IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 4, April 2018

Securing Cloud from Intrusion Attacks using Intrusion Detection System in Virtual Machine Environment

Prof. Mr. M. K. Nivangune[1], Ms. Kanchan Bairagi[2], Ms. Archita Kotkar[3], Mr. Vishnukant Salunke[4]

Department of Information Technology, Sinhgad Academy of Engineering, Pune, India[1,2,3,4]

Abstract: In a public cloud computing environment, consumers cannot always just depend on the cloud provider's security infrastructure. They may need to monitor and protect their virtual existence by implementing their own intrusion detection capabilities along with other security technologies within the cloud fabric. Traditional Intrusion Detection Systems (IDSeS) are not suitable for these environments due to their openness and limited essence. Intrusion Detection System as a Service (IDSaaS), monitors a system or a network for illegal activities in public cloud (IaaS) by providing intrusion detection technology that is portable, secure and fully controlled by the cloud service providers (CSPs). A prototype of IDS is described.

Keywords: Security; Cloud Computing; Intrusion Detection System, Handling bulky attacks.

I. INTRODUCTION

Considering today's scenario, security is the major concern in vast and fast growing computer networks. We have heard of various data breaches due to poor and limited essence of an Intrusion Detection Systems (IDSeS). Hence a strong Intrusion Detection System is the essential task in day-to-day life practices. There are various approaches being used for intrusion detection. In this paper, we present an Intrusion Detection System model to be deployed using K-Mean Algorithm for determining various types of intrusions or attacks. We offer accuracy and effectiveness of an approach to generate rules for different types of anomalous attacks. KDDCUP99 dataset is being used for setting standards and rule generation. The KDDCUP99 training and testing dataset has a standard set of data to be audited and includes signatures for wide variety of intrusion attacks. An age-old concern for every educational institution is how to provide the student with the best environment for learning. For teaching network security related courses, the network environment could be built using a scenario of intrusion detection system (IDS). This approach provides students with an actual network to carry out experiments; however, the equipment is expensive and it is time consuming to actually set up all of the network devices, data, and communication. Instead of using extra tools, virtualization technology is employed to build the network with multiple virtual machines. Within a single physical host machine, multiple virtual machines are created and operated simultaneously. In each virtual machine, applications and services are implemented and the virtual machine executes the code just as a physical machine would. This approach eases the load of network administration as mistakes can be easily fixed while the network stays up and running. When a network change is required to conduct desired intrusion or attack experiments, it can be easily reconfigured in a virtual environment.

II. LITERATURE REVIEW

We have many IDSeS available in open space, but here our analysis is limited to two HIDS open-source tools; OSSEC and Tripwire.

A.OSSEC:

OSSEC is an Open Source, Host-based Intrusion Detection System (HIDS) that performs log analysis, file integrity checking, Windows registry monitoring, Unix-based root kit detection, real-time alerting and active response. It runs on most operating systems, including Linux, Mac OS, Solaris, Open BSD, FreeBSD, HP-UX, AIX and Windows. The OSSEC HIDS can be installed as a stand-alone tool to monitor one host or can be deployed in a multi-host scenario, one installation being the server and the others as agents. The server and agents communicate securely using encryption. OSSEC also has ability to react to specific events or set of events by using commands and active responses. OSSEC consists of a main application, a Windows agent, and a web interface. Main Application is required for distributed network or stand-alone installations. It is supported by Linux, Solaris, BSD, and Mac environments. Windows Agent is provided for Microsoft Windows environments. The main application needs to be installed and configured for server mode to support the Windows Agent. Web Interface provides a graphical user interface[4].

DOI 10.17148/IJARCCE.2018.7429

OSSEC Architecture
OSSEC is composed of multiple sections. It monitors the information being provided from agents, syslog, databases and from agent less devices using central manager utility. It stores the file integrity by checking databases, the logs, events and system auditing entries. OSSEC Agent is a small program or collection of programs installed on the systems which are need to be monitor. The agent will collect information in real time and forward it to the manager for analysis and correlation.

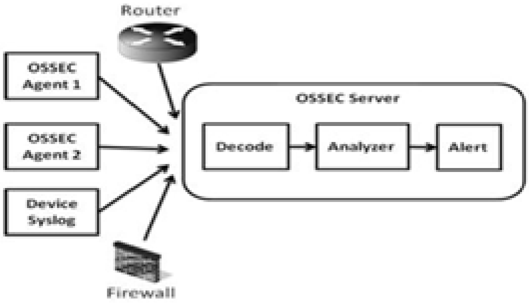

Fig. 1 Architecture of OSSEC

Analyse logs from multiple devices and formats. An active response system. But in OSSEC, it is more difficult to upgrade between versions; OSSEC comes with default rules and they get overwritten on every upgrade.

