ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**

# Client and Server-side Attacks Detection in the Cloud Environment

**K.G. Maheswari**

Assistant Professor (Senior), Department of MCA, Institute of Road and Transport Technology, Anna University, Erode, Tamil Nadu, India

**Abstract:** The rapid development in IT (Information Technology) has needed of security in multiple levels. The latest technology developments are integrated with all technology like grid, web and cloud computing. An IDS (Intrusion Detection System) is an effective approach to detect and prevent the intruder in network, application and in the system level. In cloud environment, the dynamic input manipulation is possible in the application level and the infrastructure level. In application level, consists of SQL injections, XSS attacks and the Spoofing attack etc. The infrastructure level malware injections and co-resident attack are possible. In this paper, cloud hybrid detection algorithm is proposed to detect dynamic input manipulation attack. The performance of proposed algorithm provides better efficiency detection of cloud manipulation.

**Keywords**: Cloud attacks, Malware injection, co-resident attack, Worms

## I. INTRODUCTION

The latest technology of computer and information technology in cloud computing, which is the integration of many existing technology like grid computing, autonomic computing etc. It provides the service to the cloud user on demand basics. When an organization migrates to cloud services, particularly public cloud service, most of the computing system may be in the control of third party CSP. So a lot of security challengers in cloud computing. Like web application attacks, DOS, DDos, manipulation attacks etc are also possible in cloud computing environment.

Mell et al[1] malware is defined as it is a software program which is injected into the system with intent of compromising (CAI paradigms) victim's applications OS or data. Based on the functionality of malware, it is classified as virus, trojans, worms, backdoors and rootkits and spyware. In Table 1.1 shows that list of malware properties.

**1.1 List of Some Malware Properties**

| Name | Functionality | Host Required | Damaged Level | Example |
|---|---|---|---|---|
| Virus | Self replication | yes | Low | Autorun.abt, CIH |
| Worms | Exploits and effect the network | no | High | Code red, Netsky |
| Back door | This type of malware used to Listens on ports and gain access | no | Medium | Hupigon ,Rexob |

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**

| | | | | |
|---|---|---|---|---|
| | through them later | | | |
| Trojan | This type of malware that spoofs a harmless or useful program. but, actually stores other malware. | yes | Medium | Limbo/NetHell, Pidief |
| Spyware | This type of malware used to spy on victim's activities and also used to steal sensitive information. | no | Medium | SecurityToolbar, Virtumonde |
| Rootkit | This type of malware alters the operating system memory data structure and hide themself. | no | High | Rustock, Mebroot |

**Cloud malware detection essential**

- In Cloud environment, Malware propagation is very fast compare web environment.
- Malware propagation can open the door for even more serious threats
- Cloud Malware can lead to data breaches

**Cloud security frame work layers**

CSA (Cloud Security Alliance) provides the frame work layer for cloud security and Figure 1.2 shows the cloud frame work layers. They are

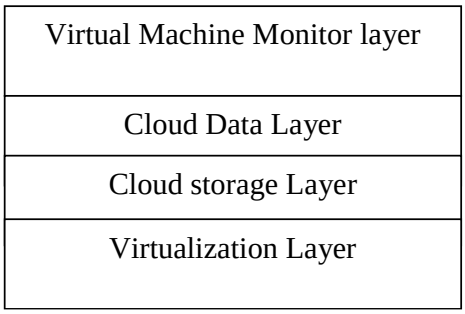



Figure 1.2 Cloud Frame Work Layers

**Virtualization Layer**

This layer is in the cloud infrastructure (IaaS) service model. In this model more than one VM is created by CSP. Hypervisor also enclosed in this layer to maintain the performance of VM.

**Cloud storage layer**

In this layer, it provides flexible data storage and execute dynamically.

**Cloud data layer**

It is the very important layer it is used to store the data and also update the data dynamically.

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**

**Virtual Machine Monitor layer**

It is the top most layer of cloud security frame work. It also provides end to end communication for highly virtualized infrastructures. There are different types of virtualization platforms are available such as VMware, Xen and KVM. This layer is used to monitor the response time and also validating the network components.

