the safety valve for many services and applications. In Anil (2007) the research draws a road map toward cloud-centric development and theX10 language is one way to achieve better use of cloud capabilities of massive parallel processing and concurrency (Mohamed and Abdullah, 1985). As stated, a robust set of policies and protocols are required to help secure transmission of data within the cloud. Concerns regarding intrusion of data by external nonusers of the cloud through the internet should also be considered. Measures should be set in place to make the cloud environment secure, private and isolated in the Internet to avoid cyber criminals attacking the cloud. That is what we are trying to achieve in our research. In Steven (1999) the researcher points out the value of filtering a packet-sniffer output to specific services as an effective way to address security issue shown by anomalous packets directed to specific ports or services. An intrusion detection system that uses statistical anomaly detection to find remote to local attacks targeted at essential network has been presented. The system uses service-based approach that uses statistical data for requests to different services to improve the system capability. More improvement is needed for the system to be reliable within the cloud. Raj *et al*. (2009) suggest resource isolation to ensure security of data during processing, by isolating the process or caches in virtual machines, and isolating those virtual caches from the hypervisor cache. Two approaches to provide cache-based security and performance isolation—cache hierarchy aware core assignment, and page-coloring based cache partitioning have been present. The approaches are effective in providing required isolation properties. Isolation for avoiding the impacts of confidentiality and integrity is considered to be addressed later. Hayes (2008) points out that there is no way to know if the cloud providers properly deleted a client's purged data, or whether they saved it for some unknown reason. As stated, the third-party service is presumably less likely to bear all the losses. In some circumstances the tenant might not even be informed that his documents have been released. It seems likely that much of the world's digital information will be living in the clouds long before such questions are resolved. A tough security framework yet needs to be developed.

Basta and Halton (2007) suggest one way to avoid IP spoofing by using encrypted protocols wherever possible. They also suggest avoiding ARP poisoning by requiring root access to change ARP tables; using static, rather than dynamic ARP tables; or at least make sure changes to the ARP tables are Logged. As pointed out, the largest gaps between cloud security practice and cloud security research lies in the fact that the assumptions in the research leave out some very important differences between cloud security and virtual machine security, in our research we are trying to bridge this gap.

4. Methods

4.1. The Delivery Models

Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) form the core of the cloud and they exhibit certain characteristics like on-demand self-service, multi-tenancy, ubiquitous network, measured service and rapid elasticity which are shown in the top layer. These fundamental elements of the cloud require security which depends on and varies with respect to the deployment model that is used, the way by which it is delivered and the character it exhibits. Some of the fundamental security challenges are data storage security, data transmission security, application security and security related to third party resources.

In our research, we are targeting an integrated security model targeting different levels of security of data for a typical cloud infrastructure. Our research questions will center on data integrity and data security over the cloud, and we intend to develop a framework by which the security methodology varies dynamically from one transaction/ communication to another. The primary methods for achieving the goals and objectives of the project and as illustrated in the proposed architecture (Figure 1) will be:

a. To design a security protocol that has different security levels for assuring end-to-end data security, there will be an extra tag with some fields that have functions for the integrity verification within the payload data that is attached to the protocol as depicted in Figure 1.

b. One of the Tag's field returns to the customer the number and details of the intermediate servers needed to reach the final location where the customer's data is located. This will help the customers to determine whether their data is secured enough.

c. The Tag contains a data for testing locations' hostility where the customer's data is mirrored and the different paths for reaching these locations. Our methodology here is to locate some data that is respond to tamper by some indicators, the type of tamper and details of the source should be reported to the tenant when the Tag returned back to him (Figure 2). This will be specified later.

