Security from various Intrusion Attacks using Honeypots in Cloud

Renu Meghani¹, Sanjay Sharma²
M.Tech Scholar¹, Assistant Professor²
Oriental Institute of Science and Technology, Bhopal, M.P., India.

Abstract — Cloud Computing means accessing the data from their own datacenters such that the chances of eavesdropping have been reduced and storage cost is reduced. Now here in this paper an efficient technique of intrusion detection is proposed using the concept of honeypots. The data to be send by the cloudlets is accessed by the brokers of the cloud where we have implemented some of the rules for the detection of intrusions. The proposed technique implemented here provides less false alarm ratio and provides more accuracy for the detection of intrusions.

I. INTRODUCTION

Now-a-day, more and more people are using internet all over the world. Most of the collected data are valuable since any traffic come to the honeypot/net is suspicious. The honeypots varies in the interaction they provide to the attackers, from the low interaction to medium and high, each type has its advantages and disadvantages. [1]

The aim of the honeypot is analyzing, understanding, watching and tracking hacker’s behaviors in order to create more secure systems. Honeypot is enormous method to get better network security administrators’ knowledge and learn how to get information from a victim system using forensic tools. "A honeypot is a closely monitored computing resource that we want to be probed, attacked, or compromised. More precisely, a honeypot is “an information system resource whose value lies in unauthorized or illicit use of that resource" [2]. A significant amount of work is available that details the benefits of honeypots [12], [4]. Other papers go into some detail about the strategic considerations involved when using honeypots [12]. There are also papers that describe specific applications of honeypots as building blocks for a system such as a honeycomb, which is used to create intrusion detection signatures [11].

Honeypots are divided into two general categories: production honeypots as well as research honeypots. Production honeypots add value to the security of a specific organization and help moderate risk, and are characteristically implemented within an organization as they help in detecting attacks.

Production honeypots can be deployed easily, since such type of honeypots require less functionality. This type of honeypots gives less information about the attackers than research honeypots. Research honeypots requires gain information. The primary goal is to research the threats that organizations may face, such as who the attackers are, how they are organized, and what kind of tools they use to attack other systems, etc. Research organizations such as universities and security research companies often use research honeypots. Honeypots according to their aims, now it is time to look into more details in levels of interactions way. Level of interaction stands for how much the hacker will be able to interact with the system. Based on the needs and the purpose of the experiment that one would like to examine, there are three categories of levels of interactions in honeypots[13].

With low interaction honeypots, one can get the least amount of data compared to other honeytrap systems. They are limited, so the risk that was taken from intruder is not big either proportionally. They can be used to identify new worms or viruses and analyzing the traffic that is going on through network. Low level of interaction honeypots are easy to configure and understand the level of interaction honeypot, which is Honeyd. Honeyd is a small daemon that creates virtual hosts on a network and runs on both UNIX and windows platform [8].

But this time, more information and more complicated attacks from the hacker can be obtained. As it is more advanced, it has more security holes so that hacker can access the system. Mycollect, honeytrap and Nepenthes are some of the medium interaction honeypots that are used today.

High interaction honeypots are the most advanced honeypots. Unlike low interaction and medium interaction honeypots, there is an operating system. As an effect, the hacker can achieve anything. Proportionally, added data can be imprisoned from the hacker’s activities. However, it is the most risky one when it comes to security as it provides such an access to the hacker that he does not have any restrictions.
These kind of honeypots are very time consuming and difficult to maintain. Honeywall is a good example of a high interaction honeypot.

Advantages And Disadvantages Of Honeypots

There are many security solutions available in the market. Anyone can browse the variety of choices through internet and find the most suitable solution for their needs. Here are the reasons why we should choose honeypots according to [13]. Honeypots can capture attacks and give information about the attack type and if needed, thanks to the logs, it is possible to see additional information about the attack.

As there are several important advantages of using honeypots, there are also some disadvantages of them as well [13].