**CLOUD BASED MANIPULATION ATTACKS**

In cloud environment, they are two attacks which are manipulated by the attackers. They are

1. Cloud malware injection attack

Attacker can inject the malicious code in application and perform application level attacks like SQL injection attack, XSS attacks etc. In infrastructure level, they can inject malware into VM and perform the malicious activity. This type of attacks are called cloud malware injection attack.

2. Co-resident attack

In Cloud environment, the multiple number of VMs are on a single CPU. There may be a chance to the unknown and untrusted users are in the same location or co–resident. This leads security problem unless perfect isolation by hypervisor. Suppose hypervisor is compromised. The attacker can perform malicious activities and affect the data integrity and confidentiality. This type of attack is called as co-resident attack.

## II. RELATED WORKS

There are many researchers discussing about different types of detection method and technique for Cloud malware Injection attacks, Nancy et al [2], Marnerides, , Bodkhe,PAP,[3] Saeed, IA, et al.[4], Singh, MA&Shrivastava, DM [5],Deshmukh, P&Agarkar, [6] Gamayunov et al. [7] proposed malware detection in infrastructure level for that they proposed one class SVM at the system level and network level node. The proposed cloud resilence architecture for detection of malware, they got 90% detection accuracy.

Sagar,V & Kumar, VR [8] different types of service and mobile cloud infrastructure and also proposed network features to identify the malwares in mobile cloud infrastructure. For the malware detection, they used RF machine learning Algorithm. They detect the malware using abnormal behaviour. Masud, MM et al. [9] have proposed two classifier multipartition, multichunk ensembles, these two classifier are used to reduced classification error and they proposed data stream to get the features because there is no fixed set. Harrison, K, et al. [11] identifying the symptoms of malicious behaviour as opposed to looking for the malware itself.

Park, J-O [10] propose virtual machine monitor agent based bare metal Hypervisor in order to get better vulnerabilities of bare metal. Hypervisor method among various virtualization technologies of clouds services. The proposed base structure was intended to detect and block various malware codes and viruses by including kernel driver inside agent.

In cloud environment, malicious user can get the sensitive information from other users because of improper isolation of VM. To detect the such a security threat many researcher discussed and proposed new method and techniques related VM co-residency attack detection. Zhang et al. [12] the author proposed the concept of Home Alone, the concept in Home Alone is to invert the usual application of side channels rather than exploiting a side channel attack. Home alone uses a side-channel (in the L2 memory cache) as novel, defensive detection tool.By analyzing cache usage during periods in which “friendly” VMs coordinate to avoid portions of the cache, a tenant using home alone can detect the activity of a co-resident “foe” VM and key technical contributions of home alone include classification techniques to analyze cache usage and guest operating system kernel modifications that minimize the performance impact of friendly VMs sidestepping monitored cache portions. The implementation of

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**

home alone on Xen-PVM requires no modification of existing hypervisors and any special action or cooperation by a cloud provider. Maheswari KG[13] discuss the web based attacks in server side and client side environment. Maheswari KG[14] proposed the web based attacks using the minikowski distance concepts. Maheswari KG[15] proposed security scheme in cloud environment to detect the cloud based attacks in the cloud environment. Nalinipriya G [16,17]discuss the VM allocation policies to allocate the load in the cloud environment and also detect the Co- Resident attacks.

## III. PROPOSED ARCHITECTURE

In cloud computing, malware injection may possible in the network service, storage and application level. In this work, cloud malware injection is detected in the application model and infrastructure model. In application-level malware injections are detected using different types of detection tools and add-ons. The infrastructure level malware injections are detected using memory forensics analysis. Because memory structures are used to extract useful information from the computer's physical memory. The proposed hybrid detection technique provide better detection rate comparing to a single method. The proposed architecture of hybrid detection technique is as shown in Figure 3.1

