Rapid and Proactive Approach on Exploration of Vulnerabilities in Cloud based Operating Systems

S. Ramachandran
Department Of Computer Science and Engineering
Anna University of Technology, Tiruchirappalli.

A. Ramachandran
Department Of Computer Science and Engineering
Anna University of Technology, Tiruchirappalli.

ABSTRACT

Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These assets can be dynamically reconfigured to adjust to a variable scale, allowing also for optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the infrastructure provider by means of customized service level agreement (SLA). Cloud computing was originally designed for dealing with problems involving large amounts of data and/or compute-intensive applications. The vulnerabilities inherent in the Cloud systems should be addressed so they can be eliminated before exploited by malicious software or hackers. Our approach plays a major role in detecting and managing vulnerabilities present in the Cloud infrastructure. Implementation of this methodology proves to be cost effective and saves analyzing time .

Keywords

Cloud Computing, Cloud Security, Cloud Legal Issues, Cloud Storage, Security Implications, Architecture, Implementation, Exploitation, Vulnerabilities.

1. INTRODUCTION

Cloud computing has been developed to reduce IT expenses and to provide agile IT services to individual users as well as organizations. It moves computing and data away from desktop and portable PCs into large data centres. Cloud computing depends on the internet as a medium for users to access the required services at any time on pay-per-use pattern. However this technology is still in its initial stages of development, as it suffers from threats and vulnerabilities that prevent the users from trusting it. Various malicious activities from illegal users have threatened this technology such as data misuse, inflexible access control and limited monitoring.

1.1 Understanding Cloud Computing

Cloud computing is a significant advancement in the delivery of information technology and services. By providing on demand access to a shared pool of computing resources in a self-service, dynamically scaled and metered manner, cloud computing offers compelling advantages in speed, agility and efficiency. Today, cloud computing is at an early stage in its life cycle, but it is also the evolution and convergence of several trends that have been driving enterprise data centres and service providers over the last several years.

Cloud computing is a model for enabling convenient, on-demand network access to a public 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. This definition from the National Institute of Standards has gained broad support from the industry.

Cloud computing has been largely classified based on different capabilities such as Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS).

Software as a Service (SaaS) – Applications delivered as a service to end-users typically through a web browser. There are hundreds of SaaS offers available today, ranging from horizontal enterprise applications to specialized applications for specific industries, and also consumer applications such as web-based email. Google Docs is an example of a SaaS model that provides both multi-tenant as well as single-tenant options, depending on the customer's preference.

Platform as a Service (PaaS) – An application development and deployment platform delivered as a service to developers who use the platform to build, deploy and manage SaaS applications. The platform typically includes databases, middleware and development tools, all delivered as a service via the Internet.

Infrastructure as a Service (IaaS) – Compute servers, storage, and networking hardware delivered as a service. This infrastructure hardware is often virtualized. So virtualization, management and operating system software are also part of IaaS as well. An example of IaaS is Amazon's Elastic Compute Cloud (EC2) and Simple Storage Service (S3).

1.2 Data Protection and Portability

Within the cloud, data may be hosted anywhere in the world, but needs to be protected against privacy intrusions and fraudulent use the sensitive data, it is highly vulnerable for any organization [33]. In general, all the cloud services are offered based on some basic agreement between customer and a service provider. But one day all the agreements will reach the termination date or expiry date. What will happen when the contract is terminated and client doesn't want to continue anymore?

On the other hand, in exceptional cases when the client is not satisfied with the cloud computing services offered by an organization the client eventually withdraws from the organization leaving behind confidential data that may be vulnerable to the organization from which the client has withdrawn. In this situation, data protection and portability relics as one of the main weaknesses of cloud computing.

1.3 Vulnerability Management

Vulnerabilities are flaws or bugs in operating systems, software applications or networking protocols that can be exploited by adversaries to obtain access or elevate access privilege of the computer systems. Vulnerability management is a security practice by which organisations set up procedures to discover vulnerabilities in its systems/applications and to fix the vulnerabilities pro-actively before being exploited. Patches are additional programs added to the original software or systems to address the vulnerabilities.

Like bandages, patches fix vulnerabilities or reduce the impact of the vulnerabilities once exploited. Applying patches timely is critical since attackers these days usually exploit vulnerabilities soon after they are discovered. Hence, automatic vulnerability scanning and a standard patch management process enable the risk of vulnerable systems/applications.

1.4 Auditing Vulnerability

Auditing is no longer a financial term but is associated with evaluation of the internal controls of the system. IT audit is concerned with determination of the risks that are likely to attack the target system and mitigate these risks. Protection of Information assets of an organization is served by IT auditing tools which reviews and evaluates the organization's data availability, confidentiality and integrity.

With rapid growth in both the number and sophistication of cyber attacks, it has become imperative that cyber defenders be equipped with highly effective tools that identify security vulnerabilities before they are exploited. Vulnerability can be defined as a set of conditions which if true, can leave a system open for intrusion, unauthorized access, denied availability of services running on the system or in any way violates the security policies of the system [2].