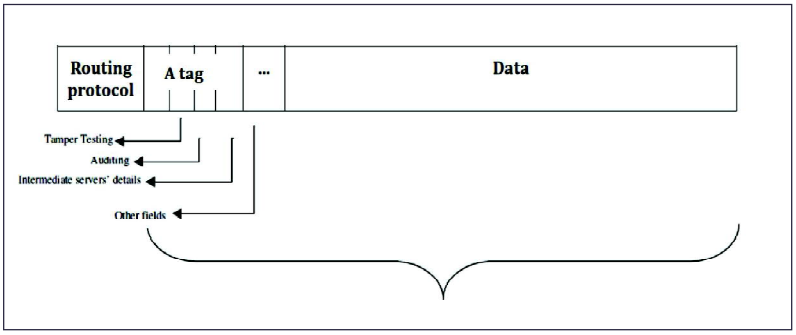


Figure 1: A Protocol with Extra Tag

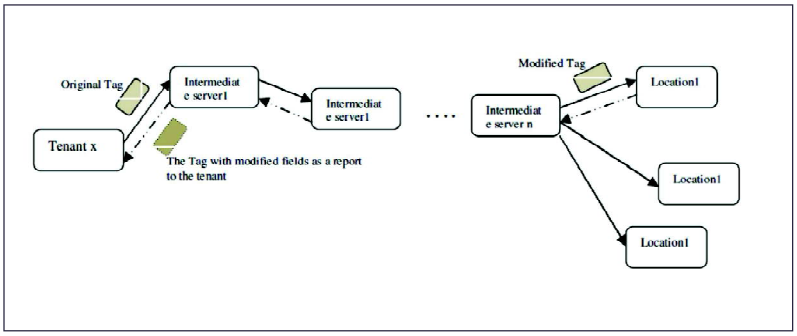


Figure 2: Tracing Trust Within Cloud

d. Moreover, we aim to deploy a cryptography method where applicable to ensure the data will not be revealed while transmitting through the intermediate locations. A survey will be made to find out the appropriate cryptography method to be deployed within the (T2S) protocol.

e. An extra mechanism in the T2S is an auditing mechanism that records the provider's details (including time, administrators' details, maintenance details, etc.).

f. Another mechanism to be used with (T2SP) is the Immune inspired protocol where applicable (particularly within the authentication points in the cloud in Figure 3) to more securing the communications among the different parties within the cloud. This is to be specified later.

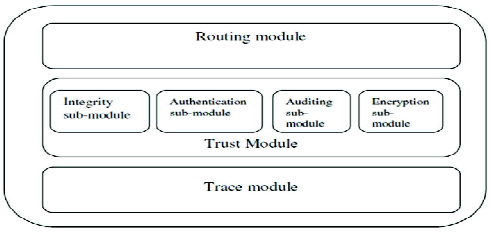


Figure 3: T²SP Sub and Main Modules

4.2. Protocol Description

As a preliminary description of the T2SP; there are three main modules for T2SP that handle the security, privacy and authenticity within the cloud as illustrated in Figure 3:

4.2.1. Routing Module

It handles the interoperability with the existing routing protocol that is shoulder Figure 3. T2SP sub and main modules the responsibility of delivering the data from source to destination. The significance of this module appears as the new other purposes protocols need to be specified in the routing ones and to define the fields in which those protocols could be taken care of.

4.2.2. Trust Module

This module handles the privacy, data security and authenticity parts. It has four sub-modules that can be defined as follows:

4.2.3. Integrity Sub-Module

The function of this sub-module is to ensure the integrity within the data that is being sent from the tenant to the provider. The tamper's details will be reported to the tenant upon receiving back the protocol report.

4.2.4. Authentication Sub-Module

The tenant should have a list of all the provider's authorized personnel those may have any services to do with tenant's data. This sub-module will ensure the right deployment of the Service Level Agreement (SLAs) in terms of authenticity.

a. Auditing Sub-Module: This sub-module records the different processes made by the third parties and returns back a report to the tenant. This trace helps the tenant to decide the third parties' trust.

b. Encryption Sub-Module: An encryption mechanism will be deployed in this sub-module to ensure the modules concealment; this will provide a protection to the protocol from being hacked.

c. Trace Module: This module is responsible of tracing the tenant's data while rerouted within the cloud from the source to the final destination. Being acquainted with locations needed to reach your data or where your data is located will help to analyze and evaluate the threats and risks you need to protect your data against.

4.3. Security Algorithms

The two main algorithms to be specified in our research are:

a. Trace Algorithm: It specifies how the protocol can trace the data while rerouted through different networks ensuring the privacy and data confidentiality.

b. Trust Algorithm: It specifies whether the tenant can trust the third parties and the providers that are hosting the tenant's assets.

4.4. Security Management Part

In this project, the following issues will be considered:

a. A method for isolating customer's data networks should be implemented.

b. There should be a technique to secure the customer's access to the cloud-based resources.

c. Consistent backup for the cloud-based resources.

d. Applying access control and auditing mechanisms.

e. There should be a library and up-to-date OS and applications.

f. A mechanism for preventing denial of services.

g. Data encryption method should be applied.

h. There should be identity manager.

