Research on Various Next Generation Honeypot Systems

Roshni M. Kinariwala
Department of Computer Engineering,
Parul Institute of Engineering and
Technology, Vadodara – 391760

Prof. G. B. Jethava
Department of Computer Engineering,
Parul Institute of Engineering and
Technology, Vadodara – 391760

Abstract

*The increasing use of next generation cloud computing in Internet for providing the different services leads us to the security problems. There are many security problems like DoS (Denial of service) attack, hacking intrusions viruses and worms and many more. Since all the resources are connected with each other and monitored centrally by controller in a cloud environment make an easiest way for hackers. If any of a resource is hacked, it may also expose other resources in the cloud. They cannot be eliminated but they can be reduced by using a tracing system Honeypot. Honeypot is used to collect information about hackers. It tries to gain knowledge about attacker's patterns, purposes and motivation for attack. In this paper we will discuss the basic idea of honeypot, different Honeypot systems, and finally the comparison of today's honeypots available.***

Keywords - **Cloud computing, Cloud Security, Honeypot, HoneyD, Honeynet, Intrusion Detection.**

1. Introduction

Cloud computing [1] is increasingly used in the current world of Information Technology and also in offices, schools and homes. The reason behind this is to share or exchange and communicates the data or information via these devices for effective and efficient communication of data in cloud and for that they require extra services which is provided by the Internet. These services are provided by the Internet Service Provider (ISP). The increasing use of cloud computing in Internet is now becoming next generation architecture of IT Enterprise.

Use of cloud computing leads us to major security issues like DoS (Denial of service) attack, hacking intrusions viruses and worms and many more. For the protection of network, information and property from theft, corruption, or alteration many techniques are used. These attacks are not totally eliminated but they can be reduced. So we use techniques for detecting and preventing of these attacks. There are mainly 3 categories for security of cloud, Detection of attacks, prevention of attack and respond to attack. Intrusion detection systems (IDS) and vulnerabilities scanners come into detection of attacks, whereas Intrusion prevention system (IPS), Antivirus and firewalls comes into prevention of attacks and incident response team comes into response to attacks.

In classical networking, a Honeypot is a trap set for the attackers making them believe that the system is vulnerable [1]. It is a system which is placed open on network for tracking the hackers and their activities. The data gathered by honeypot is then analyzed to learn the tools, patterns, motives and techniques used by the hackers for protecting the systems on the network.

2. Types of Honeypot

Honeypots are generally classified into based on their deployment, level of design criteria or as per the classes of attacks.

2.1 Based on the deployment two main honeypots are as follows:

- Production honeypots
- Research honeypots[2]

Production honeypots are primarily used and they are easy to use and it capture only small amount of information. Production honeypots are used to assist an organization in protecting its internal IT infrastructure. They are valuable to the organization especially commercial, as they help to reduce the risk that a specific organization faces. They secure the organization by policing its IT environment to identify attacks. Research honeypots are used to gather much information about hackers and their activities. They are complex in nature and not specifically valuable to organization.

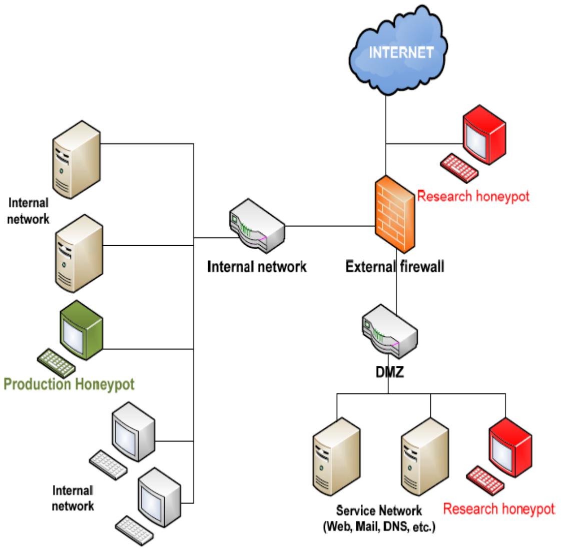

Figure 1 Production and Research honeypot deployment

2.2 Based on the design criteria honeypot can be classified into three categories:

- High-interaction honeypots
- Medium-interaction honeypots
- Low-interaction honeypots [3].