While breaches happen at every corner of an enterprise network [26], often the security of the end hosts is the most brittle line of defense. A breach of security occurs when a stated organizational policy or legal requirement regarding information security, has been contravened.

The only way for a business to protect its most critical data from both outside hackers and unscrupulous insiders is to ensure that the database is properly configured, patched, and locked down.

Our approach is to find security holes and configuration problems in the operating system and then present the user with a report detailing the issues found and recommended fixes. Our identification of missing patches, weak passwords, and insecure configuration settings is the main purpose for using this proactive approach. It can be configured to perform configuration policy automatically determining if applications are configured properly, and calling out exactly which settings violate the policy. The proposed architecture has the following advantages over the conventional design. Our proactive approach plays a major role in detecting and managing vulnerabilities in complex computing systems.

It provides an overall view of the vulnerabilities in the operating system, by automatically scanning them with minimum overhead. It gives a detailed view of the risks involved and their corresponding ratings. Based on these priorities, an appropriate mitigation process can be implemented to ensure a secured system. The results show that our tool could effectively optimize the time and cost involved when compared to the existing systems.

1.5 Vulnerabilities on Cloud Computing

- ➢ Abuse use of cloud based operating system
- ➢ Anxious interfaces and APIs
- ➢ Malicious insiders
- ➢ Shared technology issues
- ➢ Data loss / leakage
- ➢ Account / Service hijacking
- ➢ Unknown risk profile

1.6 Attacks

The cloud storage attacks can be classified into three broad categories network, resource based and browser based attacks [22].

1.6.1 Denial of Service

Denial of Service (or DoS for short) attacks are a kind of attacks against computers connected to the Internet. DoS attacks utilize bugs in a specific operating system or vulnerabilities in TCP/IP implementation. Unlike a privacy attack, where an adversary is trying to get access to resources to which it has no authorization, the goal of DoS attacks is to keep authorized users from accessing resources. The infected computers may crash or disconnect from the Internet. In some cases they are not very harmful, because once you restart the crashed computer everything is on track again; in other cases they can be disasters, especially when you run a corporate network[39][43][35].

1.6.2 Buffer Overflow

A buffer overflow occurs when a program or process tries to store more data in a buffer (temporary data storage area) than it was intended to hold. Since buffers are created to contain a finite amount of data, the extra information can overflow into adjacent buffers, corrupting or overwriting the valid data held in them. Although it may occur accidentally through programming error, buffer overflow is an increasingly common type of security attack on data integrity. In buffer overflow attacks, the extra data may contain codes designed to trigger. Specific actions, in effect sending new instructions to the attacked computer that could, for example, damage the user's files, change data, or disclose confidential information [21][40][42].

1.6.3 Virtual Machine Based Root kit

Virtual machine (VM) technology has many positive uses. However, when a VM is paired with a rootkit, you have a problem called a VM-based rootkit (VMBR). The way a VMBR works is to load itself underneath the existing OS. The existing OS then runs as a VM on top of the VMBR [29]. When running this way, a VMBR could go undetected unless special tools are used to look for its existence. VMBRs are possible for both Linux and Windows platforms.

VMBR to run underneath an existing OS, the system's boot sequence must be modified so that the VMBR loads first. Modifying the system boot sequence requires a high level of privilege or an easily duped user. The white paper authors point out several possible inroads, including remotely exploitable system vulnerabilities, a malicious bootable CD-ROM or DVD, software from a corrupt vendor, and of course malicious software run by a naive user who's logged on with Administrator privileges. A Virtual Machine Based Rootkit is installed underneath the installed operating system. A computer would boot the rootkit, which in turn loads the existing operating system in a virtual machine. While the user unknowingly works in that virtual environment, a second, hidden virtual machine would perform all kind of devious tasks [38].

1.6.4 Man in the Middle

The man-in-the middle attack intercepts a communication between two systems. For example, in an http transaction the target is the TCP connection between client and server [23]. Using different techniques, the attacker splits the original TCP connection into 2 new connections, one between the client and the attacker and the other between the attacker and the server [36].

1.6.5 Replay Attack

Replay attack is a network attack in which a valid data transmission is repeated or delayed. In this attack, the attacker captures the network traffic and replays it at a later time. Thus he can gain access even to the encrypted unauthorized resources. The effect on Dynamic Rights Management (DRM) is more high as the rights over the resource changes over the time or number of times/ the bandwidth used. Here, the user not only loses privacy, but also the expenditure involved in the dynamic rights. When a user tries to access a file from cloud storage service, the charge based on the amount of data transferred will be high due to this replay attack [24]. For example, suppose in the communication of two parties A and B; A is sharing his key to B to prove his identity but in the meanwhile attacker C eavesdrop the conversation between them and keeps the information which are needed to prove his identity to B. Later C gets in contact with B and proves its authenticity.