5. Results and Discussion

Conducting a test in three nodes (one mobile and two stationary wireless nodes) for 15 hours it has been found that 1,306,197 patterns had been captured where 5,711 detectors successfully passed the test to be valid detectors, i.e., for

every 238 patterns there will be one valid detector. These detectors generation complexity and the storage required space.

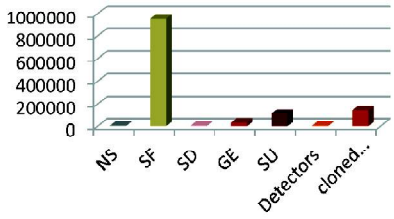


Figure 4: Incoming Patterns Classification and the Detectors Profiles

```
mysql> select * from detector_profile where detector_id = '2694';
+-------------+----------+---------------------+
| DETECTOR_ID | DETECTOR | DETECTOR_CREATED_TIME |
+-------------+----------+---------------------+
|        2694 | eFB3jA   |  2009-09-23 15:54:48 |
+-------------+----------+---------------------+
1 row in set (0.06 sec)

mysql>
```

Figure 5: High Scored Detector to be Cloned

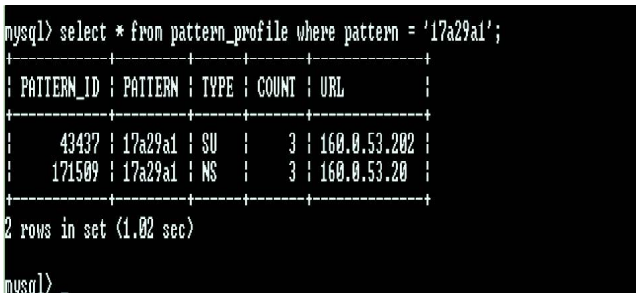
During the test period it has been found that the genes patterns which are frequently occurring patterns (GE), self from detectors (SD) patterns, Self-patterns, Suspect patterns, and non-self-patterns were 32,283; 6,821; 954,412;111,795; and 5,518 respectively. The five data types are shown in Figure 4.

Figure 5 shows how high scored detector is cloned in a new one. Detector eFB3jA shown in Figure 5, has gained a high score in detecting the nonself patterns, hence cloned to a new detector 19cd344 shown in Figure 6.

```
ysql> select * from cloned_detectors_profile where cloned_id = '13680';
+-----------+---------------+-----------------------------+-------------
---------------+----------------------------+
| CLONED_ID | CLONED_DETECTOR | CLONEDDETECTOR_FROM_DETECTOR_ID | CLONEDDETECTOR
MATCHED_GENE_ID | CLONED_DETECTOR_CREATED_TIME |
+-----------+---------------+-----------------------------+-------------
---------------+----------------------------+
|     13680 | 19cd          |                        2694 |
          344 |         2009-09-27 15:16:18 |
+-----------+---------------+-----------------------------+-------------
```

Figure 6: High Scored Detector to be Cloned

On the other hand, the incoming pattern had firstly been classified as a suspect as depicted in Figure 7, then after it caused a change, alteration or a modification in one of the NTFS files in the system for three counts it is then reclassified as nonself.

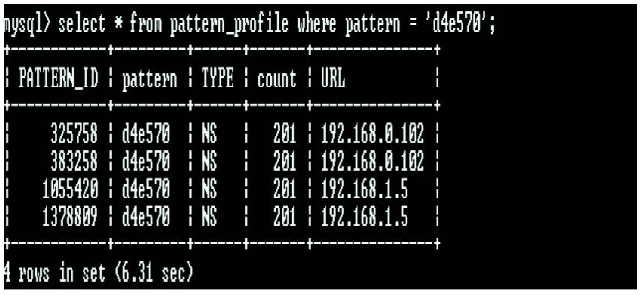


```
mysql> select * from pattern_profile where pattern = '17a29a1';
+------------+---------+------+-------+-------------+
| PATTERN_ID | PATTERN | TYPE | COUNT | URL         |
+------------+---------+------+-------+-------------+
|      43437 | 17a29a1 | SU   |     3 | 160.0.53.202 |
|     171509 | 17a29a1 | NS   |     3 | 160.0.53.20  |
+------------+---------+------+-------+-------------+
2 rows in set (1.02 sec)

mysql>
```

