ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

An Audit of Server Hardening in Multi-Tenantarchitecture System in Cloud Computing

Venkatesan Balu*, Dr .L.Josephine Mary
*Research Scholar of St. Peter's University, St.Peter's University, Avadi, Chennai, Tamilnadu, India
Professor &Head, MCA, Sri Ram Engineering College , Perumalpattu, Chennai, Tamilnadu, INd

Abstract

In cloud computing, multi-tenancy is the architecture in which hardware and software can be shared with multiple users (tenant). In existing system, there is no protection on the operating system level when corporate or individual is reserving any system in cloud computing by default. Most of the corporate or individuals are using either "Single Tenant" or "Multi-Tenant" model. In both of the approach there are vulnerable attacks. And they do not have much knowledge of the "Server Hardening" to secure their servers.

The main objective of the proposed system "An Audit of Server Hardening in Multi-Tenant Architecture System in Cloud Computing" is to secure multi-tenancy architecture system. In this model, the server or system is secured by applying technique called 'Server Hardening' using Linux server from 'Amazon' called "Amazon Elastic Compute Cloud (Amazon EC2". To audit the level of security a cloud system provides an open source tool called 'Lynis'. This tool was utilized to determine the state of hardening of the present available system such as Fedora, Red Hat, Ubuntu etc., that cloud services vendors provides. The future of the system, to apply the 'Server Hardening' to secure the system or server, need to create a module for major Linux family operating systems like Ubuntu, RedHat and SuSE.

Keywords: CC-Cloud computing, AWS- Amazon Web services, MT-multi-tenant.

Introduction

Cloud computing is presently the IT solution for excessive budgeting of many organizations[1]. To manage the amount of IT solutions firms have to spend a lot on development of infrastructure. But, since we are in the evolutionary phase of IT solutions, companies are switching towards the idea of cloud computing. It is a revolutionary IT solution as it downgrades the infrastructure expenditure manifolds. Cloud computing offers assets like software and hardware on-demand basis. This kind of availability of software and hardware including sizeable memory and storage resource along with computing capabilities allows the IT companies to reduce their infrastructure expenditure. Cloud has three subsets namely, Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS)[2].

Cloud computing is a concept that has gone through an evolutionary change since 1950's. The idea of providing shared assets is not something new. The concept of time sharing was introduced in 1950 with large scale mainframe systems which were accessible via Remote Terminal. This concept can be thought as the birth of cloud computing. Since the advancement in dot-com services there has been an ever increasing demand for advance computer networking solutions. Amazon played a key role in developing cloud computing and started providing Amazon Web Services as API's to the customers. It is very expensive to maintain on-premises devices and upgrading them as and when the technology takes a new turn. It is physically, financially and time wise not conceivable. Most importantly it eats time to upgrade to higher capacity. So there's an infrastructure constraint, but not just that think about the fact that the software is developing every day with more and more advanced features. For instance when there was a dual core CPU, applications were written accordingly, but with more advancement such as quad core processors applications have become more and more robust giving rise to need of upgrading the systems. But it's not feasible to keep upgrading whenever a new feature in an application is introduced as it takes long man hours to migrate from one system to another.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Requirement of cloud computing

The fact is that it is very expensive to maintain on-premises devices and upgrading them as andwhen the technology takes a new turn. It is physically, financially and time wise not conceivable. Most importantly it eats time to upgrade to higher capacity. So there's an infrastructure constraint, but not just that think about the fact that the software is developing every day with more and more advanced features. For instance when there was a dual core CPU, applications were written accordingly, but with more advancement such as quad core processors applications have become more and more robust giving rise to need of upgrading the systems. But it's not feasible to keep upgrading whenever a new feature in an application is introduced as it takes long man hours to migrate from one system to another.

Thus, these constraints gave rise to Cloud Computing as it provides a solution for Infrastructure as well as provides a stable platform thus eliminating migrations often.

