Volume 4, Issue 2, February 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Distributed Cloud Multi Tier Intrusion Detection System

K.Vignesh Kumar*, M.P.Ramasubash

Computer Science and Engineering,
SriGuru Institute of Technology,
Coimbatore, Tamilnadu, India

Abstract— Cloud computing refers to the deliverance of computing resources over the Internet. In its place of holding data in our own hard drive or updating applications for our requirements, you utilize a service over the Internet, at a different location, to accumulate our data or utilize its applications. By doing this, provide rise to certain confidentiality implications. Cloud stores almost infinite amount of data using virtualization. There may be a possibility of data outflow in cloud. So by preventing the data from unsigned users called intruders, we need to prevent the access from them so, we go for a technique called multi tier intrusion detection system. Providing security to distributed systems needs more than user certification with passwords or digital certificates and privacy in transmission of data. Distributed architecture of cloud makes the vulnerable and flat to complicated distributed intrusion attacks similar to Cross Site Scripting as well as Distributed Denial of Service. To hold large scale network access traffic and executive control of information and application in cloud, new multi-threaded distributed cloud IDS architecture has been projected. Our projected cloud IDS holds huge flow of data packets, examine them and produce reports competently by integrating information and performance study to become aware of intrusions.

Keywords— Cloud Computing, Multi Tier Application, Intruder Detection System

I. INTRODUCTION

Cloud computing is the deliverance of computing services in the Internet. Cloud services let individuals and businesses to utilize software and hardware that are supervised by third parties at remote locations. Examples for cloud include social networking, e-mail, and online file storage [1]. The cloud computing model allows access the data and computer resources from anywhere that a network connection is accessible. Cloud computing Provides a shared group of resources, including data storage pool, networks, computer processing authority, and specialized corporate and user applications [1]. Due to their ubiquitous use of personal or corporate information, clouds have always been the chance of attacks. These attacks have recently become more assorted, as interest has been shifted from attacking the front end to exploit vulnerabilities of the cloud.

In this paper, we present Multi tier IDS, a system utilized to sense the attacks in cloud. Our idea can create familiarity models of isolated user sessions. To reach this, we utilize a lightweight virtualization technique to allot each user's session to a faithful container, an remote virtual computing environment. We utilize the container ID to accurately associate the request with the successive queries. Thus, Multi tier IDS can construct a fundamental mapping profile by taking both the provider and user in the account [2]. To address the challenge while constructing a mapping representation for cloud, we first generated an individual training model for the basic operations given by the provider. We demonstrate that this idea works good in practice by using traffic from a live blog where we gradually modeled nine operations. Our results show that we can able to recognize all attacks, including more than 99 percent of the normal traffic as the training model is developed.

II. RELATED WORK

An Intrusion Detection System is of two types: anomaly detection and misuse detection. Anomaly detection first needs the IDS to recognize and describe the accurate and acceptable static form and dynamic behaviour of the system, which can then be utilized to detect irregular changes or anomalous behaviours [3] [4].

The edge between adequate and anomalous forms of stored code and information is accurately definable. Performance models are constructed by performing a statistical study on past data [5], or by using rule-based approaches to state behavior patterns [6]. An anomaly detector compares real usage patterns against traditional models to spot abnormal events. Our detection idea belongs to anomaly detection, and user depends on a training stage to construct the accurate model. As some justifiable updates may cause model drift, there is more number of approaches that are trying to resolve this difficulty. Our detection may possibly run into the same trouble; in this case, our model should be retrained from each shift. Intrusion alerts link [7] provides a set of resources that transform intrusion detection sensor alerts into concise intrusion reports in order to minimize the number of pretend alerts, false alerts, and irrelevant alerts.

It might fuse the alerts from various levels for describing a single attack, with the objective of producing a concise overview of security-related action on the system. It focuses mainly on abstracting the low-level sensor alerts and provides the complex, logical, high-level alert proceedings to the users. Multi tier IDS differs from this kind of approach

that correlates alerts from independent IDSs. Fairly, Double-Guard operates on several feeds of system traffic using a Single IDS that looks across the sessions to generate an alert without correlating or sum up the alerts generated by other independent IDSs.