Figure 7: High Scored Detector to be Cloned

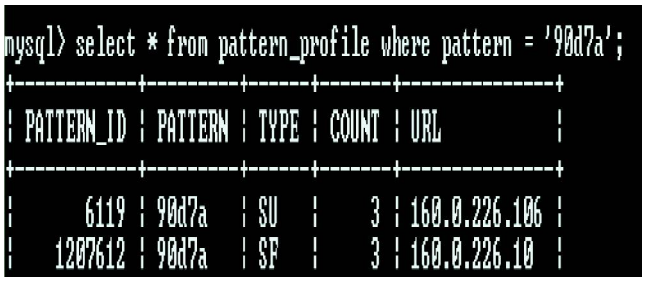
As depicted in Figure 7, regardless of source address, the same incoming pattern will be captured, e.g., pattern 17a29a1 had been classified during the first three counts as a suspect pattern where it has been assigned id 43437 sent from 160.0.53.202, the same pattern is captured again with different id 171509 as a non-self-sent from 160.0.53.20 when a change in one of the system files has been discovered. The pattern that is classified as a non self will be broadcasted to the rest of the nodes so as to be treated properly as a non self in the future. Figure 8 depicts that a same pattern is classified by different nodes as a non self.



```
mysql> select * from pattern_profile where pattern = 'd4e570';
+------------+---------+------+-------+--------------+
| PATTERN_ID | pattern | TYPE | count | URL          |
+------------+---------+------+-------+--------------+
|     325758 | d4e570  | NS   |   201 | 192.168.0.102 |
|     383258 | d4e570  | NS   |   201 | 192.168.0.102 |
|    1055420 | d4e570  | NS   |   201 | 192.168.1.5   |
|    1378809 | d4e570  | NS   |   201 | 192.168.1.5   |
+------------+---------+------+-------+--------------+
4 rows in set (6.31 sec)
```

Figure 8: A Same Pattern will have the Same Classification by Different Nodes

The entire incoming patterns that have not yet been classified will be stored as suspect patterns (SU) as shown in Figure 9. If it has been encountered more than three times and at the same time it doesn't affect in system harm, it is then considered as a self as shown in Figure 9.



```
mysql> select * from pattern_profile where pattern = '90d7a';
+------------+---------+------+-------+--------------+
| PATTERN_ID | PATTERN | TYPE | COUNT | URL          |
+------------+---------+------+-------+--------------+
|       6119 | 90d7a   | SU   |     3 | 160.0.226.106 |
|    1207612 | 90d7a   | SF   |     3 | 160.0.226.10  |
+------------+---------+------+-------+--------------+
```

Figure 9: Suspect Patterns

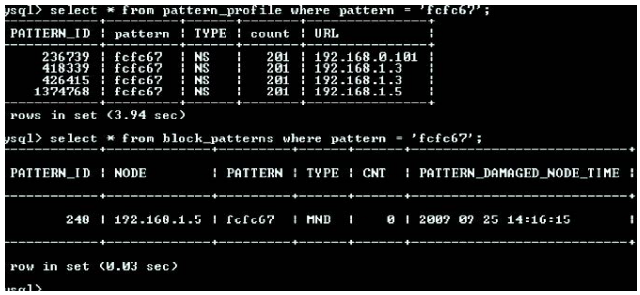


Figure 10: Block Patterns

As considered, the corrupt patterns that harm the NTSF files are classified as non self, and it will be blocked. In Figure 10 a pattern fcfc67 has been encountered as non self with three different pattern_id incoming from three different nodes; a search in the block patterns table shows that the same pattern is then blocked.

6. Conclusion and Future Work

The success in mapping immune features and properties is expected to contribute to the ad hoc security field with a new auto anomaly detection mechanism, memory mechanism for better future reaction, self-isolation for the malicious nodes resulting in a trusty communication environment, and nodes' survivability. The immune-based security protocol for MANETs has been presented. An intelligent agent that contains three profiles, gene's profile, non-self-profile, and detectors' profile is created. Replicas of the agent are distributed to all nodes inside the domain upon connection establishment. A combination of negative selection, clonal selection, and danger theory mechanisms has been mapped, expecting a self-organized system could be accomplished. The major well-thought-out issues are scalability and the bandwidth conserving, which mainly characterize the ad hoc networks.