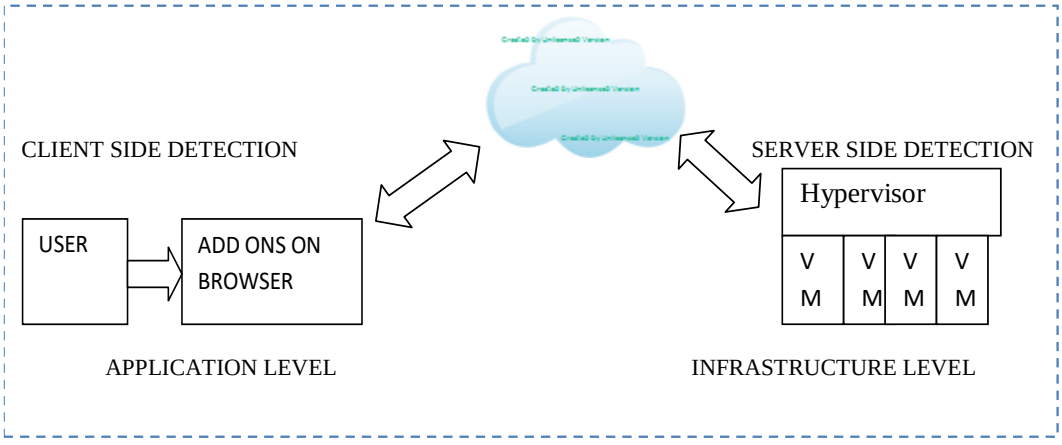



**Figure 3.1 Cloud Hybrid Detection Technique**

### 3.1 Detection of malware injection attacks in application level

Injection attack is one of the category of web application attack. Attacker can exploit vulnerabilities of applications and inject the malicious code to make a desired changes in execution and behavior. The crafted

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**

malicious program, application or virtual machine are injected into the cloud service models like IaaS, SaaS and PaaS. When the malicious injection is completed, attacker can do any data manipulation in storage, service etc. Figure 3.2 shows the client communications with cloud through the browser.

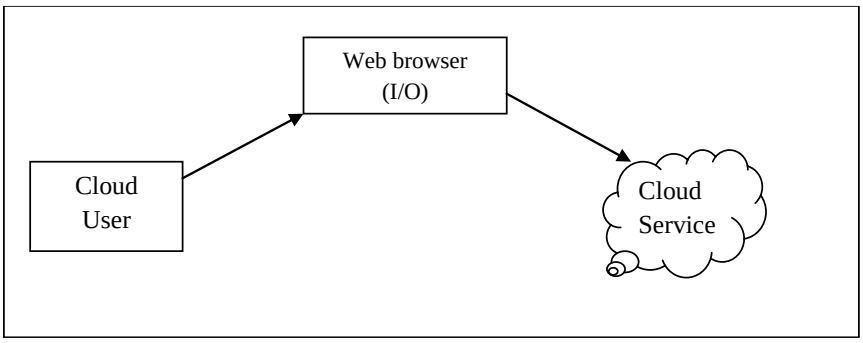



**Figure 1.2 Client Communications with Cloud**

In this thesis work, various types of detection tools are used to detect the different types of SQL injection attacks. Among the all types of detection tools DVMA tool is used to detect the all types of SQL injection attacks.

### 1.2 Detection of Malware Injection Attack in Infrastructure Level

The proposed architecture diagram as shown in Figure 6.10 Client request is send to the CSP. After getting the client request, based on the VM allocation policy, VM will allocate the VM to the server. Network flow monitor is used to monitor and analysis the various network traffic task to detection of malicious activity. In this work, three conditions are proposed to detect the co-residency possibilities in the cloud infrastructure.

**Case 1:**

Least VM and Server Allocation Policy

All VMs are allocated to the different server and within short time, server will not be selected in the second time. So probability of co-resident with another VM, is very less.

**Case 2:**

Most VM and server Allocation Policy

All the subset of the VM may reside on the same servers. So the probability of co-resident with target is more.

**Case 3:**

Randomly VM and server allocation policy:

All the VM are allocated to the different server and there is no particular pattern to allocate VM with server. So the probability of co-resident with target is medium.

To find the co-location of victim VM, the following table 3.3 detection of co-resident VM algorithm proposed and implemented.