Honeypot Perceptions And Advancement To Their Performance

We now take a look at the main concepts of honeypots and a few different ways in which they can be implemented. Honeypots are digital network bait and use deception to attract intruders [12], thereby distracting them from real production systems. A honeypot with several layers can slow down an attack, increasing the possibility of the attack being detected, and the possibility of countering the intrusion before it succeeds [3]. Intrusion detection and logging applications can be deployed within the honeypot to listen for and log unauthorized activity. Since no interaction with a honeypot is authorized, there is no need to filter through the information collected by a honeypot for suspicious traffic. This information can then be used to learn how the intruders operate, and to come up with suitable countermeasures. In summary, the main concept of a honeypot is to learn from the intruder’s actions [12].

Additionally, honeypots are not designed to be the sole source of security for any network; they should be used in conjunction with other security measures.

Approaches to Honeypot Implementation To implement a honeypot, some factors you need to consider include:

II. LITERATURE REVIEW

In 2012 Tekin Bicer et. Al. proposed a modeling-driven resource allocation framework. This technique supports time and cost sensitive execution and it is useful for data-intensive applications which executed in a hybrid cloud setting. They describe a modeling-driven resource allocation framework to support both time and cost sensitive execution for data-intensive applications executed in a hybrid cloud setting.

They explore resource allocation in the aforementioned hybrid cloud environment. Also describe a model-driven resource allocation framework to enable time and cost sensitive execution for data-intensive applications executed in a hybrid cloud setting. This framework considers the acquisition of cloud resources to meet either a time or a cost constraint for a data analysis task, while only a fixed set of local compute resources is available. Furthermore, they consider the analysis of data that is split between a local cluster and cloud storage. Than monitor the data processing and transfer times to project the expected time and cost for finishing the execution. As needed, allocation of cloud resources is changed to meet the specified time or cost constraint. While the framework is dynamic, it tries to converge to a fixed number of cloud resources, so as to avoid allocating and de-allocating resources during the entire execution [14].

In 2011 Haoming Liang, Wenbo Chen and Kefu Shi proposed an approach which analyses the programming and task scheduling model according to the present-used cloud computing system. It gives examples to explain the process of programming and its modifying directions, as well as the process within which services and resources exchange. It gives explanation of Cloud computing, how social network may increase the Qos through changing the service load will be discussed. The appearance and rapid development of Cloud computing, is the evolution of virtualization technology, data intensive computing technology, and the reflection of application enrichment tendency in Internet. At present, there is no unified standard for Cloud computing [16].

Alberdi’s solution allows honeypots to monitor actual malicious activities by bots, worms, and viruses without letting them leave the honeypots [9]. Alberdi proposed “redirection kit”, which redirects outgoing attacks, such as messages bots used to coordinate attacks, to other honeypots to prevent the malicious attacks to other production servers through honeypots at the same time it prevents detection of the honeypots by the attackers. Using this mechanism, the bot masters still believe that their bots communicate with hosts outside of the network, while they are actually communicating with another honeypot in the same network.

Alosefer developed a low interaction client-side honeypot, “Honeyware”, for detecting malicious web servers [17]. Alosefer tested Honeyware against 94 URL’s he collected in advance (84 malicious, 10 benign). Honeyware detected 83 of the malicious URL’s. Capture-HPC detected 62 malicious URL’s, counted 23 as benign and misrecognized the remaining 9.
Since Honeyware is a low interaction honeypot, the data collected by it must be processed by an external processing engine, which takes time. Honeyware averaged 1 minute per URL, where Capture-HPC took 17 seconds. Alosefer concluded that high-interaction and low-interaction honeypots should be integrated to take the advantages of both in future honeypot design. Such integration should take the advantage of a low interaction honeypot, which is easily installed but requires external data analyses while high interaction honeypots have opposite properties.

Das proposed a solution to mitigate denial of service attacks by hiding production servers behind an access gateway, called “Active Server (AS)” [18]. Each AS works as a required gateway to reach a production server. Each AS authenticates its clients and once a client is authenticated, a path is opened between the client and a server. If an AS does not authenticate a client, it behaves as a honeypot, trapping the client there. If a client has access to multiple ASes, the client can be authenticated by any AS. Since the authentication of all clients at an AS must happen prior to access to the server, it will prevent DoS attackers from clogging the path from an AS to the protected server. This will save the legitimate clients who have access to only one AS to reach the protected server.

