IDSaaS: Intrusion Detection System as a Service in Cloud

Pranali Mandaokar[1], Deepak Sharma[2]

[1] Student, Computer Engineering, K J Somaiya College of Engineering, Mumbai, India
[2] Professor, Computer Engineering , K J Somaiya College of Engineering, Mumbai, India

---***---

Abstract - ***In today's world as many people and organizations are moving towards cloud, without knowing the safety of their data. So, in such a scenario user can themself monitor their data on cloud by implementing a intrusion detection system. In this research a intrusion system was installed on the cloud which will help the user to monitor the unnecessary user to access the uses data. Here we propose a IDSaaS that targets the security at the infrastructure level of cloud. Here the Snort is used as a intrusion detection system, which is a open source software. Other software used are Barnyard2, Snorby, Onikmaster, Joomla.***

Key Words: *Cloud computing, intrusion detection system, vpc, public and private subnet.*

1. INTRODUCTION

As in today's world there is a constant fight regarding the space available, increasing the disk space is also not solving the purpose, hence most of the people are trying to move to the cloud to store their data. But how safe it is to store the data on the cloud is the question comes to every cloud user. After saving data on the cloud can the user rely on the security provided cloud. Is there a guarantee that the user data will not be misused for various reasons without the knowledge of the user [1].

In this paper we propose a safety mechanism for the cloud user that will be completely maintained and operated by the cloud user without anybody's interference.me Therefore, IDSaaS, will be used as a safety mechanism that will monitor and log all suspicious activities taking place in the network in between the virtual machines within a pre-defined virtual network in the cloud [2].

Through this work one can make a comparison between the overhead in the regular cloud setup and the proposed cloud setup.

2. IDSAAS IN CLOUD

The cloud users should not only depend on the security provided by the cloud providers, but be able to monitor and protect their own data in the cloud environment.

The following are the features IDSaaS provides to cloud users:-

- On-demand Elasticity: IDSaaS components are responsible for monitoring the packets flowing through the network based on the packet volume.
- Portability: The IDSaaS components are build using a collection of VM's (Virtual Machine) Hence the IDSaaS components can reside in a public or private cloud or in multiple regions within a single cloud.
- Fully controllable: IDSaaS's functionality and architecture is independent from the control of the cloud providers.
- Customizable Signature: IDSaaS has some predefined threats scenario's for faster and accurate threat detection.
- Reliability: IDSaaS has the feature of storing the alert for the future reference so that using the stored references there is faster recovery of data.

3. IDSAAS ARCHITECTURE

3.1 IDSaaS Components

There are five main components that makeup IDSaaS, they are as follows:

1. Intrusion Engine
2. Output processor
3. Event Database
4. Alert Management
5. Rule-set Manager

We will now look at each one of them in brief:

Intrusion Engine, is the core component of IDSaaS which is responsible for the incoming packet capturing and detecting this packet by comparing payload section of the packet to the rules that are stored in the Event Database. For this the a open source intrusion detection system called snort is used [3]

~~Output processor, is mainly used to help the intrusion~~ engine work smoothly, this is done by formatting the output log files and storing them into the Event Database. This can be achieved using the Branyard2 [4].

Event Database, this is used to store the updated events by the Output Processor.

Alert Management, is used to provide a easy access to the events taking place by providing a GUI tool. For this purpose we will be using the Snorby [5]

Rule-set Manager, will automatically download the most updated rules from different locations. Rules can be either obtained for free from public community or by a subscription. For this we will be using Onikmaster[6], is a simple perl script, which compares the locally stored packets with the incoming packets.

