International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 5 Issue: 5 711 – 714

A Survey on Active Defense Honeypot Mechanism for Information Security

Binal S. Naik
Department of Computer Engineering
Parul Institute of Engineering and Technology, Parul University
Vadodara, Gujarat, India
binni.naik594@gmail.com

Harshal Shah
Department of Computer Science and Engineering
Parul Institute of Engineering and Technology, Parul University
Vadodara, Gujarat, India
harshal.shah@paruluniversity.ac.in

Abstract— Information security is a rising concern today in this era of the internet because of the rapid development of the new attack techniques. The existing security mechanisms such as traditional intrusion detection systems, firewalls and encryption are the passive defense mechanisms. This has led to growing interest in the active defense technology like honeypots. Honeypots are fake computer Systems which appears vulnerable to attack though it actually prevents access to valuable sensitive data and administrative controls. A well designed and developed Honeypot provide data to the research community to study issues in network and information security. In this paper we examine different Types of Honeypots, Honeypot concepts and approaches in order to determine how we can intend measures to enhance security using these technologies. In this work a web application honeypot architecture is proposed.

Keywords-*web application Honeypot;Types of honeypot;net defense;intrusion detection system;Interaction honeypot*

*****

I. INTRODUCTION

The number of attacks on computer systems has increased in last few years. The network environment becomes more and more complicated. The threat is becoming the multi-source and dynamic [1]. Security mechanisms such as routing security, identity authentication, encryptions and firewall are static, passive security mechanisms but only the passive defense is not sufficient.

Intrusion detection system is divided into two categories: anomaly detection and signature detection (misuse detection). Anomaly detection based on protocol can verify the unknown attacks effectively, but cannot detect attack violating an agreement [2]. Misuse detection system matched attack action with stored attack signature in intrusion rule databases. This method achieves a high detection rate and required less time. However, signature detection system is unable to distinguish new type of attacks or a large number of complicated attacks [2]. IDS can't give alert when intrusion occurred using new signature.

Honeypot technology is not to replace the traditional security mechanisms and defense technologies, but it's supporting and complementary. A honeypot system can detect attack behavior and redirect such attacks to a strictly controlled environment to protect the practical running system by giving real systems, services and applications [3]. It can provide forensic evidence that is admissible in a court of law. It can be used as legal evidence as long as it is deployed correctly and is not advertised Honeypot technology proactively detect and respond to network intrusion and attacks [4].

This paper is organized as follows. In section 2, system design of Honeypot is explained. Section 3 presents the classification of Honeypot based on certain criteria. Section 4 shows honeypot system related work. In section 5 web application honeypot architecture is proposed. Finally, future research trends and conclusions are drawn.

II. SYSTEM DESIGN OF HONEYPOT

Definition

"A Honeypot is an information system resource whose value lies in unauthorized or illicit use of that resource [5]."

Honeypot is a decoy, put out on a network to attract attackers. A Honeypot works by fooling attackers into believing it is a legitimate system. Attackers attack the system without knowing that they are being observed completely. Honeypot looks like a really host provided important service by creating the appearance of running full services and applications, with open ports that might be found on a typical system or server on a network. This way honeypot create confusion for attackers and monitor the intruder without risk to production servers or data.

The related information of the attackers such as the IP address, motives of the attackers entering the system and attack behavior of the attacker will be collected generally through the implementation of the background software [6]. This monitors and records the network communication data between the attackers and honeypot host, and uses some analytical tools to interpret and analyze these data.

A. Honeypot System Processing Flow

Honeypot system has generally three modules which are induced, deceive and analysis. The induced module is used to attract the attackers to attack on the Honeypot system. The deceived module calls the simulation information from the database for the deceived host to generate false information which will be sent to the attackers [6]. All the induction and deception events of the system are recorded in the remote log

server, and analyzed by the analysis module for adjusting the induction and deception strategy. That is shown in Figure 1.

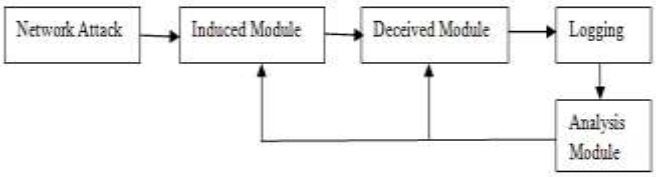

Figure 1. Honeypot System Process

B. Component of Honeypot

1) Data Capture: Data capture is the monitoring and recording of all threat activity in the Honeynet architecture, these captured data are then analyzed to learn the tools, tactics, and motivation of the attackers [7]. It is a difficult section to any honeypot as the encrypted channels like SSH,SSL and IPSec are used by attackers to perform attack activities [8].

2) Data Control: There was always the possibility of an attacker or malicious code uses a honeypot to attack or harm non-honeynet systems [9]. Techniques such as bandwidth restrictions, counting outbound connections, or intrusion prevention gateways can be used for data control. The technique used is limiting outgoing connections [10].