Lauinger warned that the concept of honeypot is being abused by attackers in recently prevailing instant messaging (IM) applications [7]. The technique, called “social engineering”, performs phishing attacks in IM by hijacking sentences human users create in real-time. A high-interaction honeypot, which is set up by an attacker, first invites two human users to an IM session, one at a time. Then, it establishes a connection between the two human users and the honeypot relays the messages as a hidden middleman between them. When the honeypot relays the messages, it monitors the ongoing message exchanged between the human users and modifies the messages on the fly. Using this technique, the honeypots can create messages that look like those created by actual human users.

In 2011 Guannan HU and Wenhao ZHU introduced A Dynamic User-integrated Cloud Computing Architecture. It is a dynamic user-integrated cloud computing architecture. This architecture integrates clients with storage capacity and computing competency to data center dynamically, it expands the scale of cloud computing data center. Collaboration of clients with the data center provides services to the other users. They offered a dynamic user-integrated cloud computing architecture, which integrates the clients into the datacenter dynamically; meanwhile, the clients collaborate with datacenter to provide services to other clients.

Not only can it reduce the investment in building datacenter, but also improve service capacity and quality the cloud computing datacenter provides over the existing network. Meanwhile, we introduce a new payment policy to encourage clients to integrate into datacenter and become a part of datacenter. The cloud computing architecture based on the architecture dynamically integrates clients which have storage capacity and computing competency into datacenter, it can overcome unreliable services which exist in the existing architecture, using existing resources to improve service capacity and quality. Datacenter is built by cloud computing services vendor, virtual datacenter is the sum of datacenter and clients. Datacenter scales up while it integrates clients and collaborates with them to provide services to other clients. Users must sign Service Level Agreement (SLA) with cloud computing vendor when they use the services. The storage capacity and computing competency of the clients will be as resources to integrate into datacenter. Storage integration is that storage capacity of clients dynamically integrates into datacenter and it is available to other users as resources. Resources of existing cloud computing architecture are centralized management. Software and hardware resources and personal data are stored in datacenter. The premise that the cloud service is provide in centralized way is that network must meet the requirement of network bandwidth and its stability, but it is very difficult to achieve in existing public network. So datacenter integrates the clients into it, encoding and encrypting data and storing the data in datacenter, while manages the data with access control [5].

In 2011 Xiang Li, Jing Liu, Jun Han and Qian Zhang proposed. The article describes design of micro-learning platform architecture constructed through cloud computing technology, details the layered architecture design of micro-learning platform based cloud, aiming at depending on the powerful computing capacity and mass storage of cloud to better meet the practical learning requirements of life-long learners. Better meet the practical learning requirements of a life-long learner, micro-learning platform can be built based on cloud computing. The emergence of cloud computing has an important significance for us to build a unified, open, flexible micro-learning platform. he construction of micro-learning platform makes meeting the real-time and practical learning needs undoubtedly, namely, helping the learners achieve his practical learning purposes, helping them solve practical problems in daily life possible. Building repository of micro-content for storing can provide support of real-time updates content for micro-learning. Content in the repository can be expanded in two ways: add micro-information directly and design the information in a miniaturization way.
Besides the users can share and exchange information. There are three main requirements: to ensure the consistency, security, reliability and scalability of data when a large number of data is managed and maintained, to achieve a unified centralized management of physically distributed data resources, to achieve the dynamic storage of data resources. Obviously, these features put forward a high demands for the data storage system. But the emergence of cloud storage technology provides a new storage solution for the micro-learning resources. Centering on cloud computing server, deploy the virtual hardware servers in a dynamic way. For the traditional distance education system, teaching resources is usually in a simple distribution, compatible badly; low use efficiency, duplication of resources, weak system expansion, high cost and any other defects. Micro-learning platform based on cloud computing has greater progress in the education service capacity, resources sharing, configurability and scalability. It can ensure the informal learning progress smoothly, suitable for lifelong education particularly [6].