Problem discussion

Before we dwell into how to make our multi-tenant system secure we must look at the points of vulnerability. It is believed that the multi-tenant system is provided with primitive security, is secure but in concept it is susceptible to threat. Let's look at the threats one by one. At the *core programming level*, Java and .NET, widely used software development environments, considered secure is quickly becoming a myth. The .NET susceptibility can be explained by finding out the target DLL**[3]** from the Global Assembly Cache**[4]** (GAC), analyzing it, modifying it and recompiling it and overwriting the code with the tampered one. Whereas, Java is susceptible by using Java Snoop**[4]** which can intercept the method calls, alter parameters and insert the code on the fly. At the *VM* level, where tenants reside, debugging cause challenges as the tracking of the process can be done without needing any privileges. For example, LiveCloudKD**[6]** which allows us to run the Windows Kernel Debugger for guest windows user from Microsoft Hyper-V R2 hypervisor can monitor the process thus making it vulnerable.

Operating System that resides on top on VM are prone security lapses. The privilege separation is by and far the only way to make system secure but is vulnerable to Trojan Horses and buggy software thus compromising security. Exploitation of vulnerabilities in the OS kernel or inadvertent execution of untrusted software by a privileged user may lead to introduction of covert malware[5] inside the kernel; i.e., a kernel rootkit.

The *application layer* vulnerabilities cannot be overlooked as the data is not secure at either the Database or the application that uses the data. Finally, the *web security* will also bought into contention as the Cloud Computing utilizes the web as the means to access the remote computer[5]. The vulnerability can exist at the server side wherein, the content flows from and into the browser. The content is secured by using protocols (HTTP) that prevent intrusion, but data becomes vulnerable as there is always a cross domain data communication. Access control intrusion occurs when access control in web application is incorrect or missing, allowing unauthorized access to privileged resources.

Nemesis is a system that automatically tracks the flow of user credentials when authentication is performed. It generates an additional HTTP cookie to track requests from an authenticated user by dynamic information flow tracking (DIFT) and runs shadow authentication with this information to enforce developer-specified access control rules. It requires no modification to the application, either to the authentication or its access control system. Client side validation offers more protection by monitoring the data both at end point and middle box.

A Cloud provider would provide a system with a certain set of configuration with standard security. Vendor will be just a custodian to the system. Since the system has standard security, it can be vulnerable. To prevent this vulnerability this paper would highlight certain techniques that would make your system secure and will not hamper your functionality in any way. Such technique is called *Server hardening*. It is an existing concept which is implemented in to secure individual or corporate systems. This paper will look into how this particular concept can be implemented in Cloud Computing.

A model of server hardening in multi tenancy architecture system in cloud computing server hardening

Server Hardening is the process of implementing security through variety of means, thus making the system secure. To protect our servers we must establish solid and sophisticated server hardening policies for all servers in our organization.

In Fig.1, we can visually analyze the advantage of running the script and presents an understanding of server hardening in multi-tenancy architecture. The approach towards server hardening should start by creating a checklist. Let's peek into what the checklist would like for server hardening.

- Avoid Using FTP, Telnet and rlogin / rsh. Instead, need to us OpenSSH, SFTP or FTPS
- Keep operating system up to date, especially security patches.
  1. User Accounts and Strong Password Policy Password Aging
  2. Restricting Use of Previous Passwords
  3. Locking User Accounts After Login Failures
- Do not permit empty passwords.
- Disable root Login. Logging and Auditing Disable unwanted Services Configure the system firewall (IP tables) and consider also using a hardware firewall
- Encrypt Data Communication.
- Linux Kernel /etc/sysctl.conf Hardening Separate Disk Partitions, Protecting Files, Directories and Email
- Use A Centralized Authentication Service (OpenLDAP)
- Install And Use Intrusion Detection System

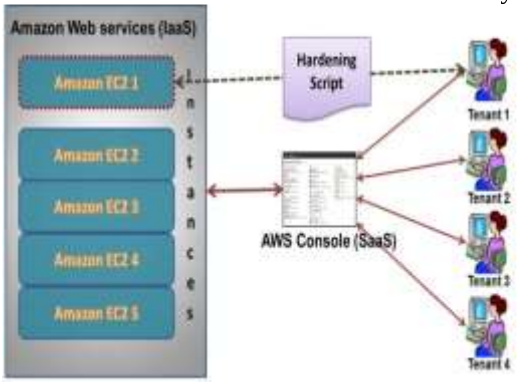

Fig 1. A Model of Server Hardening in Multi Tenancy Architecture System in Cloud Computing

Once the checklist is ready with us, it's as simple as following the checklist. It is tedious to manually carry out the disabling of the unwanted services, root logging, configuring a system firewall, Linux kernel hardening and install an Intrusion detection system. Thus, this process needs to be automated. The checklist that is provided in this section is just a basic checklist; it can be made exhaustive thus, making the system impenetrable to the attacks from external sources. It is just a matter of updating the shell script to our needs and executing it. The Shell Script can be automated to run while booting so that we are presented with a secure environment to work on.