1.6.6 Resource Exhaustion

Resource exhaustion is a simple denial of service condition which occurs when the resources necessary to perform an action are entirely consumed, therefore preventing that action from taking place.

This is one of the most specific types of attack caused by undefined consumption, allocation and depletion of system resources. Hence resource exhaustion vulnerability can be exploited to cause Denial of Service (DoS) attacks due to bad design, inefficient utilization of resources on the service side and resource leakage. It is difficult to identify the cause unless it is monitored effectively. “Antunes et al” projected a implemented Predator methodology to predict resource exhaustion vulnerability [25].

1.6.7 Byzantine Failure

In cloud storage many servers and users interact with each other by accessing a single source [27]. If any of the server or user fails due to some crash or malicious activity, then it is called as Byzantine Failure. Wang et al [37] proposed Agreement Protocol for cloud computing (APCC), which consists of two processes namely interactive consistency process and the agreement process [20]. The interactive consistency process is executed at the server nodes where the initial messages are stored and shared. This server node then aggregates the results and transmits the message to client nodes. The agreement process is executed at the client nodes to receive the agreed message.

1.6.8 Malware

Malware (for "malicious software") is any program or file that is harmful to a computer user [44]. Thus, malware includes computer viruses, worms, Trojan horses, and also spyware, programming that gathers sensitive information about a computer user without permission [28].

Normally, Web browsers get affected by this malware as it supports third party programs through "Add-on" or "Plug-in". Louw et al has proposed a malware program in which sensitive information such as passwords can be captured even when the communication is done using Secure Socket Layer (SSL)[31].

1.6.9 XML Wrapping

A very important technology to implement Service Oriented Architecture (SOA) for interoperable and platform independent services is Web Services. Extensible Markup Language (XML) is the underlying markup language used to communicate between server and client. A SOAP message with a signed body produces a valid hash by moving to different wrapper message without changing their signatures [30]. This is called XML Wrapping attack. According to Gruschka et al this XML wrapping attack can be mitigated using SOAP message security validation and XML schema validation. But the formal proof of safety is missing [32].

1.6.10 SQL Injection.

It is where a hacker tries to "inject" his SQL code into someone else's database and forces that database to run his SQL [41]. This could potentially ruin their database tables, ad even extract valuable or private information from their database tables [19]. The idea behind SQL injection is to have the application under attack run the SQL injected coding that it was never supposed to run.

2. LITERATURE REVIEW

With the rapid development of more complex systems, the chance of introduction of errors, faults and failures increases in many stages of software development life-cycle [8]. This class of system failures is commonly termed as software vulnerabilities. These security vulnerabilities violate security policies and can cause the system to be compromised leading to loss of information [9]. Vulnerabilities can be introduced in a host system in different ways; via errors in the code of installed software, mis-configurations of the software settings that leave systems less secure than they should be (improperly secured accounts, running of unneeded services, etc.) [10].

Vulnerability analysis, also known as vulnerability assessment, is a process that defines, identifies, and classifies the vulnerabilities (security holes) in a computer, network, or communications infrastructure [11]. Vulnerability analysis can be used to predict the effectiveness of the proposed countermeasures and evaluate them after they are put into use. Vulnerability analysis begins with gathering, defining and classifying network or system resources. The resources can be classified according to their level of importance in the network. This stage of identification of threats can be performed by probing the network or system to discover potential weak points.

A vulnerability assessment tool or scanner can be defined as a utility that can be used to test the capability of a systems or networks security and discover their points of weakness [2].

These tools themselves do not provide any kind of security or protection to the system, rather they gather and report information, which can be used to instate a different tool, policy or mechanism to secure the system. Vulnerability assessment tools can be broadly classified into network based and host based analyzers as described in the following Sections 2.1 and 2.2 respectively.

2.1 Network Based Analyzers

A network-based scanning assessment may also detect extremely critical vulnerabilities such as mis-configured firewalls or vulnerable web servers in a De-Militarized Zone (DMZ), which could provide a security hole to an intruder, allowing them to compromise an organizations security [4].

Network assessment tools gather information and may also have network mapping and port scanning abilities. Typical network based scanner architecture is shown in Figure 1.

Furthermore the scanning engine is the main component of the network based scanner. It performs the assessment as instructed by the interactive console by sending specially constructed packets for the test.

Results repository is the final component, which holds the entire scan results received and is also used for report generation for the system administrators [12]. The strengths of network based scanners lie in the fields described below:

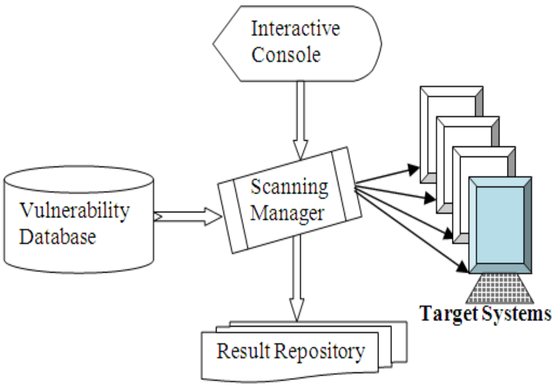

Fig 1: Architecture of a typical Network-Based Analyser

However network-based scanners also have the following demerits:-

1. A network-based scanner does not have direct access to the target's file system. Thus, it is not able to check for file permission.