### III. PROPOSED METHODOLOGY

1. First of all we create a cloud environment which consists of a number of nodes.
2. We apply honey pots using the following algorithm on these clouds to prevent from different types of attacks.
3. The sender when send the packets to any other node then first of all get the ip of the sender.
4. The honeypots contains a control center in which we have applied some snort rules.
5. The honey gateway is implemented at each level of the cloud nodes.
6. If the packet contains a victim or any type of attack then the honeypot gateways are used for the repeated detection of attacks in the packet.

| VM | Virtual Machine’s |
| Npkts | No. of Packets |
| U | User of cloud |
| N | No. of packets |
| Pkt | Packets |
| DC1 | Data Center 1 |
| Ninst | No. of instance value |
| BR1 | Broker |
| DATApkt | Data Packet |
| COSTest | Estimated Cost |
| TIMEest | Estimated Time |
| Totalpkts | Total packets send |

### Table 1. Notations Used

Input: No. of cloudlets, datacenters, VM’s, Brokers, Honey pots, Npkts
Output: Intrusion Detected.

```
Repeat
If user ‘U’ of any cloud will sends ‘n’ number of packets ‘pkt’ to the datacenter ‘DC1’
Npkts := n{pkts} / Ninst ;
Ninst=0;
i=0;
Repeat while i<=Npkts
Npkts[i] -> BR1, that packet is send to the Broker
If Totalpkts > threshold
Then flooding attack
else
NEWdata=DATApkt -> UTF-8 format
NEWdata → Control_Center
Control_Center ← DC1
DC1 → Check(NEWdata);
If DC1{NEWdata} → UTF-8 format
Msg(No intrusion detected);
Else
Msg(intrusion detected);
```

IV. RESULT ANALYSIS

The figure shown below is the result analysis of the false alarm ratio on the basis of probability of detecting intrusion in the packets. Here the alarm ratio is detected on the cases of worst and best.

As the number of detection probability increases the chances of false alarm ratio decreases in both worst and best case.
Complexity based False Alarm Ratio

The figure shown below is the detection of false alarm ratio on the basis of probability of the detection of intrusion in the packets.

<table>
<thead>
<tr>
<th>Complexities of detection flooding attack</th>
<th>Complexity of flooding attack detection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packets</td>
<td>Threshold Probability</td>
</tr>
<tr>
<td>50</td>
<td>0.5</td>
</tr>
<tr>
<td>100</td>
<td>0.6</td>
</tr>
<tr>
<td>150</td>
<td>0.7</td>
</tr>
<tr>
<td>200</td>
<td>0.8</td>
</tr>
<tr>
<td>250</td>
<td>0.9</td>
</tr>
<tr>
<td>300</td>
<td>&gt;0.9</td>
</tr>
</tbody>
</table>

Detection Time of intrusion based on packets

<table>
<thead>
<tr>
<th>Packets</th>
<th>Detection Probability</th>
<th>Detection Time(ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>98</td>
<td>3.7</td>
</tr>
<tr>
<td>10</td>
<td>98</td>
<td>3.6</td>
</tr>
<tr>
<td>15</td>
<td>100</td>
<td>3.2</td>
</tr>
<tr>
<td>20</td>
<td>100</td>
<td>3.1</td>
</tr>
<tr>
<td>25</td>
<td>100</td>
<td>3.2</td>
</tr>
<tr>
<td>30</td>
<td>100</td>
<td>2.9</td>
</tr>
</tbody>
</table>

V. CONCLUSION

Here in this paper a new way of detecting intrusions in the cloud environment is proposed using honeypot. The technique implemented here provides security various attacks in the wired or in wireless network. The result analysis shows the performance of the proposed methodology. Also the bandwidth utilization is more and the performance rate for the detection of intrusions in the packets is more.

REFERENCES