High-interaction honeypots are more complex and difficult for real time systems. They allow attackers to interact with real time application or systems and also capture the information for analysis of their behaviour shown in Fig. 2.

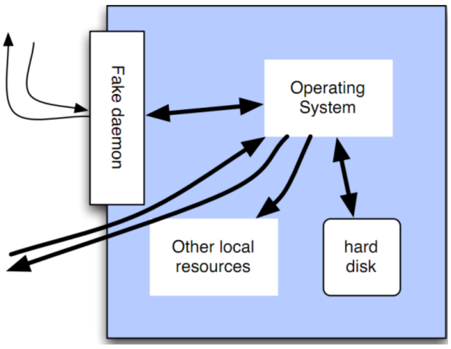

Figure 2 High-interaction honeypot

A medium interaction honeypot gathers more information about attacks compared to a low interaction honeypots. It provides a facility to the attacker to interact a bit more with the honeypot, as shown in Fig. 3. It takes the attacker one step ahead so that the honeypot can able to reply to a specific commands, by using preconfigured messages. Low-interaction honeypots are the very easy to install, configure, deploy, and maintain because of their simple design and basic functionality.

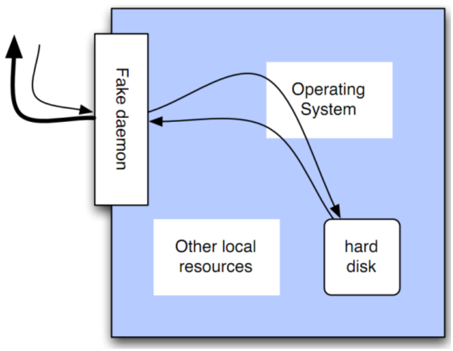

Figure 3 Medium-interaction honeypot

A low interaction honeypot provides a limited communication between the honeypot and the attacker, as shown in Fig. 4. They are easy to install, configure, deploy, and maintain because of their simple design and basic functionality. Because of its simplicity it has the lowest level of risk.

2.3 Honeypots can be classified as per the attack classes and targeted attacks like client side and server side attacks [4]. Honeypots which gives the deep knowledge of client side attacks are Client honeypots also called Honeyclient or active honeypots. In opposite honeypots which gives the deep knowledge of server side attacks are Server honeypots also called passive honeypot. They are widely used in today’s research area of cyber security.

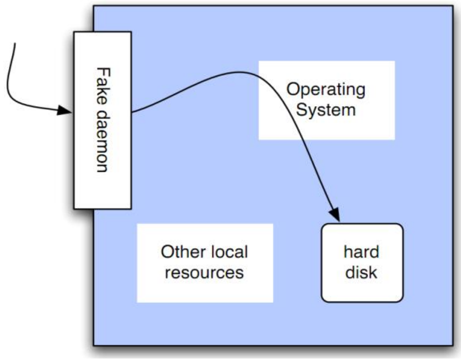

Figure 4 Low-interaction honeypot

The difference between these levels of interaction is shown in below Table 1.

3. Different Honeypot systems

There are various honeypot systems used for security purposes. The main five honeypots are explained in this section:

- HoneyD
- HoneyNet
- ManTrap
- Back officer friendly
- Specter

3.1 HoneyD

HoneyD is an application which enables the setup of multiple virtual honeypots on a single machine, each with different characteristics and services [7]. HoneyD is normally represented for low level of interaction. It is primarily designed as a production honeypot for detecting an unauthorized activity. By using HoneyD we can setup honeypot with any of the personality and any of the services like HTTP, SMTP, SSH etc.