2. Another problem which network based scanners face is their inability to scan targets behind a firewall. Complicated measures have been taken to let scanners get to these target systems.

2.2 Host-Based Analyzers

Host based analyzers also scan the system for vulnerabilities like the network scanners, however they are able to scan much more due to the fact that they have a service/agent residing on the target system. A typical host-based architecture is shown in Figure 2. The host-based analyser is installed on a network by first installing a scanning manager on the network.

When the manager wants to initiate a scan, it sends the necessary information like scanning policy to the agent on the host.

The scanning policy consists of the different vulnerability checks. The agent on the host scans accordingly and reports back the results of the scan. As new vulnerabilities are discovered frequently the security definitions have to be regularly updated on each agent.

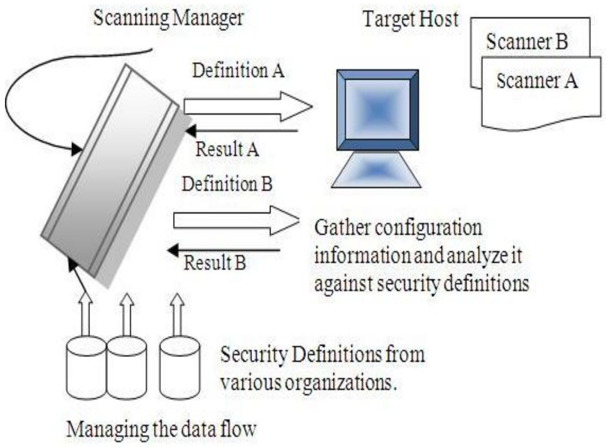
Scanning Manager
Target Host
Scanner B
Scanner A
Definition A
Result A
Definition B
Result B
Gather configuration information and analyze it against security definitions
Security Definitions from various organizations.
Managing the data flow

Fig 2: Architecture of Conventional Host-Based Analyzer.

2.3 Attack Injection Methodology

A penetration test is a method of evaluating the security of a computer system or network by simulating an attack by a malicious hacker (Figure 3). The process involves an active analysis of the system for any weaknesses, technical flaws or vulnerabilities.

This analysis is carried out from the position of a potential attacker, and can involve active exploitation of security vulnerabilities. Any security issues that are found will be presented to the system owner together with an assessment of their impact and often with a proposal for mitigation or a technical solution [1].

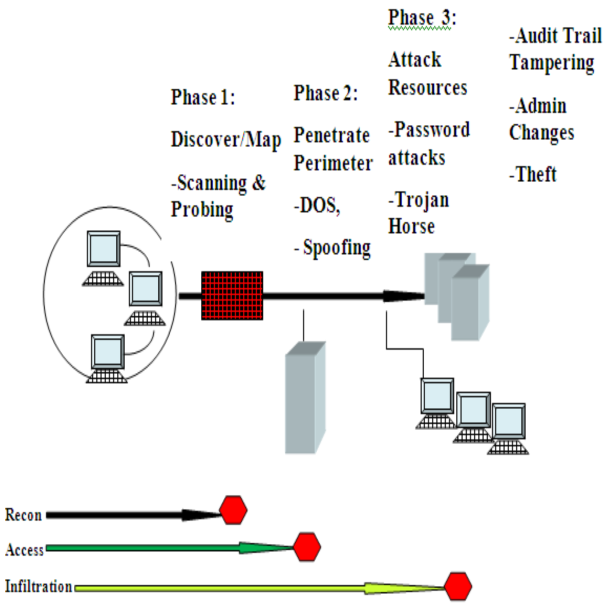

Fig 3: Penetration Test

Target system is attacked automatically by using attack generation algorithm. Discovery of vulnerabilities in server applications is based on the behavior of malicious adversaries. Generate several attacks against a target network server, while observing its execution. An attack Injection Methodology architecture is shown in Figure 4.

In the existing system attack injection methodology has been used which is not suitable for online computation system example ATM, Banking, etc.).

Since all online system are incorporated with special user profile mechanism they never allow to inject the attacks (i.e., When we enter wrong user credential more than 3 time the corresponding account get locked automatically based on predefined profile settings

Penetration testing focuses on vulnerabilities that allow command execution. Most command-execution vulnerabilities are buffer over flows, which inherently run the risk of crashing computers or services.