On the other hand, security has always been seen as the biggest barrier to putting applications in the cloud. Applying this protocol to securing clouds will have affirmative great impact. The achieved results represent only one component of the protocol: trust module, in the future the other modules will be applied.

References

Aickelin, U., Bentley, P., Cayzer, S., Kim, J., and McLeod, J. (2003). Danger Theory: The Link between AIS and IDS? In Proceedings ICARIS- 2003, 2nd International Conference on Artificial Immune Systems, 147-155.

Anil, Somayaji. (2007). Future of Biologically Inspired Computer Defenses. *Information Security Technical Report, 12*, 228-234.

de Paula, F.S., de Castro, L. N., and de Geus, P. L. (2004). An Intrusion Detection System Using Ideas from the Immune System, 1059-1066.

Forrest, S., Hofmeyr, S., Somayaji, A., and Long Staff, T. (1996). A Sense of Self for UNIX Processes. *In Proceedings of the 1996 IEEE Symposium on Computer Security and Privacy*, IEEE Press.

Forrest, S., Balthrop, J., Glickman, M., and Ackley, D. (2002). Computation in the Wild. *In the Internet as a Large-Complex System*, edited by Park, K., and Willins, W. Oxford University Press.

Hofmeyr, S., and Forrest, S. (2000). Architecture for an Artificial Immune System. *Evolutionary Computation Journal, 8*(4), 443-473.

Jeffry, O. Kephart., Gregory, B. Sorkin., William, C. Arnold., David, M. Chess., Gerald, J. Tesauro., and Steve, R. White. (1995). Biologically Inspired Defense Against Computer Viruses, 985-996.

Jimmy, Mcgibney., Dmitri, Botvich., and Sasiharan, Balasubramanimam. (2007). A Combined Biologically and Socially Inspired Protocol to Mitigating Threads in Mobile Ad hoc Networks.

Le Boudec, J., and Sarafijanovic, S. (2003). An Artificial Immune System Approach to Misbehavior Detection in Mobile Ad-hoc Networks. Technical Report IC/2003/59, École Poly technique Féderalé de Laussane (EPFL).

Mohamed, Y., and Abdullah, A. (2009). Immune Inspired Approach for Securing Wireless Ad hoc Networks. *International Journal of Computer Science and Security (IJCSNS), 9*(7), 206-212.

Nauman, Mazhar., and Muddassar, Farooq. (2007). Bee AIS: Artificial Immune System Security for Nature Inspired, MANET Routing Protocol, Bee Ad Hoc. Artificial Immune Systems. DOI:10.1007/978-3-540-73922-7_32

Sarafijanovic, S., and Le Boudec, J. (2004). An Artificial Immune System for Misbehavior Detection in Mobile Ad-hoc Networks with Virtual Thymus. *Clustering. Danger Signal and Memory Detectors*, 342-356.

Sarafijanovic, S., Le Boudec, J.Y. (September 2005). An Artificial Immune System Approach with Secondary Response for Misbehavior Detection in Mobile Ad hoc Networks. *IEEE Transactions on Neural Networks, 16*(5).

Sergio, Marti, T.J., Giuli, Kevin, Lai., and Mary, Baker. (2000). Mitigating Routing Misbehavior in Mobile Ad Hoc Networks. *In Proceedings of MOBICOM*, 255-265.

Somayaji, A., Hofmeyr, S., and Forrest, S. (1998). Principles of a Computer Immune System. In 1997 New Security Frameworks Workshop, 75-82, ACM.

Stephanie, Forrest., Steven A. Hofmeyr., and Anil, Somayaji. (1997). Computer Immunology. *Communication of the ACM, 40*(10), 88-96.

Steven A. Hofmeyr. (1999). An Immunological Model of Distributed Detection and Its Application to Computer Security. Ph.D. Thesis, University of New Mexico, Albuquerque, NM.

Zheng, You., and Jian, Wang. (2006). *DIMH: A Novel Model to Detect and Isolate Malicious Hosts for Mobile Ad hoc Networks*, 660-669, Elsevier.

Cite this article as: Mohammed I. Alghamdi (2021). Cloud Computing Mechanism for Security. *International Journal of Cryptocurrency Research,* 1(1), 81-89. doi: 10.51483/IJCCR.1.1.2021.81-89.