HoneyD follows the principle that whenever it receives any connection or request for a system which does not exist, it immediately assumes that that request is a fake or an attack. HoneyD suppose the IP address of the proposed goal when it receives such type of traffic. After that it starts the emulated services for the port that the request or connection is attempting. Then the emulated service starts, it will interact with an attacker and captures all his activities. The emulated service is closed when the attacker stop his work. After this HoneyD continuously wait for new more traffic or connection request to system that do not exist. It is a very efficient method and it repeats this process of assuming the IP address of the proposed victim. It can emulate more than one IP addresses and interact with all the attackers at the same time.

3.2 HoneyNet

Honeynets are representing the high level of interaction, deployed within a highly controlled network. It provides multiple honeypots instead of only providing the attacker to a whole operating system for attack and for interaction. The value of honeynet is being probed, attacked or compromised so because of its nature it becomes a honeypot. The highly controlled network will capture the all activities of attacker which happens in the honeynet and decreases the security risk using those activities. Honeynet follows the same principle as a Honeypot. Any request sent to honeynet is suspect; probably a probe or even an attack and anything sent from a honeynet intend that it has been conciliated when an attacker or tool is launching activity. Honeynets lead us to one step ahead of honeypots: Honeypot contains a single system where as honeynets allows physical networks of multiple systems. We do not have to install honeynets and we cannot put honeynet on a network. Instead, Honeynets are an architecture that builds a highly controlled network, within which you can place any system or application you want [8].

3.3 ManTrap

”ManTrap can create a virtual minefield that an internal attacker must successfully navigate in order to reach his target. One step in the wrong direction and the attacker is exposed [9].” ManTrap is a first high-interaction commercial honeypot created, maintained, and sold by Recourse Technologies, now called Decoy Server. ManTrap is more powerful and unique in that it is designed to be not only attacked but also compromised. It creates a highly controlled operating environment that an attacker can interact with. Instead of limited operating systems it creates a fully functional operating system containing virtual cages. The attacker is unable to exit and also unable to attack the host system because the cages are from logically controlled environments. Mantrap creates cages that the mirror copies of the master operating system rather than creating an empty cage and filling it with certain functionality. Each cage is a fully functional operating system that has the same capabilities as a production installation. Each cage has few limitations on its own virtual world. Customization of each cage is possible or done by an administrator because of a separate physical system. The administrator can create users, install application, compile his own binaries and run processes. The attacker assumes that the cage is a truly separate physical system when an intruder attacks and gains access to a cage. He is unaware that he is in a caged environment where every action and keystroke is recorded [9].

3.4 BackOfficer Friendly (BOF)

BackOfficer friendly, normally called as BOF is extremely simple to install, easy to configure, and low maintenance. Because of its simplicity it is one of the excellent available tools now days. BackOfficer Friendly is represented as a low interaction honeypot. It can run on almost all Windows based platform, even the older ones like Windows 95, Windows 98, etc. However, this simplicity comes at a cost and has extremely limited capabilities. It has a small set of services and whenever any hacker attempts a connection with any of these services, BackOfficer Friendly listens to it on that port and generates an alert and close the transaction. It logs attackers IP address and operations he tries to perform. None of the services emulate a specific application or version, only the functionality of the service [10].

3.5 Specter

Specter is a commercially supported honeypot whose value lies in detection. Specter is also a low-interaction and a primarily production honeypot which has a greater functionality and capabilities than BOF. It can also emulate different operating systems and vulnerabilities instead of only emulate a more services. It also has extensive alerting and logging capabilities. Because of emulating limited interaction services, Specter is easy to deploy, maintain, and is low risk. It can gather a limited amount of information compared to high-interaction and medium-interaction honeypots. It has a limitation that it cannot listen on or monitor a port that is already owned by another application and also it has the same limitation as BOF has. That means FTP port cannot monitored by Specter. It also has the capability of emulating different operating systems. This is done by changing the behaviour of the services to mimic the selected operating system [11].