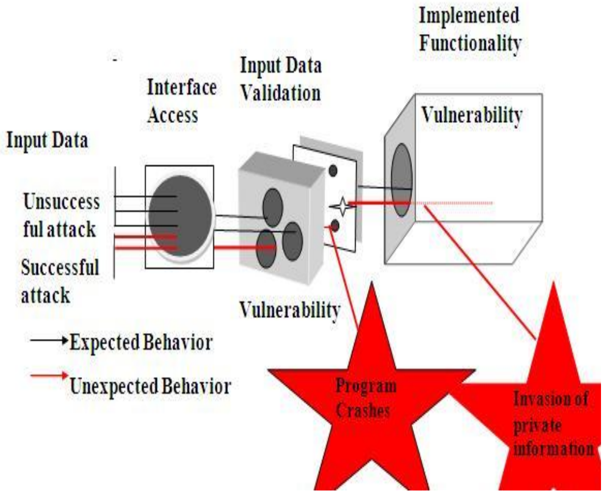

Fig 4: Attack Injection Methodology

2.4 Vulnerability Exploits In Platform As A Service (Paas)

In today's complex environment, organizing data has become the most important concern of all industries. Information security is the supreme challenge and it has become essential to secure the databases from the attacks of the unknown outsiders.

Many organizations tend to executed client and network security solutions designed to protect workstations and network resources. This is because data is not just accessed by the company's employees but is open for access by its partners as well as its clients. Thus it becomes important to protect databases from the attacks of the unknown outsiders, ID Theft and data breaches.

Vulnerability scanner involves an absolute operating system plug-in that checks a large range of vulnerabilities that could be remotely exploited in addition to remote scanning. In a typical network vulnerability assessment, a remote scan is performed against the external points of presence and an onsite scan is performed from within the network. However, neither of these scans can determine local exposures on the target system [3][6]. By using secured credentials [7], our approach [46] can be granted local access to scan the target system without requiring an agent. This can facilitate scanning of a very large network to determine local exposures or compliance violations.

The approached credential scan can quickly determine which system is out dated on patch installation. This is important especially when a new vulnerability is made public and executive management wants a quick answer regarding the impact to the organization.

3. PROPOSED SYSTEM

Rapid And Proactive Vulnerabilities Scanning (RPVS) is analogous to NAC (Network Admission Control) commonly implemented at the Network Router level to identify and quarantine new hardware systems, till they are thoroughly scanned for vulnerabilities.

When the term "Vulnerability assessment" is used in the context of vulnerability scanners it means "the process" of finding known vulnerabilities in a network. This process identifies vulnerabilities so they can be eliminated before exploited by malicious software or hackers. In most cases the vulnerabilities are known and can therefore be found. The vulnerabilities that constitute threats in a network include software defects, unnecessary services, mis-configurations and unsecured accounts [2]. Proposed architecture is shown in Figure 5.

The vulnerability scanner works with a proactive approach, it finds vulnerabilities, hopefully, before they have been used

A Zero day exploit is unknown to security professionals which mean that information about the exploit is not publicly available [5].

Our proactive approach provides an overall view of the vulnerabilities present in the operating system, by automatically scanning them with minimum overhead. It gives a detailed view of the risks involved and their corresponding ratings. The results show that our tool could effectively optimize the time and cost involved when compared to the existing systems.

Our approach is to find security holes and configuration problems in the operating system and database, and then present the user with a report detailing the issues found and recommended fixes. Our identification of missing patches, weak passwords, and insecure configuration settings is the main purpose for using this proactive approach.

3.1 Architecture Description

RPVS will detect Operating system, in the domain (range of IP addresses of an enterprise), seamlessly, quarantines it, and also scans for OS, user, data, privileges, and OS vulnerabilities. Our Approach is based on the premise that your OS, may have very critical/confidential data like SSN, Credit Card info, etc., and so the schema/metadata have to be checked before allowing users access to it, to prevent malicious users accessing confidential data.

- ➢ Scanning of ports,
- ➢ Identification of vulnerabilities,
- ➢ Generation of reports and validation of the state of security of the information technology infrastructure.

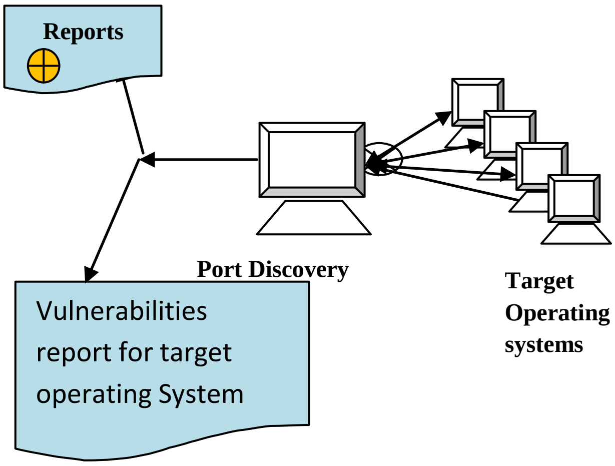

Fig 5: Proposed System

4. DISCUSSION

This chapter begins with shedding some light on the security knowledge base which is a very significant part of the architecture as shown in Figure 6. RPVS is a comprehensive enterprise Vulnerability Scanner Tool. It functions to track risk assessment of operating system. It performs port discovery of hosts and application, identification of your operating systems vulnerabilities, generation of reports.