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**

INPUT : Give the set of malicious VM
OUT PUT : Find the possible co-resident of target VM
STEP 1 : Client Request send to the CSP
STEP 2 : CSP will allocate VM on server based on allocation policy
STEP 3 : Supose victim VM allocate at time $t_1$
STEP 4: If VM allocate using Most allocation policy at $t_1$ time
Client request allocate to same server and occurrence of co-resident may be high
else
STEP 4 : If VM allocate using Random allocation policy at $t_2$ time
Client request allocate to server randomly and occurrence of co-resident may be Medium
else
STEP 5 : If VM allocate using Least allocation policy at $t_3$time,
Client request allocate to server randomly and occurrence of co-resident may be Low
STEP 6: Repeat the step2

3.3 Detection of co-resident VM algorithm

## IV. RESULT AND DISCUSSIONS

### 4.1 SQL injection attack

Various types of SQL injection detection tools are used to detect different types of SQL injection attacks in the cloud environment. the comparative analysis of the tools are as shown in Table 2.1.

**2.1 Comparison of various tools for detection of SQL injection**

| Attack / Tools | Tautology | Piggy-Back | Illegal Queries | Union Query | Inference Based Attacks | Alternate Encoding | Stored Procedure |
|---|---|---|---|---|---|---|---|
| DVWA | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Tautology Checker | Yes | No | No | No | No | No | No |
| SQLrand | Yes | Yes | No | Yes | Yes | No | No |
| SQLCheck | Yes | Yes | Yes | Yes | Yes | Yes | No |
| SQLGuard | Yes | Yes | Yes | Yes | Yes | Yes | No |

SQL injection tools are compared against all the types of the vulnerabilities such as tautology, union, illegal, piggy back, inference, alternative encoding and stored procedure. Some of the tools are not able to detect all the types of vulnerabilities but DVWA is capable of detecting all the vulnerabilities.The Figure 2.2 shows the vulnerabilities detection rate of various tools. Out of all tools DVWA perform well with the 70 % of detection rate.

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**

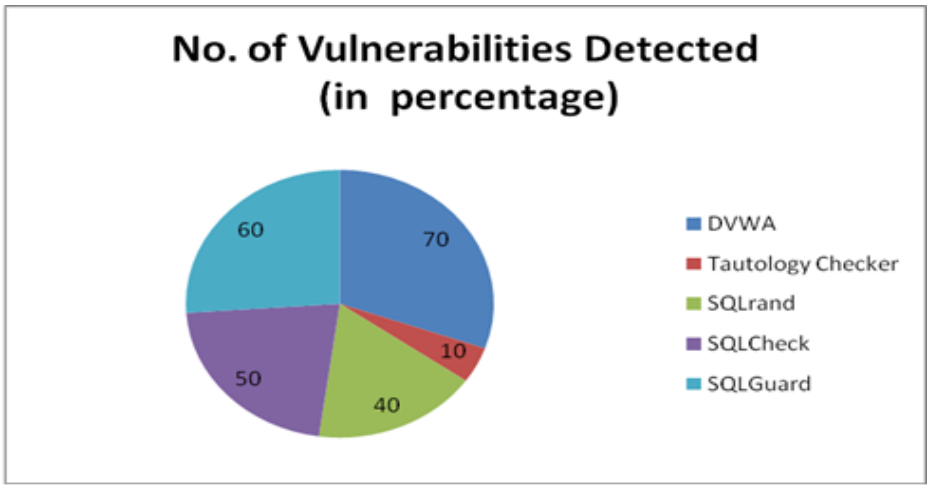



**Figure 2.2 Comparisons of Vulnerabilities detection percentage**

**4.2 Cross Site Scripting Attack**

The Table 6.4 shows the comparison of different types of browser with different types of XSS attacks. From the comparison shows that firefox with XSS- Me add-on performs better than other exiting browser.