4. Comparison of various Honeypots

In this section we have discussed the comparison between the commercially available honeypots. BackOfficer Friendly being the simplest is low interaction honeypot with very limited capabilities and a small set of services to emulate. Specter has more options and can generate fake replies, as discussed. HoneyD is the most popular Linux based honeypot that can simulate many machines at the same time. It is open source and most flexible of all low interaction honeypots. Mantrap based on Solaris is middle-high interaction honeypot and provides caged environment to the attackers while simulating multiple OS. Following Table 2 shows the comparison between various honeypot systems:

5. Advantages and disadvantages of honeypots

5.1 Advantages

- Honeypot captures small amount of information. That means instead of capture data in GB, it captures only 1 MB of data in a day.
- Honeypots are designed to capture anything thrown at them, including tools or tactics never seen before.
- Honeypots require minimal resources, they only capture bad activity.
- Honeypots can collect in-depth information that few, if any other technologies can match.
- Finally, honeypots are conceptually very simple. There are no fancy algorithms to develop state tables to maintain, or signatures to update. Thus there will be mistakes or misconfigurations.

5.2 Disadvantages

- Honeypots can only track and capture activity that directly interacts with them. Honeypots will not capture attacks against other systems, unless the attacker or threat interacts with the honeypots also.
- Honeypots have the risk of being taken over by the bad guy and being used to harm other systems. Depending on the type of honeypot, it can have no more risk than an IDS sensor, while some honeypots have a great deal of risk.

6. Conclusion

This paper demonstrates that the Honeypots are the security systems which can be used to provide network security and business profit. The honeypots systems are being classified according to the level of interaction, design criteria and attack classes. In this paper we have shown the difference between each level of interaction and also we have discussed the various honeypots systems. Also the comparisons between those systems with several parameters are shown. The main purpose of this paper is that by implementing honeypot systems in a cloud computing can be used to reduce the security risk.

REFERENCES

[1] M Balamurugan , B Sri Chitra Poornima "Honeypot as a service in cloud", *International Conference on Web Services Computing , IJCA* 2012

[2] Levin, J., Labella, R. Henry,: "The Use of Honeynets to Detect Exploited Systems Across Large Enterprise Networks", IEEE Proceedings, June 2003.

[3] http://en.wikipedia.org/wiki/Honeypot_(computing)

[4] Design and Implementation of Linux Based Hybrid Client Honeypot Incorporating Multi Layer Detection

[5] Design & Implementation of Honeyd to Simulate Virtual Honeypots

[6] Liu Dongxia & Zhang Yongbo, "An Intrusion Detection System Based on Honeypot Technology", 2012 International Conference on Computer Science and Electronics Engineering, 978-0-7695-4647-6/12, 2012 IEEE

[7] http://security.rbaumann.net/download/honeyd.pdf

[8] Levine, J., Grizzard, J. "Using honeynets to protect large enterprise networks," Security & Privacy Magazine, IEEE, vol. 2, pp. 73-75, 2004

[9] http://www.recourse.com/index.html

[10] Bao, J., Gao, M. "Research on network security of defences based on Honeypot", International Conference on Computer Applications and System Modelling, 2010.

[11] Spitzner, L.: Tracking Hackers. Addison Wesley, September 2002.

[12] BackOrifice.http://www.cultdeadcow.com/tools/bo.html

[13] Specter: www.specter.com/index.html

[14] Mantrap:www.recourse.com/products/mantrap/trap.html.

[15] Honeyd: www.honeyd.org/.

[16] Honeypots: Tracking Hackers- http://www.tracking-hackers.com/index.html

Table 1: Differences between each level of involvement based on [7]

Degree of involvement	Low	Medium	High
Real operating system	-	-	Yes
Installation and configuration effort	Easy	Medium	Difficult
Deployment and maintenance effort	Low	Low	Very High
Information Gathering	Low	Medium	Extensive
Level of Risk	Low	Medium	High
Compromised wished	-	-	Yes
Knowledge to run	Low	Low-Mid	High
Knowledge to develop	Low	Mid	Mid-High

Table 2: Comparison of various Honeypots

	HoneyD	HoneyNet	ManTrap	BOF	Specter
Open Source	Yes	Yes	No	No	No
Interaction Level	Low	High	High	Low	High
OS Simulation	Yes	Yes	Yes	No	Yes
Log File Generation	Yes	Yes	Yes	No	Yes
Services	Unlimited	Unlimited	Unlimited	7	13

supported					