RPVS runs on any desktop machine, It serves as an agent-less tool. In general, a vulnerability scanner is made up of four main modules, namely, a Scan Engine, Scan Database, report module and user Interface.

- ➢ Agent-less installation and data transmission on target nodes.
- ➢ Encryption of reports for secure transmission.

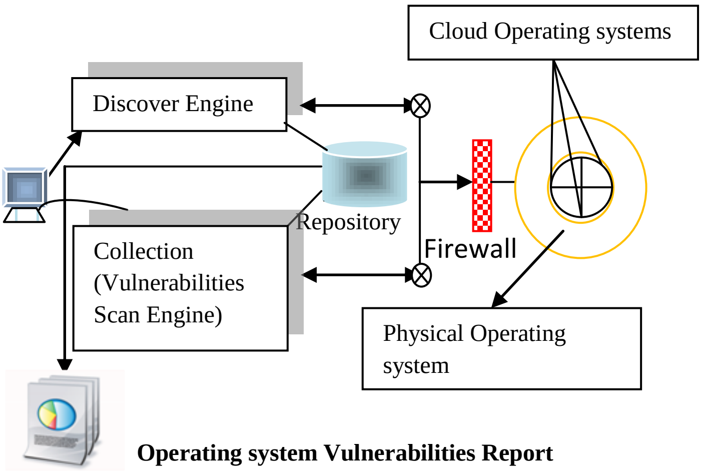
Cloud Operating systems
Discover Engine
Repository
Firewall
Collection (Vulnerabilities Scan Engine)
Physical Operating system
Operating system Vulnerabilities Report

Fig 6: RPVS System Modules

4.1 Discovery

The RPVS Discovery and Assessment Server (DAS) offers automated network-based database discovery. Identification of IT assets, such as hosts and databases installed in your network by specifying IP Addresses and Ports is done by RPVS using the auto-scan feature that scans all the IP Addresses for servers in the enterprise up to 'n' Operating assets.

- ➢ Discovers servers within your infrastructure.
- ➢ Detects unknown hosts and machines.
- ➢ Consolidates management of hosts and databases.
- ➢ Auto-scan all the IP addresses for servers in the enterprise.
- ➢ It can scan all the operating System.

4.2 Data Collection

Credentialed scans can perform any operation that a local user can perform. The Scan Engine executes security checks according to its installed plug-ins, Identifying system information and vulnerabilities.

4.2.1 Linux Systems

On Linux systems, we have uses Secure Shell (SSH) protocol version based programs (e.g., SSH, etc.) for host-based checks [16]. This mechanism encrypts the data in transit to protect it from being viewed by sniffer programs. It supports three types of authentication methods for use with SSH: username and password, implicit public/private keys generation.

4.2.2 Windows System

The configuration of Windows OS is mostly based on the registry. The registry is composed of thousands of keys. Every key can have one or more entries as well as child keys. Each entry consists of a name, a data type and a value. Windows® Management Instrumentation (WMI) is a component of the Windows operating system through which Windows resources can be accessed and configured. WMI is the underlying management technology for Windows operating systems. Primitive scanners that are basically sets of scripts code exploits producing simple plain-text files as scanning results [34]. Updates to these primitive scanners are infrequent and require manual intervention.

It has features designed to help you to secure your applications and the network,

1. Easy to use - Auto Scanning and Zero Configuration
2. Support Both Windows and Linux Operating systems Vulnerability Scanning
3. Secured Repository

4.3 Report Generation

The report module provides different levels of reports on the scan results, such as detailed technical reports with suggested remedies for system administrators, summary reports for security managers, and high-level graph and trend reports for executives. This report gives graphical representations and the information about the vulnerabilities in OS for two different configurations. Risk level identify is to monitor the operating vulnerabilities.

Microsoft Windows OS

- ➢Authentication
- ➢File systems security
- ➢Network security
- ➢System Information

Linux Operating system

- ➢Authentication
- ➢File systems security
- ➢Network security

- ➢System Information

4.3.1 Windows Authentication

Operating system verifies users, identifying by using authentication mechanism [13][14][15]. By default, Windows stores account passwords in encrypted format within the registry and file system. The table that stores the passwords is restricted from direct access by any user account. Table 1 represents the check list of Windows authentication [17].

Table 1(Windows Authentication)

Authentication Category
User Accounts
Groups In System
Users In The System
Administrator Account Not renamed
Current Login User Name
Guest Account Not Disabled
Account Expiration Of The Users
Last Logon Users With Time

4.3.2 Linux Authentication

Operating system verifies users, identifying by using authentication mechanism [13][14][15][18]. By default, Linux stores the passwords in either the etc/passwd. Table 2 represents the check list of Linux authentication [17].