B. Tripwire:
Open Source Tripwire is an HIDS for monitoring and alerts user on Specific file change(s) in network. Tripwire monitors Linux system to detect and report any unauthorized changes to the files, directories. A baseline is created to monitor and detect as to which file is added/modified, what are the changes etc. It can also serve many other purposes, such as integrity assurance, change Management, and policy compliance [4].
Tripwire encrypts its database and configuration files by storing hashes instead of storing contents. Tripwire encrypts its database and configuration files by storing hashes instead of storing contents. But it does not generate real-time alerts upon an intrusion occurs. Tripwire is only suitable for monitoring small number of Linux servers where centralized control and reporting is not required.

III. PROPOSED MODEL

IDSaaS ARCHITECTURE
IDSaaS, as shown in Figure 2, consists of five main components: the Intrusion Engine, the Output Processor, the Events Database, the Alerts Management, and the Rule-set Manager.
Figure 2. IDSaaS Components [1]

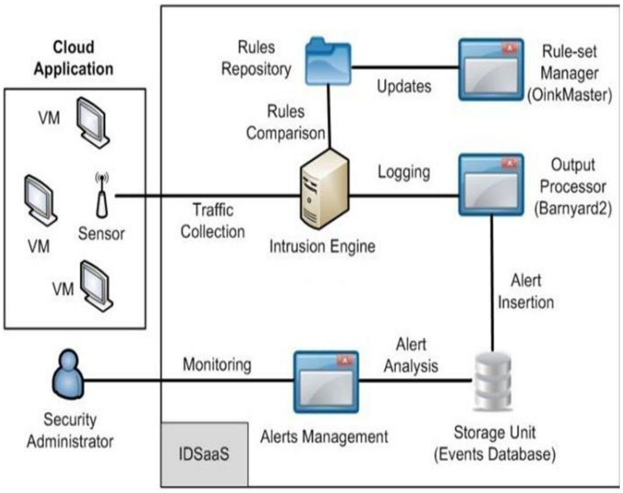

Fig.2 Proposed system architecture

A. Intrusion Engine: Initially, the sensor taps into the network and collects network packets, which are decoded for the analysis step. The Intrusion Engine is the brain of the system. It pre-processes the incoming packets and examines their payload section looking for any matching pattern of a threat defined in the loaded attacking rules. The processed packet is logged only if it matches a rule. The output binary file is a collection of captured alerts. The signature-based detection model is selected because of its suitability to the cloud environment. Simplicity, flexibility and ease of sharing signatures are some of the advantages of this approach. Also, it will enforce the elasticity feature by eliminating the learning time for the system's behaviour required for the anomaly-based approach.

B. Output Processor: The main purpose of the Output Processor is to increase the performance of the intrusion engine by formatting the output log files and inserting them into the Events Database. This allows the intrusion engine to focus on processing network packets and logging alerts while leaving the relatively slow process of database insertion to the Output Processor component.

C. Events Database: The Events Database stores the formatted events generated from the Output Processer component. Also, the database stores other relative information like sensor ID, event timestamp and packet payload details.

D. Alert Management: The Alert Management component is used as a GUI tool to view the generated alerts and correlate them. It allows the security administrator to extract events and relate them to predefined attacking situations. Moreover, it provides the ability to generate reports based on time, source of the attack, or types of threat.

E. Rule-set Manager: IDSaaS is a rule-based IDS system, and its rule base has to be updated frequently to comprise the new threats and attacking scenarios. The Rule-set Manager automatically downloads the most up-to-date set of rules from multiple locations. Rules are generally obtained either for free from the public community service or through a subscription service such as the Source fire VRT[1].

IV. SYSTEM OVERVIEW

Intrusion Detection System

An intrusion detection system monitors computer systems, and looks for signs of intrusion (unauthorized users) or misuse (authorized users overstepping their bounds). Intrusion detection systems help computer systems prepare for and deal with attacks. It provides security by means of gathering and analysing the information from various system and network sources for symptoms of security problems.