**Table** Error! No text of specified style in document.**.1 Comparison of browsers for XSS attacks detection**

| **Different types of XSS attacks** | **Firefox** | **Firefox with XSS-Me Add on** | **Internet Exploer** |
|---|---|---|---|
| Fragmented XSS | Exploitable | Non- Exploitable | Exploitable |
| Hyperlink injection | Exploitable | Non- Exploitable | Exploitable |
| Injection into body script taken from other source | Exploitable | Non- Exploitable | Exploitable |
| Using JavaScript url | Non-Exploitable | Non- Exploitable | Non- Exploitable |
| Iframe | Exploitable | Non- Exploitable | Non- Exploitable |
| Image src | Exploitable | Non- Exploitable | Exploitable |

Table 6.5 show evaluation results of web site which is tested for XSS attacks. For the XSS detection, add-on XSS-Me extension is used in firebox browser.

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**

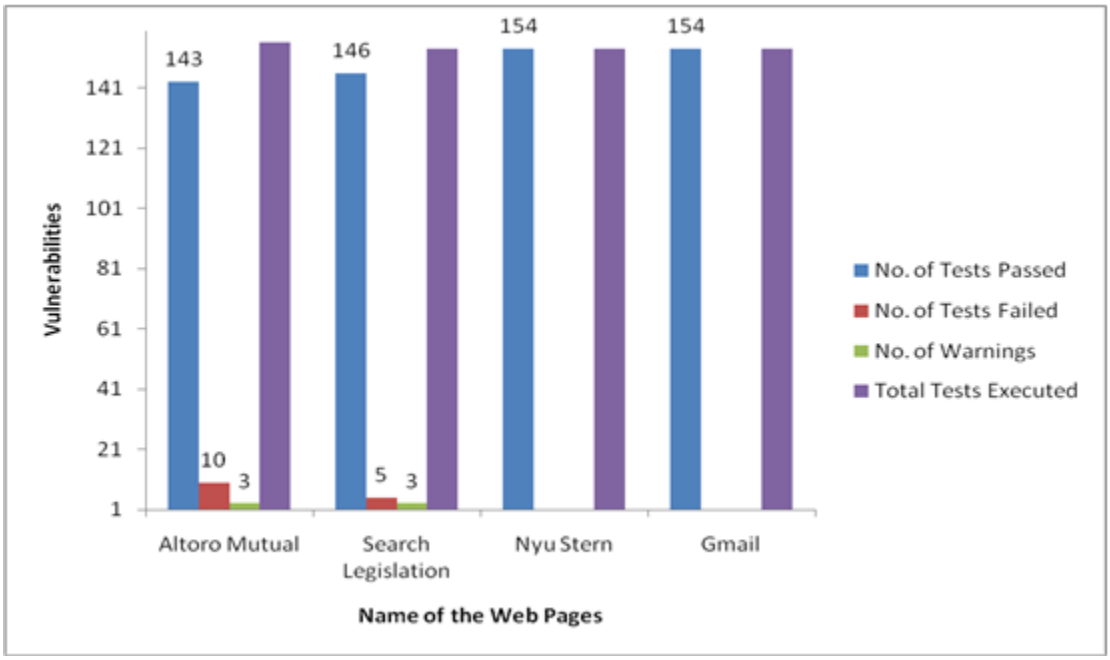



Figure 6.5 shows the comparisons of different web pages. This chart exhibits name of the web pages and vulnerabilities in X and Y axis and explains about result evaluation for some of the websites. The evaluation result demonstrates the count to tests passed, failed and warnings.

**Co-resident attack**

Table 6.7 show the parameters cloud user, Cloud data center, capacity of each data center and server are used in this work to detect the possible co-resident to the target VM .

**Table** Error! No text of specified style in document.**.2** **Cloud Resources Used**

| Parameters | Used |
|---|---|
| Server | 3 |
| Cloud Data center used | 1 0 |
| Capacity of each datacenter | 1tb |
| Cloud Users | 20 |

The victim client request instances are launched in T1and $T_2$ time intervals. After 5 mints a separate malicious web request instances are launched using VM allocation policy. The numbers of co-resident with a victim instance are tabulated in the Table 6 .8

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**

**Table** Error! No text of specified style in document..**3** **Co-resident occurrence at Time $T_1$ and $T_2$**

| Time | Victim Request | Attacker Request | Attack Coverage VM Allocation Policy |
|---|---|---|---|
| $T_1$ | 1<br>5<br>15 | 10<br>10<br>20 | 0/1<br>3 /5<br>10 /15 |
| $T_2$ | 1<br>7<br>13 | 10<br>10<br>15 | 0/1<br>4/7<br>12/13 |

Table 6.9 shows that results of launching malicious request 5 minitus after the launching victim request. The right most column specifies sucess coverage, the number of victims for which a malicious request was co-resident over the total number of victims.