Table 2(Linux Authentication)

Authentication Category
Allowed Terminals For Root Login
Users In Oinstall, wheel, dba Groups
Users Having Maximum Password Usage
Users Having Minimum Password Usage
Users Password Last Changed (in Days)
Users Never Login In
Users Last Login Info
Currently Logged On Users

5. CONCLUSIONS

In this research paper we have discussed the characteristics of a cloud that contains threats and vulnerabilities. Cloud computing has a dynamic nature that is flexible, scalable and multi-shared with high capacity that gives an innovative shape of carrying out business [45]. Through the discussion, we identified three major factors that affect cyber security information in cloud computing: data-asset decoupling, composition of multiple resources and external resource usage. By this approach we can optimize time and cost, such that the vulnerabilities can be eliminated before exploited by malicious software or hackers, in the network component. The complexity of modern enterprises, their reliance on technology, and the heightened interconnectivity among organizations are rapidly evolving developments that create widespread opportunities for theft, fraud, and other forms of exploitation by offenders both outside and inside an organization. Internal and external perpetrators can exploit traditional vulnerabilities in seconds. As detailed in this paper, it is envisioned that using the vulnerability scanning as process on a regular basis, it will response to problems identified will alleviate these risks.

6. REFERENCES

[1] Antunes, J.; Neves, N.; Correia, M.; Verissimo, P.; Neves, R.; Vulnerability Discovery with Attack Injection, 2010, vol. 36, pp. 357 - 370

[2] Citadel™ Security Software Inc. https://hercules.citadel.com/docs/301VulGuide.pdf,Page visited 051104.

[3] Vulnerability Assessments: http://www.bitpipe.com/tlist/Vulnerability-Assessments.html

[4] IBM Security product Information Center http://documents.iss.net/whitepapers/nva.pdf.

[5] Zero-day attack : http://en.wikipedia.org/wiki/Zero-day_exploit, page visited 061203.

[6] DATABASE

[7] Security Technical Implementation Guide : www.databasesecurity.com/dbsec/database-stig-v7r1.pdf.

[8] Oracle Database Listener Security Guide : www.integrigy.com/.../Integrigy_Oracle_Listener_TNS_Security.pdf.

[9] B.Marick, The craft of software testing, Prentice Hall.1995.

[10] I. V. Krsul, Software Vulnerability Analysis, PhD Thesis, Purdue University, 1998.

[11] Search security: www.symantec.com/connect/articles/vulnerability-assessment-survey.

[12] vulnerability analysis (vulnerability assessment); http://searchsecurity.techtarget.com/sDefinition/0, , sid14_gci1176511, 00.html

[13] R. Fussell, Vulnerability Assessment: Network based versus host based, Technical report, SANS Institute, 2002.

[14] Corregedor, M.; Von Solms, S.; "Implementing rootkits to address operating system vulnerabilities", Information Security South Africa (ISSA), 2011, pp.1-8.

[15] Lee, S.C.; Davis, L.B.; "Learning from experience: operating system vulnerability trends", IT Professional , 2003, vol. 5, pp. 17-24.

[16] Jihong Song; Guiying Hu; QuanSheng Xu; ‟Operating System Security and Host Vulnerability Evaluation", Management and Service Science, 2009. MASS '09. International Conference, 2009, pp. 1-4.

[17] Butt, S.; Ganapathy, V.; Swift, M.M.; Chih-Cheng Chang; ‟Protecting Commodity Operating System Kernels from Vulnerable Device Drivers", Computer Security Applications Conference, 2009. ACSAC '09. Annual , pp. 301-310.

[18] Alhazmi, O.H.; Malaiya, Y.K.; ‟Application of Vulnerability Discovery Models to Major Operating Systems", Reliability, IEEE Transactions, 2008, vol. 57, pp. 14-22.

[19] Shumei Liu; "Research of operating system virus defense strategy", Computer Science and Service System (CSSS), 2011 International Conference, 2011, pp. 3419-3421.

[20] A. Kieyzun, P. J. Guo, K. Jayaraman, and M. D. Ernst. "Automatic creation of SQL injection and cross-site scripting attacks", Proceedings of the 2009 IEEE 31st International Conference on Software Engineering, pp. 199-209, 2009.

[21] C. Cachin and S. Tessaro. "Optimal resilience for erasure-coded Byzantine distributed storage." Distributed Computing, pp. 497-498, 2005.

[22] C. Cowan, P. Wagle, C. Pu, S. Beattie and J. Walpole. "Buffer Overflows: Attacks and Defenses for the Vulnerability of the Decade," oasis, pp.227, Foundations of Intrusion Tolerant Systems (OASIS'03), 2003.