An IDS [8] also utilizes temporal data to sense intrusions. Double Guard, still, does not correlate events on a time source, which runs the threat of wrongly considering independent but parallel events as correlated events. Multi tier IDS does not have such a constraint as it utilizes the container ID for every session to causally map the associated events, whether they may be synchronized or not. Some earlier approaches have detected intrusions by statically analyzing the source code. In Multi tier IDS, the new container-based structural design enables us to divide the different data flows by every session. This provides a means of tracking the data flow for every session. Our idea does not need us to examine the source code or know the application logic[2].

III. SECURITY ISSUES IN CLOUD COMPUTING

Security issues that can arise in cloud computing are

A. *Cloud Data Confidentiality*

Confidentiality of data in cloud is one of the conspicuous security concerns. Encrypting the data can be done with the conventional methods. Though, encrypted data can be protected from a malevolent user but the confidentiality of data even from the proprietor of data at service provider's end could never be unseen. Penetrating and indexing on encrypted information still remains a point of concern in that situation. Above mentioned cloud security problems are a little and dynamicity of cloud structural design are facing new challenges with quick execution of new service prototype.

B. *Cloud Security Auditing*

Cloud auditing is a complicated assignment to check conformity of all the security policies by the purveyor. Cloud service provider has the have power over responsive user information and processes, so an computerized or third party auditing method for forensic investigation and data integrity check is required. Confidentiality of data from third party auditor is another problem of cloud security.

IV. MULTI TIER IDS WITH INCIDENT RESPONSE

A. *Monitoring and Incident Response*

The ability to monitor logs change when computing resources are moved to a Cloud environment. Since incident response relies heavily on log data for detection and forensics, an organization's incident response plan will need to be modified.

1) Monitoring

One of the potential draw backs of moving data processing to a Cloud environment is losing direct control. Administrators who previously had direct access to physical servers and console access now have limited accessibility. Organizations are accountable for the risk incurred by use of services provided by external providers and address this risk by implementing compensating control. The challenge has become to work with the vendor to ensure adequate monitoring tools are in place to capture logs from operating systems, applications and hardware devices. These logs often contain crucial information if a system is compromised. From a security perspective, access to log information is important to being proactive in detecting malicious activity. Unfortunately, depending upon the service offering (i.e. Iaas, PaaS, SaaS) only some of the necessary logs will be available.

Obtaining IaaS logs are the most easily obtained Security, application, or system logs from a Windows server and sys log output from a Linux server can be captured as if the servers were housed locally. PaaS and SaaS log retrieval may be more difficult. When organizations opted to public cloud computing, the main theme of application logging will raise, since in SaaS and PaaS environments recognizable OS logs merely not exist. forlornly, organizations at present are having trouble analyzing application logs from conventional on building applications, even without the entire cloud feature blended in. The issue is further complicated by the fact that logs are aggregated in a multi-tenant environment and not shared by the CSP. In contrast to a traditional data center where the administrator has direct access to Windows, Linux, or syslogs, logs gathered in a Cloud environment may be a combination of various different customers combined into one log. In an article about SLAs by Buck and Hanf, SLAs need to detail the exact logs available.

Monitoring at the OS or VM level is a basic means to monitor your systems but to Closely monitor attacks tools such as an Intrusion Detection System (IDS) can be used. There are more considerations when implementing virtual IDS in a virtual environment. For example, because of an internal virtual network, it is more difficult to place a traditional IDS appliance in-line into a virtual environment. Host based IDS will function in a virtual world but an agent based host IDS will consume resources from the resource pool. If there are numerous host based IDS (HIDS) installed, then performance will be affected.