A. General Overview of IDS System with Network Following diagram shows the system architecture of IDS:

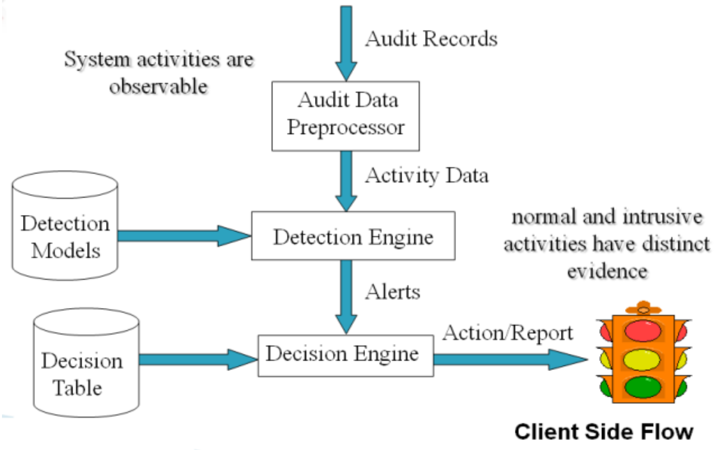

Fig. 3 System Architecture

The first, most commonly method is to compare these information to large databases of attack signatures, each reflecting an attempt to bypass or nullify security protections. The second is that it looks for problems related to authorized users overstepping their permissions (e.g., a clerk searching executive payroll records).Some intrusion detection systems perform statistical analysis on the information, looking for patterns of abnormal activity that might not fall into the prior two categories (e.g., accesses that occur at strange times, or an unusual number of failed logins). In some cases, intrusion detection systems allow the user to specify real-time responses to the violations. Intrusion detection systems can perform a variety of functions:

1. Monitoring and analysis of user and system activity
2. Auditing of system configurations and vulnerabilities

3. Assessing the integrity of critical system and data files
4. Recognition of activity patterns reflecting known attacks
5. Statistical analysis for abnormal activity patterns

Some Intrusion detection systems can monitor the operation of firewalls, encrypting routers, key management servers and files critical to other security mechanisms provide additional layers of protection to a secured system.

B. Real Time Use of IDS

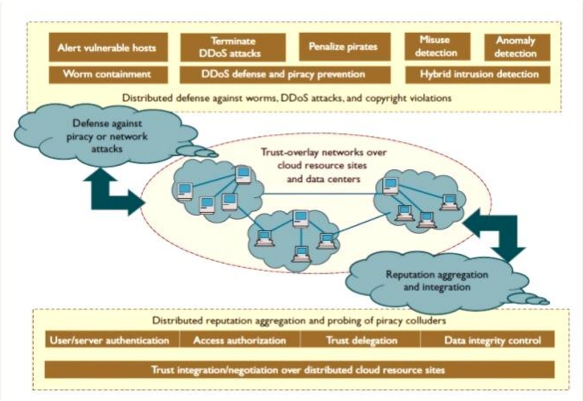
Alert vulnerable hosts
Terminate DDoS attacks
Penalize pirates
Misuse detection
Anomaly detection
Worm containment
DDoS defense and piracy prevention
Hybrid intrusion detection
Distributed defense against worms, DDoS attacks, and copyright violations
Defense against piracy or network attacks
Trust-overlay networks over cloud resource sites and data centers
Reputation aggregation and integration
Distributed reputation aggregation and probing of piracy colluders
User/server authentication
Access authorization
Trust delegation
Data integrity control
Trust integration/negotiation over distributed cloud resource sites

Fig. 4: Real time use of IDS\

C. Firewall versus Intrusion Detection System

Protecting a full time Internet connected system is becoming more important than ever. A growing number of software based personal firewall products are readily available and can be installed. An evaluation of needs should be conducted before selecting a product as concept, method and features vary. Firewalls act as a barrier between internal local networks and the outside world (Internet), and filter incoming traffic and screen unwanted services according to a security policy. However, there are limitations to what a firewall can do and it is no longer sufficient to combat intrusion. The firewall is the security equivalent of a security fence around your property and the guard post at the front gate. It can keep the most unwanted characters out, but cannot necessarily tell what is going on inside the compound. A commonly raised question is how the intrusion detection complements firewalls. Intrusion detection systems are the equivalent of multi-sensor video monitoring and burglar alert systems. They centralize this information; analyse it for patterns of suspicious Behaviour in much the same way a guard at a monitoring post watches the feeds from Security cameras, and in some cases, deals with problems they detect. Most loss due to computer security incidents is still due to insider abuse. Intrusion detection systems, not firewalls, are capable of detecting this category of security violation. To enhance security, an intrusion detection system can be run against the connection. However, there are limited intrusion detection system that supports operating system (e.g. Windows 9x/ME) commonly used by home user.

D. Generic Intrusion Detection Model

There are many different Intrusion Detection Systems deployed world-wide and almost as any different designs for them. There are also several ideas in the literature about how to perform intrusion detection. Since there are so many different Intrusion Detection Systems, it helps to have a generic model within which to consider all of them.

The Common Intrusion Detection Framework (CIDF) defines a set of components that together define an intrusion detection system. These components include monitor (sensor) which generates events, analysis engines, storage mechanisms, and countermeasures. A CIDF component can be a software package in and of itself, or part of a larger system. Figure 4.1 shows the manner in which each of these components relates.