3) Data Analysis: Data collected by the honeypot should be converted in to useful information,for that we must have some ability to analyze data [11]. This information can prove important in analyzing the attackers' activities.

III. CLASSIFICATION OF HONEYPOT

According to the Design Deployment honeypot can be classified into Production and Research honeypot.

A. Production Honeypot

A production honeypot is one used within an organization's environment to protect the organization and help mitigate risk [12]. Production honeypots emulate the production network of the company. Attackers interact with them in order to expose vulnerabilities of the production network. Uncovering these vulnerabilities and alerting administrators of attacks can provide early warning of attacks and help reduce the risk of intrusion [13]. Production honeypot require less functionality then a research honeypot. They are easier to build and deploy. Although they identify attack patterns, they do not give much information about the attackers than research honeypots.

B. Research Honeypot

Research honeypots are designed to get knowledge about the blackhat community. They are used primarily by research, military, or government organizations. It gives real operating systems and services that attackers can interact with. So they involve higher risk, collect extensive information and intelligence on new attack techniques and methods. It gives the information as who is the attacker, how do they attack, why do they attack, and when? This intelligence gathering is one of the most unique and exciting characteristics of honeypots [14]. Research honeypot is more complex to deploy and maintain.

According to the Honeypot with Different Attacker Interaction Level, honeypot is divided into three major classes: low-Interaction, medium interaction, and high-interaction.

C. Low-interaction Honeypot

Low-interaction honeypot systems do not provide intruders with the actual operating system for remote login [3]. They simulate only the services frequently requested by attackers. A low-interaction honeypot provides specific analog services that can be conducted by monitoring a specific port [15]. Low interaction honeypots emulate network services on preconfigured port, such as FTP, SQL, Web, SSH, etc. They are easy to install, deploy and maintain, as well as minimize the risk of potential damage by an attacker. Low-interaction honeypots capture only limited amount of information.
Example: Honeyd, Specter

D. Medium–interaction Honeypot

Medium-interaction honeypots provide the attacker with a better illusion of an operating system since there is more for the attacker to interact with. More complex attacks can therefore be logged and analyzed [16]. The data collected is more beneficial than a low interaction honeypot because of a higher interaction.

In terms of security, a low risk of potential intrusion is expected as the honeypot only answer to preconfigured commands. They can capture more information. They more efficiently interact with intruder than do low-interaction honeypots but less functionality than high-interaction honeypots. It enables the system to collect high amounts of data but increases the risk of intrusion. There is one disadvantage of the medium-interaction honeypot that the attacker generally, quickly discovers that the system does not behave as it should.
Example: mwcollect, nepenthes and honeytrap

E. High-interaction Honeypot

High interactive honeypots are configured with real operating system and provide a real operating system for attackers. They are a complex solution and involve the deployment of real operating systems and applications[17]. It provides a large amount of information to the researcher about unknown attack and previously known attack. Any error in the system may allow a hacker to control the full operating system, attack other systems, or intercept messages in the application system [18].

This honeypots are best in the case of Zero Day attacks. This types of honeypots are complex and time consuming to setup or design, and involves the highest amount of risk because they involve an actual operating system [19]. This type of honeypot must be always behind a firewall and constantly monitored.
Exampls: Honeynets Sebek

F. Honeytokens

Honeytoken is a honeypot that is not a computer, but a fake digital entity. Like Honeypots, no honeytokens has any authorized use [16]. Honeytokens can be a credit card number, a database entry, an Excel spreadsheet or even a PowerPoint

Presentation. Any interaction with the honeytoken is suspicious. Honeytoken's selection depends on user's creativity, they can decide what they want to use as a honeytoken.

For example, adding a fake record to the credit card database that wouldn't normally be selected by authorized queries. If someone does access it, you know that they're abusing their privileges somehow. It helps in tracking the activities,and determining the actions, capabilities and intensions of, a malicious intruder [16]. Honeytokens are extremely flexible, there is no right or wrong way to use them. Due to their flexibility, you can customize them to easily integrate into your environment. Implementation cost of Honeytoken is minimal.

TABLE I. COMPARISON OF DIFFERENT HONEYPOTS

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

IV. RELATED WORK

One of the most frequently used low interaction honeypot is Glastopf [20]. It deals with the SQL injection, remote, and local file inclusion. Unfortunately, Glastopf does not have an ability to collect information about the attacker's identity and it is designed only for tools-generated attacks. When real humans open the Glastopf website they can easily find that it is a fake system to trap attackers. Another web application honeypot is High Interaction Honeypot Analysis Toolkit (HlHAT) [21]. It requires a dedicated server because it is a high interaction honey pot. Server must be set with a variety of security configurations to protect HIHAT. If the server is controlled by the attacker then HIHAT can be used to control other systems.

A honeypot that equipped with counter attacks to remote attacker was presented by Sintsov [22]. It uses Java applet to get attacker identity. However, this applet is now blocked by modem browser. Distributed Web Honeypots can be installed by all contributors from around the world [23]. Takeshi et.al made a proposal to fix url path to the existing high interaction web honeypot [24].