While the design to implement virtual IDS may be more challenging, it is possible to continue to use traditional security tools. According to SANS, there are three methods to allow intrusion detection monitoring in a virtual environment. Enable promiscuous mode on a Port Group or vSwitch. A virtual ID will be able to monitor traffic on the virtual network segment. In an IaaS environment, install a virtual appliance in-line. Utilize SPAN technology mirror a port to capture traffic.

A CSP may offer the customer the capability to monitor for malicious traffic by using and IDS. VMware expert Dave Schackleford, outlines several factors to be aware of if you implement your own IDS in an IaaS environment:

1) Make sure you can adequately monitor network traffic using "virtual taps" or port mirroring.
2) When using HIDS, be wary of resource consumption.
3) Consider how the IDS will be monitored. It may be necessary to connect monitoring consoles to the Cloud via a VPN connection.

2) Incident Response

The nature of incident response will be impacted when services are moved to the Cloud. In accordance to the Cloud Security coalition, the customer must consider what must be done to enable efficient and effective handling of security incidents in the Cloud. Given the possibility that log information may be directly inaccessible to the customer, the incident response team will need to take into consideration the type of service being utilized (i.e. IaaS, PaaS, SaaS) and craft a security SLA to address responsibilities of the CSP. For example, if SaaS is being utilized, then the CSP incident response team will internally respond to triggers from their Security Incident and Event Manager (SIEM), IPS/IDS tools or other log management tools.

In this scenario, the customer has no responsibility. However, it is important to include a notification process in the SLA, especially if personal information is at risk. If PaaS is being utilized, then the incident response team will have access to application logs but the CSP will still maintain server logs. The customer has more opportunity to retrieve log information from a PaaS provider by communicating to the CSP what triggers an event. Examples of triggers can be failed authentication attempts or application errors. If the service provided is IaaS, then the CSP is responsible for the infra-structure related logs such as storage, networks and hypervisors. The customer will have access to their VM logs and IDS logs during an incident. Services such as SaaS or PaaS may make incident response easier because the burden rests upon the CSP. Incidents that require obtaining an image or snapshot of the virtual machine for forensics is also easier because a virtualized environment is designed to copy or clone images, including memory states. Special software is no longer needed when the inherent capability of your virtual platform provides these functions. Procedures of the incident response team will need to be modified to accommodate the new environment.

Cloud computing provides application and storage services on distant servers. The clients do not need to worry about the maintenance and software or hardware updating. Cloud model works on the idea of virtualization of various resources, where a virtual machine monitor (VMM) server in cloud data center hosts a number of clients on one physical system.

Intrusion detection system works as a key function in the security and insistence of dynamic resistance architecture against intruder hostile attacks for any business and IT firm. IDS performance in cloud computing needs a well-organized, scalable and virtualization-based system approach. In cloud computing, user information and application is uploaded on cloud service provider's distant servers and cloud user have a partial control over those information and resources. In this situation, the administration of IDS in cloud becomes the liability of cloud provider. Even though the bureaucrat of cloud IDS should be the user and not the provider of cloud services. So by Deploying HIDS in VMM or host machine would permit the administrator to supervise the VMM and virtual machines on that VMM. But with the quick flow of high quantity of information as in cloud model, there would be difficulties in performance like congestion of VM hosting IDS and dropping of data packets.

Also if host is compromised by an aberrant attack the HIDS working on that host that would be neutralized. In this situation, a network based IDS would be more appropriate for deployment in cloud like infrastructure. NIDS would be located outside the VM servers on narrow network points such as router, gateway or switch for network traffic monitoring to have a large-scale view of the system. Such NIDS would still be facing the problem of huge amount of information through network access rate in cloud infrastructure. To handle huge number of data packets flow in the cloud environment a multi-threaded IDS idea has been projected in this paper.

The multi-threaded IDS can be able to process huge amount of information and could minimize the packet loss. After a proficient processing the proposed idea of IDS would pass the monitored alerts to the third party monitoring service, which would in turn directly notify the cloud user about their system under attack. The third party monitoring service would also grant specialist suggestion to cloud service provider for bad configurations and intrusion loop holes in the system. Figure 1, shows the proposed IDS model [9]. The cloud user makes use of the data on remote servers at service provider's location over the cloud system. User needs and actions are monitored and logged through a multi-threaded NIDS. The alert logs are eagerly communicated to cloud user with an specialist suggestion for cloud service provider.