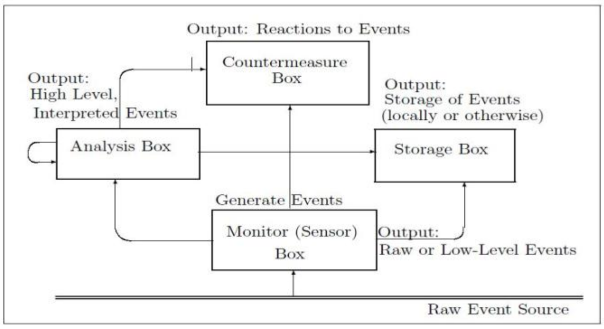
Output: Reactions to Events
Countermeasure Box
Output: High Level, Interpreted Events
Output: Storage of Events (locally or otherwise)
Analysis Box
Storage Box
Generate Events
Monitor (Sensor) Box
Output: Raw or Low-Level Events
Raw Event Source

Fig. 4 General Intrusion detection model

DOI 10.17148/IJARCCE.2018.7429

The purpose of a monitor (sensor) box is to provide information about events to the rest of the system. An "event" can be complex, or it can be a low-level network protocol occurrence. It need not be evidence of an intrusion in and of itself. Monitor boxes are the sensory organs of a complete intrusion detection system. Without it the intrusion detection system has no information from which to make conclusions about security events. Analysis box analyses input from monitor sensor which generates event. A large portion of intrusion detection research goes into creating new ways to analyse event streams to extract relevant information, the monitor sensor and analysis boxes can produce large quantities of data. These information should be made available to the rest of the system to be of any use. The data storage box component of an intrusion detection system defines the means used to store security information and make it available at a later time. Many intrusion detection systems are designed only as alarms. This allows an intrusion detection system to try to prevent further attacks from occurring after initial attacks are detected. Even systems that do not provide countermeasure capabilities can be connected into home-brewed response programs to achieve a similar effect.

V. SYSTEM FEATURES

Goal of our system is to make college lab secure and risk free that is our basic goal.
And we are going to implement our system into three layers that is different from traditional system.
There are 3 types of components

1. Users
2. Cloud Server.
3. IDS platform.

Our Project Flow will Look Like this:-

1. User Level

i. Registration of users into the cloud
ii. User profiles management
iii. Login Session Management
iv. Sharing resources, apps, data etc.

2. Client Level Management

i. Provide measures of security
ii. Monitor each activity, update threat signatures, etc.
iii. Give feedback / results

3. Administrator Level Management

i. Approval of the Client & User level communication using proper Security Algorithms
ii. Installing security plug ins for cloud
iii. Update total changes occurred with reports
iv. Handle bulky attacks & intruders in inter-system

VI. CONCLUSION

In this project when the user will upload the document (like image, text files, PDF, video, audio) the document will firstly scanned to find vulnerabilities in the file. While scanning the file signature matching will be done. Signature will be compared with the already available dataset. We have a dataset of KDDCUP99 which is having all signatures of viruses, Trojans, worms, malwares etc. If the signatures are matched then system will notify user that file contains malicious contents. System will abort the file uploading operation. But if in case the signatures are not matched with signatures in dataset then system will continue the operation and upload the file on cloud. While uploading the file onto server IDS will split the file. Then spitted file will be stored on server. While downloading the file spitted parts will combined by using pattern matching technique and original file will be obtained. Hence by using the above approach cloud data will be secured from Intrusion Attacks using Intrusion Detection System.

VIII. FUTURE WORK

To improve the usability of the IDS, the future work can be done as follows:
We can add Extra Functionality Additions like Cross Platform Security, Adding Compatibility for VPN. Addition of Security Measures(Enhancement for Wi-Fi). Scope can increased by Implementation of Chat Applications within the network. Retrieval of File Using Reverse Euclidean’s Approach And finally, a response mechanism can be introduced in order to stop intrusions before a failure occurs.

DOI 10.17148/IJARCCE.2018.7429

ACKNOWLEDGMENT

The authors are extremely thankful to **Sinhagad Academy of Engineering**, Pune for their invaluable time and suggestions towards the market survey for improvement of this paper.

REFERENCES

[1] Turki Alharkan & Patrick Martin ,”IDSaaS: Intrusion Detection System as a Service in Public Clouds”, IEEE transactions on dependable and secure computing, vol. 8, no.3, march-april 2015

[2] Hatem Hamad & Mahmoud Al-Hoby, “Managing Intrusion Detection as a Service in Cloud Networks “,International Journal of Computer Applications (0975 – 8887) Volume 41– No.1, March 2016

[3] Poonam Dabas* Rashmi Chaudhary , “Survey of Network Intrusion Detection Using K-Mean Algorithm “,International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 3, March 2013 ISSN: 2277 128X.

[4] Surya Bhagavan Ambati, Deepti Vidyarthi , “a brief study and comparison of, open source intrusion detection system tools“,International Journal of Advanced Computational Engineering and Networking, ISSN: 2320-2106, Volume-1, Issue-10, Dec-2013.