**Table** Error! No text of specified style in document..**4** **Allocation policy of VM with the possibilities of co-resident occurrence**

| Cloud Data center | Cloud Users Request | Allocation policy used | Possibilities of co- resident occurrence | |
|---|---|---|---|---|
| 10 | Less than 5( < 5) | Least Allocation policy | Low | 10 % |
| 10 | Between (15 -15) | Random Allocation policy | Medium | 30 % |
| 10 | Between (40-50) | Most Allocation policy | High | 60 % |

The Figure 6.14 shows the comparison of different policy with percentage of possible co-resident attacks. with most allocation policy co- resident attack is more possible and with least allocation policy the co-resident attack is very low.

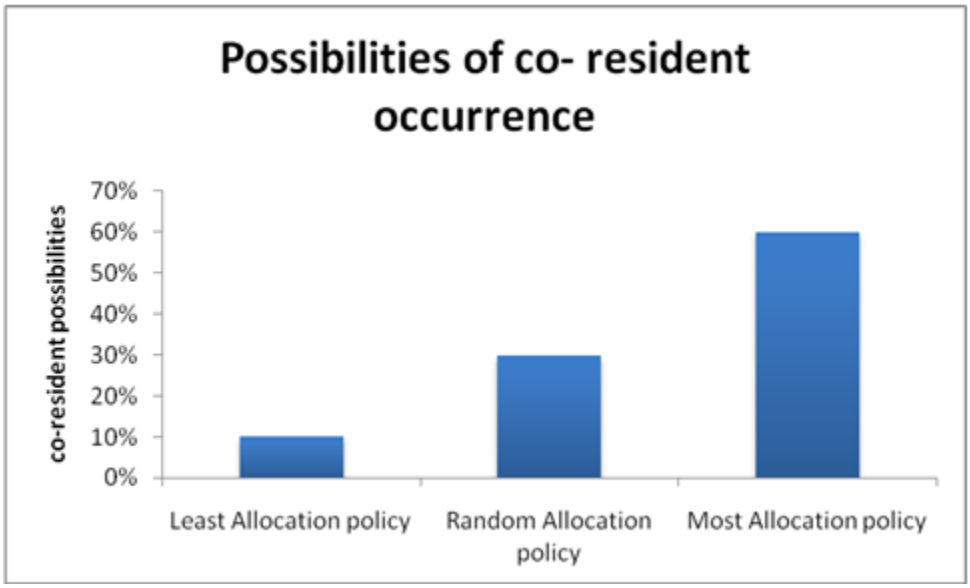
ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**




**Comparison co-resident attack with allocation policy**

## V. CONCLUSION AND FUTURE WORK

In this paper, hybrid detection techniques has been proposed and implemented to detect, the cloud malware injection attacks and co-resident attacks. The proposed work concentrates on dynamic attacks in client side, database and cloud infrastructure level. In this work, most popular event driven cloud simulator, cloudsim tool is used to create cloud environment. Java is the most powerful object oriented programming language is being used in cloudsim. Network analyzer tool is used for analyise each request and filter the normal request and malicious request. In this work, the limited cloud attacks are consider, in future other attacks may consider using more sophisticated techniques .