B. *Capture & Queuing*

The capture module, accepts the in-bound and out-bound (ICMP, TCP, IP, UDP) data packets. The captured data packets are passed to the shared queue for next process called analysis.

C. *Analysis or Processing*

The analysis and process module accepts data packets from the shared queue and analyze it against signature base and a predetermined rule set. Each action in a shared queue can have more number of threads which work in a shared fashion to develop the system performance. The main process will accept TCP, IP, UDP and ICMP packets and several threads would parallel process and match those packets against predetermined rule sets. Through a competent matching and analysis the worst packets would be recognized and alerts been generated.

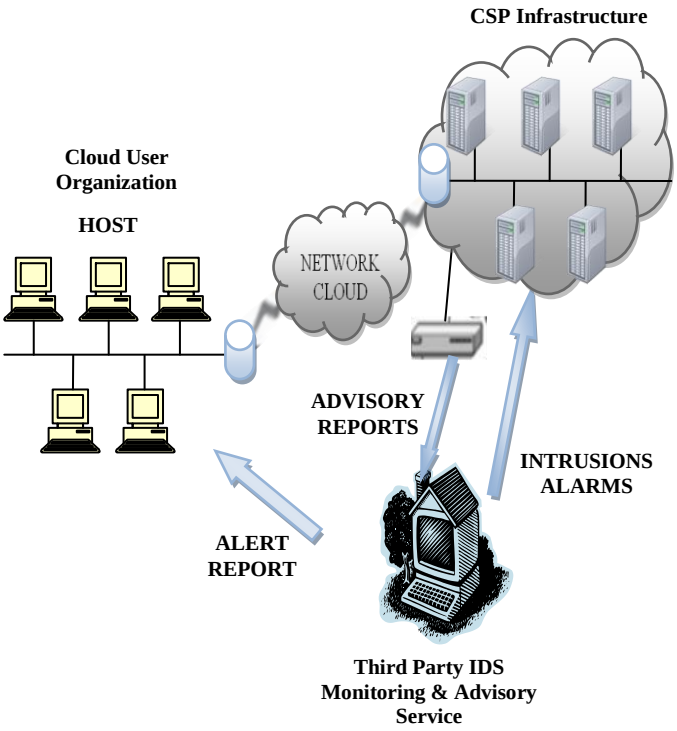

Fig 1 Architecture of Distributed multi tier IDS

Our proposed multi-threaded NIDS representation for distributed cloud infrastructure is the base of

D. *Alert*

Alert module would read the alerts from shared queue and prepares alert reports. The third party monitoring and suggestion service having knowledge and resources would instantly make a report for cloud user's data and sends a comprehensive expert suggestion report for cloud service provider. Fig 2 shows the flow chart of proposed multi-threaded Cloud IDS [9].

The action flow of the proposed system receives input packets from ICMP, IP, UDP and TCP. Then a multithreaded queue is implemented to parallelize the tickets as well as it checks for rule set matching this makes the decision to allow the packets to utilize cloud. In such case any intruder entry detects the intrusion alarm notifies the user to prevent against them. If rule set matches then it allows to utilize the cloud by cloud user and the cloud service provider must authenticate the user to utilize the cloud.IDS contains some unique rule set which determines the intruder entry. The multi threaded queue is very much useful that allows more number of data through the queue and it increases the speed of data processing in cloud