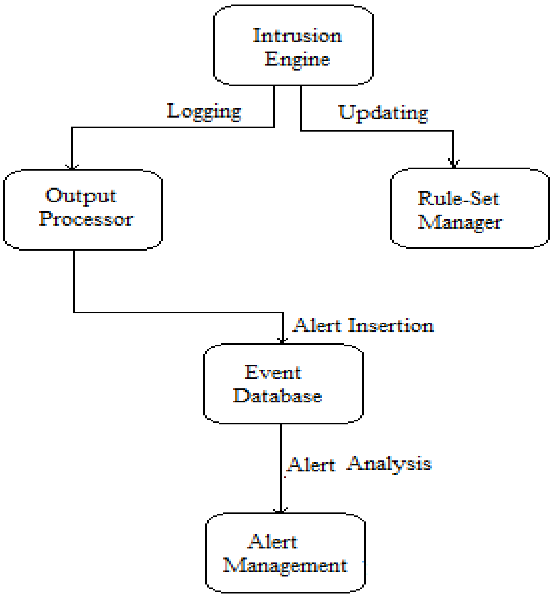

Fig 1: Block diagram to show the experiment setup of IDSaaS

4. Environmental setup

Environment will require a Virtual Private Cloud (VPC) with the public and the private network. The private subnet will contain the business application, that will remain protected here and all the traffic will have to flow from the public subnet. Any virtual machine that is kept in a private subnet will remain isolated from the cloud traffic except the traffic that is coming from a public subnet using a NAT.

4.1 IDSaaS VMs

IDSaaS manager, performs many administrative activities and is also used as a access point to configure both public and private subnets. The Event Database is also present inside the Manager VM. The alert Management component which monitors traffics is placed in the VPC.

IDS Core, is a VM that will act as a gatekeeper for the various business application residing in the private cloud. Using the intrusion system it will inspect all the incoming packets by comparing them to stored rules and a request can be stored to the business application VMs to allow or trap the threat packet by the IDS core VMs.

4.2 Security Group

Security group provides permission to the network services that are allowed to run on the cloud. These are nothing but the virtual firewalls, that will allow the incoming and the outgoing traffic based on the inbound and outbound allowance.

4.3 AppVM

This is the business application that is to be protected in this experiment. This will be present in the Private subnet of the VPC and it will consist of the Joomla [7] and the ProFTPd software [8], each one of these will be used as target for HTTP and FTP attacks respectively.

During the experiment we will make comparisons between our proposed IDSaaS and another without the IDSaaS.

Here the experimental setup without IDSaaS will consist of a VPC with public and the private subnet. Private subnet will contain the AppVM. Public subnet will contain a FwdVM, this is used so that all the traffic that want to make use of the services provided by AppVM will have to flow from the FwdVM in public subnet. As the VM kept in the private subnet cannot have direct access to the internet. There will be two attacking scenarios, one attacker i.e. the internal attacker will reside inside the cloud, but outside the VPC. The other one i.e. the external attacker will reside outside both the VPC as well as the cloud. Both the internal and the external attacker will be VMs.