## REFERENCES

1. Mell, P &Grance, T 2011, 'The NIST definition of cloud computing', NIST Special Publication, vol. 145, pp. 7-7.
2. Nancy, Silakari, S &Chourasia, U 2016, 'A Survey Over the Various Malware Detection Techniques used in Cloud Computing', International Journal of Engineering Research & Technology (IJERT), vol. 5, no. 02, pp. 398-402.
3. Bodkhe, PAP 2015, 'Cloud Computing Security : An Issue of Concern', International Journal of Advanced Research in Computer Science and Software Engineering, vol. 5, no. 4, pp. 1337-1342.
4. Saeed, IA, Campus, JB, Selamat, MA, Ali, M &Abuagoub, MA 2013, 'A Survey on Malware and Malware Detection Systems', International Journal of Computer Applications, vol. 67, no. 16, pp. 975-8887.
5. Singh, MA &Shrivastava, DM 2012, 'An Overview of Security Issues in Cloud Computing', International Journal of Advanced Computer Research, vol. 2, no. 1, pp. 41-45.

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Visit: www.ijirset.com*

**Vol. 7, Issue 1, January 2018**

6. Deshmukh, P &Agarkar, P 2015, 'Mobile Application for Malware Detection', International Research Journal of Engineering and Technology (IRJET), vol. 2, no. 2, pp. 886-893.
7. Gamayunov, D, Quan, NTM, Sakharov, F &Toroshchin, E 2009, 'Racewalk: fast instruction frequency analysis and classification for shellcode detection in network flow', in Computer Network Defense (EC2ND), 2009 European Conference on, pp. 4-12.
8. Sagar, V & Kumar, VR 2013, 'Malware Propagation Detection in Mobile Cloud Infrastructure with Architectural Change', International Journal of P2P Network Trends and Technology (IJPTT), vol. 3, no. 10, pp. 436-441.
9. Masud, MM, Al-Khateeb, TM, Hamlen, KW, Gao, J, Khan, L, Han, J &Thuraisingham, B 2011, 'Cloud-based malware detection for evolving data streams', ACM Transactions on Management Information Systems, vol. 2, no. 3, pp. 1-27
10. Park, J-o 2013, 'A Study on Detection of Hacking and Malware Codes in Bare Metal Hypervisor for Virtualized Internal Environment of Cloud Service', IJCSNS International Journal of Computer Science and Network Security, vol. 13, no. 10, pp.
11. Harrison, K, Bordbar, B, Ali, STT, Dalton, CI & Norman, A 2012, 'A framework for detecting malware in cloud by identifying symptoms', Proceedings of the 2012 IEEE 16th International Enterprise Distributed Object Computing Conference, EDOC 2012, pp. 164-172.
12. Zhang, Y, Juels, A, Oprea, A & Reiter, MK 2011, 'Homealone: Co-residency detection in the cloud via side-channel analysis', in Security and Privacy (SP), 2011 IEEE Symposium on, pp. 313-328.
13. **Maheswari KG** & Anita R 2015 "Generalization of minkowski distance metrics in mixed case analysis for web intrusion detection" in **International Journal of Soft Computing,** vol. 10, no. 4, pp. 274-278, ISSN ISSN 1816-9503.
14. **Maheswari KG** & Anita R 2015 "A dynamic tool for detection of xss attacks in a real time environment" **ARPN Journal of Engineering and Applied Science**, vol. 10, no. 10, pp. 4627- 4634, ISSN 1819-6608.
15. **Maheswari KG** & Anita R 2016 'A dynamic virtual machine security scheme against co resident attack in cloud computing' **Asian Journal of Information Technology**, vol. 15, no. 13, pp. 2116-2122, ISSN 1682-3915. (Updated Journal 2016 IF: 0.22)
16. Nalinipriya, **K.G. Maheswari**, Balamurugan Balusamy , K. Kotteswari and Arun Kumar Sangaiah , "Availability modeling for multi-tier cloud environment" **IntellIgent AutomAtIon & Soft ComputIng,** Vol. 23, no. 3, PP. 485–492, ISSN: 1079-8587, 2017.
17. G. Nalinipriya, P.J.Varalakshmi, K.G.Maheswari, R. Anita, 'An Extensive Survey on Co- Resident Attack in Dynamic Cloud Computing Environment" published in **International Journal of Applied Engineering Research ,** ISSN 0973-4562 , Volume 11, Number 5 , pp 3019-3023 , **2016.**