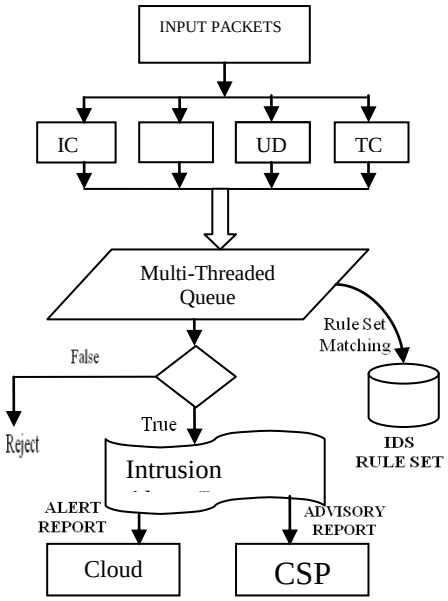

Fig. 2 Flow Chart for Multi tier IDS

V. ADVANTAGES OF PROPOSED MODEL

- High quantity of data in cloud environment could be handled by a single node IDS through a multi-threaded idea.
- CPU, memory utilization as well as packet loss would be minimized to improve the overall effectiveness of cloud IDS.
- In a host based IDS (HIDS) situation, if host becomes the sufferer of aberrant attacker and handled by the intruder, HIDS on that particular host would be compromised. In such a scenario the attacker would not let HIDS to send alerts to administrator and could play havoc with the information and applications. For better security and resistance, network IDS (NIDS) has been projected for cloud infrastructure.
- A third party monitoring and suggestion service has been proposed, who has both knowledge and resources to handle intrusion data and generate alerts for cloud users as well as suggested reports for cloud service provider.
- Being at a mid point, proposed Cloud IDS would be able to carry out parallel processing of data analysis, which is an efficient idea.

VI. CONCLUSION

Cloud computing helps to store enormous amount of data over the internet. Hence there may be probability of intrusion is more with the sophistication of intruder's attacks. Various IDS methods are used to counter malicious attacks in conventional networks. For Cloud computing, massive network access rate, relinquishing the control of information & applications to cloud service provider and distributed attacks vulnerability, an competent, trustworthy and information translucent IDS is necessary. In this account, a multi-threaded cloud IDS architecture is compiled which can be administered by a third party monitoring system for a better optimized effectiveness and precision for the cloud user.

ACKNOWLDGEMENT

The authors would like to thank the staffs and students of SriGuru Institute of technology for their valuable support and guidance.

REFERENCES

[1] Bill Williams, "The Economics of Cloud Computing: An Overview for Decision Makers", Jul 30, 2012.
[2] Meixing Le, Angelos Stavrou and Brent ByungHoon Kang, "Double Guard: Detecting Intrusions in Multitier Web Applications", IEEE transactions on dependable and secure computing, vol. 9, no. 4, July/august 2012
[3] H. Debar, M. Dacier, and A. Wespi, "Towards a Taxonomy of Intrusion-Detection Systems," Computer Networks, vol. 31, no. 9, pp. 805-822, 1999.
[4] T. Verwoerd and R. Hunt, "Intrusion Detection Techniques and Approaches," Computer Comm., vol. 25, no. 15, pp. 1356-1365, 2002.
[5] C. Kruegel and G. Vigna, "Anomaly Detection of Web-Based Attacks," Proc. 10th ACM Conf. Computer and Comm. Security(CCS '03), Oct. 2003.
[6] Brian Caswell, Jay Beale, "Andrew Baker Snort Intrusion Detection and Prevention Toolkit"By ISBN-10: 1597490997 February 2007.
[7] F. Valeur, G. Vigna, C. Kru¨ gel, and R.A. Kemmerer, "A Comprehensive Approach to Intrusion Detection Alert Correlation," IEEE Trans. Dependable and Secure Computing, vol. 1, no. 3, pp. 146-169, July-Sept. 2004
[8] A. Seleznyov and S. Puuronen, "Anomaly Intrusion Detection Systems: Handling Temporal Relations between Events," Proc. Int'l Symp. Recent Advances in Intrusion Detection (RAID'99), 1999.
[9] Irfan Gul, M. Hussain, "Distributed cloud intrusion detection model", International Journal of Advanced Science and Technology Vol. 34, September, 2011.