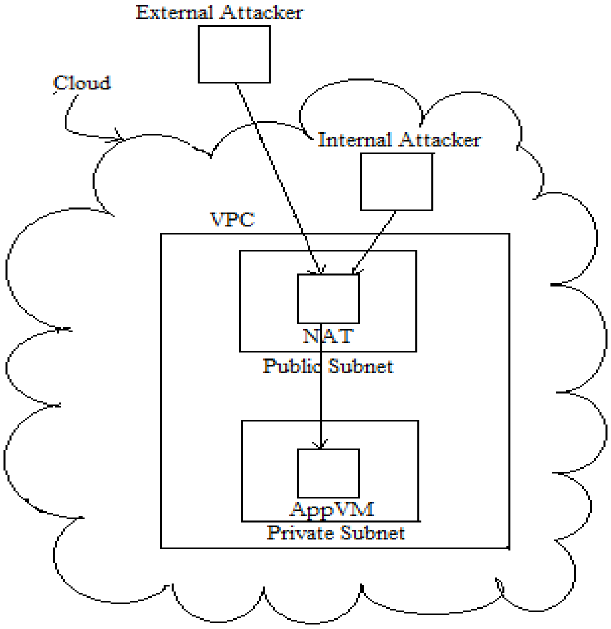

Fig 2: Experimental Setup without IDSaaS

In the second case where the proposed IDSaaS components IdsCoreVM and MgrVM used for monitoring and alert storing purpose respectively. Here two customized rules were written to capture the HTTP and FTP attacks. The goal for the HTTP rule is to alert the administrator for any traffic initiated from outside the VPC, aiming for the VPC network with the intension of accessing the user's information in the business application. FTP request an alert rule was included in the rule repository of the IDSaaS to monitor the file transfer session between the attacker and the business application. This rule will notify the administrator if the content of the uploaded file contains a match from a restricted keyword list (e.g. the name of a malware called sparkleg).

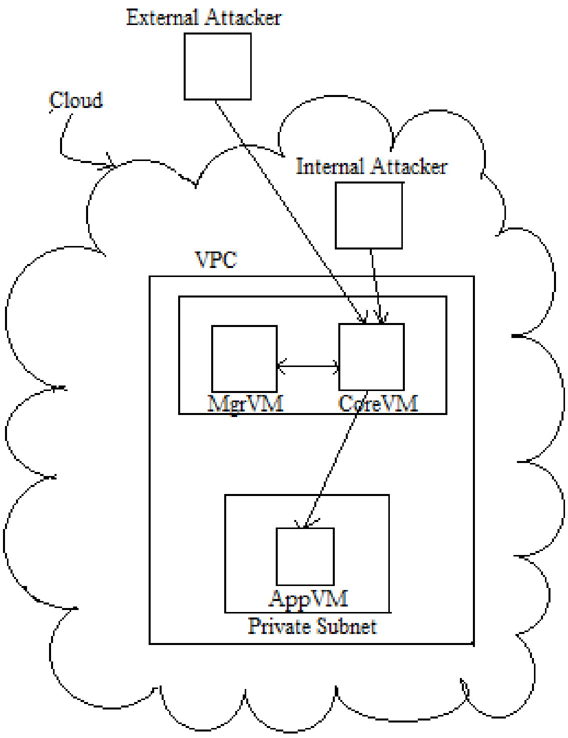

Fig 3: Experimental Setup with IDSaaS

We propose the above two experiment setup and conduct the following two evaluation to show the efficiency of the IDSaaS.

5. Experimental Evaluation

5.1 IDSaaS Components overhead Experiment.

Here we will conduct two experiments in order to test the effectiveness of the proposed IDSaaS this will be tested, by measuring the overhead that is added by the IDsaaS components that used to protect the business application that is stored in a cloud. Hence in this experiment will focus to measure the overhead that may be added due to the IDSaaS components.

5.2 IDSaaS Rules overhead Calculation

The attacking rules written for the scanning of the incoming packets may also affect the efficiency of the proposed IDSaaS that is used for capturing threats. Here we will see how this will affect the performance of the system.

6. CONCLUSIONS

In this paper proposes a IDSaaS that will help users to protect the virtual machines. IDSaaS is also compatible to the many features provided by the cloud like elasticity, portability, pay-per-use and on demand.

ACKNOWLEDGEMENT

This research is supported by Professor Deepak Sharma of K. J. Somiya College of Engineering, Vidhya Vihar, Mumbai.

REFERENCES

[1] C. Bruns, "Public cloud security remains MISSION IMPOSSIBLE", Network World. Oct, 2011.

[2] Amazon Elastic Compute Cloud (Amazon EC2) Available: http://aws.amazoncom/vpc

[3] Sourcefire, Snort (version 2.9.5) Available: http://www.snort.org.

[4] The Barnyard2 Project. Available: http://www.secrixlive.com/barnyard2.

[5] Snorby Available: http://www.snorby.org.

[6] The Oinkmaster Project Available: http://oinkmaster.sourceforge.net/download.shtml

[7] Joomla Content Management Software (Joomla v.1.7). http://www.joomla.org/announcements/release-news/5411-joomla-175-released.html.

[8] ProFTP v.1.3.4a. http://www.proftpd.org.