Audit of Server Hardening in Cloud Systems

To audit the level of security a cloud system provides an open source tool called 'Lynis'[9] was used. It is an open source auditing tool that performs security scan with following compliance; Basel II, GLBA, HIPAA, PCI DSS and SOx (Sarbanes-Oxley) and determines the state of hardening of the Linux system. This tool was utilized to determine the state of hardening of the present available system such as Fedora, Red Hat, Ubuntu etc., that cloud services vendors provides.

Results

The server hardening is measured as an index in the *Lynis* tool. The tool was run and various parameters were listed, the most useful for us being the *Hardening Index*. **The Fig. 2** shows that the hardening index of various Linux operating systems which were created from the cloud instance is measured and a chart for the same is generated. In this chart we can analyze that the hardening index of Red Hat Linux operating system is the best. But the Hardening quotient is not high.

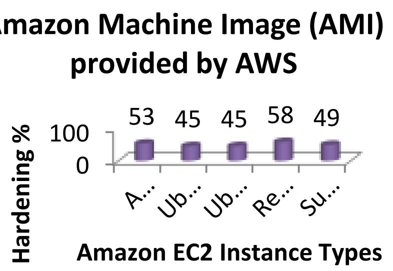

Fig 2.Chart representing the of Hardening Index of Linux OS

The **Table 1** shows that the various Linux operating system versions, kernel version and hardware

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

platform with hardening index. All the instances were created as cloud instance in Amazon Web Service. Thus, this chart proves that the instance is susceptible to various attacks.

Amazon EC2 Instance Type	Kernel version	Processor Arch	Hardening Index %
Amazon Linux 3.4.73	x86_64	x86_64	53
Ubuntu 12.04	3.2.0-58-virtual	x86_64	45
Ubuntu 13.1	3.11.0-15-generic	x86_64	45
Red Hat 6.4	2.6.32-358.14.1.el6.x86_64	x86_64	58
SuSE ES 11	3.0.82-0.7-ec2	x86_64	49

Table 1. Hardening Index of various Linux Operating Systems

Conclusion and future enhancement

The data presented in the results section enables us to understand the level of hardening that is provided in our Linux system. To work in cloud environment we need a secure system as, although we are cutting down our infrastructure cost by implementing cloud based services, doesn't necessarily mean we have to forgo security. In this modern era where systems are hacked not just for monetary and snooping gains, it becomes imperative that security of systems takes prime importance. There are ways to protect your system and we have just proposed an idea that will provide us an improved system with better security settings. The script can be customized for our needs and can provide us with a more secure system without waiving our systems capacity and capability.

References

1. "Amazon Elastic Compute Cloud", www.aws.amazon.com/ec2.
2. University of Maryland, Introduction to Cloud Computing September, 2008
3. E. Metula, ".NET Framework Rootkits: Backdoors Inside Your Framework," in BlackHat Europe, 2009.
4. A. Dabirsiaghi, "JavaSnoop: How to hack anything in Java," in BlackHatLas Vegas, 2010.
5. J. Franklin, et al., "Remote detection of virtual machine monitors with fuzzy benchmarking," SIGOPS Oper. Syst. Rev., April 2008.
6. C. Percival, "Cache missing for fun and profit," in BSDCan, 2005.
7. S. King and P. Chen, "Subvirt: implementing malware with virtual machines," in IEEE Symposium on Security and Privacy, May 2006.
8. Moonsols,"LiveCloudKd," http://www.moonsols.com/2010/08/12/livecloudkd/ , Aug. 2011.
9. Rootkit Lynis Documentation "http://www.rootkit.nl/projects/lynis.html", Lynis – Copyright 2007-2009, Michael Boelen - The Netherlands
10. "Cloud computing", www.GoGrid.com.