[23] E. Levy and I. Arce. "New threats and attacks on the world wide web." IEEE Security & Privacy, pp. 234-266, 2006

[24] Callegati, W. Cerroni, and M. Ramilli. "Man-in-the-Middle Attack to the HTTPS Protocol," IEEE Security and Privacy, vol. 7, no. 1, pp. 78-81, Jan./Feb. 2009, doi:10.1109/MSP.2009.12

[25] I. M. Abbadi and M. Alawneh. "Replay Attack of Dynamic Rights within an Authorised Domain," securware, pp.148-154, 2009 Third International Conference on Emerging Security Information, Systems and Technologies, 2009.

[26] J. Antunes, N. F. Neves, and P. J. Ver. "Detection and Prediction of Resource-Exhaustion Vulnerabilities" issre, pp.87-96, 2008 19th International Symposium on Software Reliability Engineering, 2008.

[27] J. Lee, M. Tehranipoor, C. Patel and J. Plusquellic. "Securing Designs against Scan-Based Side-Channel Attacks." IEEE transactions on dependable and secure computing, vol. 4, no. 4, pp. 325-336, 2007.

[28] K., Driscoll, B. Hall, H. Sivencrona, and P. Zumsteg. "Byzantine fault tolerance, from theory to reality." Computer Safety, Reliability, and Security, vol. 2788, pp. 235-248, 2003, doi: 10.1007/b12002

[29] M. D. Preda, M. Christodorescu, S. Jha, and S. Debray. "A semantics-based approach to malware detection.", ACM Transactions on Programming Languages and Systems (TOPLAS), vo. 30, no. 5, pp. 25, 2008

[30] M. F. Mergen, V. Uhlig, O. Krieger and J. Xenidis. "Virtualization for high-performance computing." ACM SIGOPS Operating Systems Review, vol. 40, no. 2, pp. 11, 2006.

[31] M. McIntosh and P. Austel. "XML signature element wrapping attacks and countermeasures", Proceedings of the 2005 workshop on secure web services, pp. 20-27, 2005.

[32] M. T. Louw, J. S. Lim, and V. N. Venkatakrishnan. "Enhancing web browser security against malware extensions." Journal in Computer Virology, vol. 4, no. 3, pp. 179-195, 2008.

[33] N. Gruschka and L. Iacono. "Vulnerable Cloud: SOAP Message Security Validation Revisited", IEEE International Conference on Web Services, pp. 625-631, 2009.

[34] Grobauer, B.; Walloschek, T.; Stocker, E.; "Understanding Cloud Computing Vulnerabilities" Vol: 9 Issue:2,pp. 50 - 57 ,2011.

[35] Jeffrey R. Jones, "Estimating Software Vulnerabilities," IEEE Security & Privacy, vol. 5, no. 4, 2007, pp. 28-32

[36] R. Kompella, S. Singh and G Varghese. "On Scalable Attack Detection in the Network." IEEE/ACM TRANSACTIONS ON NETWORKING vol. 15, no. 1, 2007.

[37] R. Syahputri and M. Hasibuan. "Security in Wireless LAN Attacks and Countermeasures", SNATI, pp.54-78, 2009.

[38] S. C. Wang, K. Q. Yan, S. S. Wang and C. P. Huang. "Achieving high efficient agreement with malicious faulty nodes on a cloud computing environment", Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human, pp. 468-473, 2009.

[39] S. King and P. Chen. "SubVirt: Implementing malware with virtual machines", IEEE Symposium on Security, pp. 1-14, 2006

[40] W. Liu. "Research on DoS Attack and Detection Programming", IITA, pp. 207-210, 2009.

[41] W. Speirs. "Making the kernel responsible: a new approach to detecting & preventing buffer overflows", Proceedings of the Third IEEE International Workshop on Information Assurance, pp. 21-32, 2005.

[42] X, Fu and K. Qian. "SAFELI: SQL injection scanner using symbolic execution", Proceedings of the 2008 workshop on Testing, analysis, and verification of web services and applications, pp. 34-39, 2008.

[43] Z. Chen and X. Yan. "Hardware Solution for Detection and Prevention of Buffer Overflow Attacks in CPU Micro-architecture." RESEARCH AND PROGRESS OF SSE, vol. 26, no. 2 pp. 214-219, 2006.

[44] F. Leu and Z. Li. "Detecting DoS and DDoS Attacks by Using an Intrusion Detection and Remote Prevention System", IEEE Conference and Exposition, pp. 1-15, 2009.

[45] S., Subashini, V. Kavitha. "A survey on security issues in service delivery models of cloud computing". Journal of Network and Computer Applications, vol.34, pp.1-11, 2011.

[46] S., Brohi, M., Bamiah, "Challenges and Benefits for Adopting the Paradigm of Cloud Computing", International Journal of Advanced Engineering Sciences and Technologies (IJAEST), vol. 8, pp. 286 - 290, 2011.

[47] A. Ramachandran S. Ramachandran, "Rapid and Proactive Approach on Exploration of Database Vulnerabilities", International Journal on Computer Science and Engineering vol. 4, pp. 224 - 234, 2011.