V. PROPOSED WEB APPLICATION HONEYPOT ARCHITECTURE

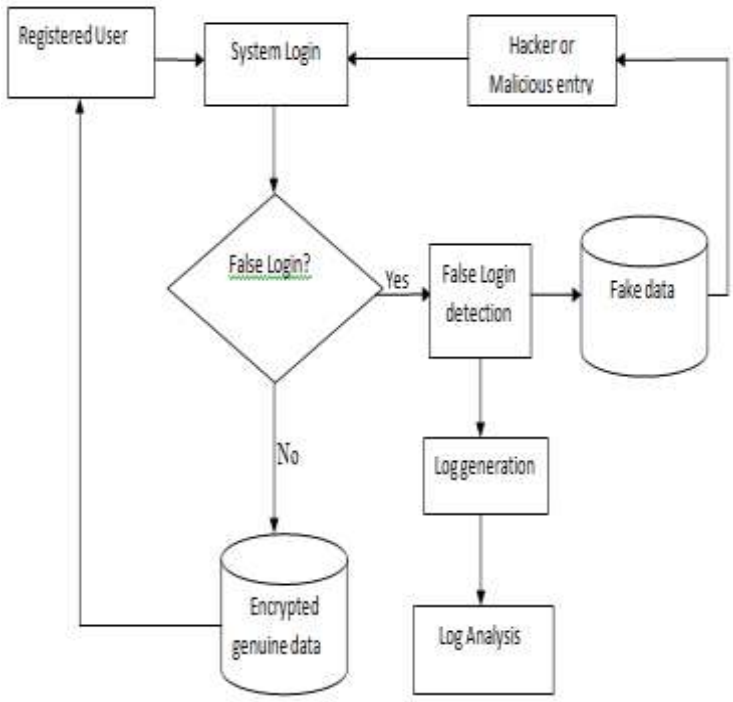

Figure 2. Proposed Web application Honeypot Architecture

- When Registered user and Hacker do the login in to web application system False login condition will be checked (e.g.if id=fake and password=fake).
- Hacker get the id and passwords for the system login through SQL injection.
- But proposed Honeypot system mechanism will give only the fake database access to the Hacker so the hacker will only get fake id and fake password.
- If false login condition satisfied then the system will detect the false login of the Hacker and Hacker will redirected towards the fake admin page.
- Else the system will give the genuine database access to the registered user. All the valuable data will be encrypted here.
- When Honeypot system will detect false login it will be recorded in to the logs.
- Further log analysis can be done on that log generated data to view attacker related information.

VI. FUTURE RESEARCH TRENDS

The research on honeypot technology can be categorized into five major areas:

- New types of honeypots to deal with emergent new security threats.
- To reduce the maintenance and configuration cost of honeypots as well as to improve the threat detections accuracy.
- Honeypot output data utilization to improve the accuracy in threat detections.

- Counteracting honeypot detections by attackers.
- Legal and ethical issues in using honeypots.

VII. CONCLUSION

Honeypot gives innovative way to attacks prevention, detection and reaction. Honeypots are simple to use, flexible to configure, occupies less resources and effective in complex environment. This paper presents detail concept of honeypots, a web application honeypot architecture, honeypot types and component, how they are designed to attract intruders to a decoy system so that their activities can be monitored without risk to production systems or data.

REFERENCES

[1] Suo, Xiangfeng, Xue Han, and Yunhui Gao. "Research on the application of honeypot technology in intrusion detection system." Advanced Research and Technology in Industry Applications (WARTIA), 2014 IEEE Workshop on. IEEE, 2014.

[2] Yang, Yun, and Jia Mi. "Design and implementation of distributed intrusion detection system based on honeypot." Computer Engineering and Technology (ICCET), 2010 2nd International Conference on. Vol. 6. IEEE, 2010.

[3] Koch, Robert, Mario Golling, and Gabi Dreo. "Attracting sophisticated attacks to secure systems: A new honeypot architecture." Communications and Network Security (CNS), 2013 IEEE Conference on. IEEE, 2013.

[4] Bao, Jian, Chang-peng Ji, and Mo Gao. "Research on network security of defense based on Honeypot." 2010 International Conference on Computer Application and System Modeling (ICCASM 2010). Vol. 10. IEEE, 2010.

[5] Martin, William W. "Honey pots and honey nets-security through deception." SANS Institute Paper (2001).

[6] Li-Juan, Zhang. "Honeypot-based defense system research and design." Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on. IEEE, 2009.

[7] Nadya elmoussaid and all, Intrusion Detection Based On Clustering Algorithm International Journal of Electronics and Computer Science Engineering 1059 Available Online at www.ijecse.org ISSN-22771956/V2N3-1059-1064, 2013.

[8] Vusal Aliyev, Using honeypots to study skill level of attackers based on the exploited vulnerabilities in the network Department. Thesis memory of Computer Science and Engineering Division of Computer Security Chalmers University of technology Göteborg, Sweden, 2010.

[9] The honeynet project ³capture et étude de plusieurs attaques en utilisant des honeynets de 1ère génération (GenI) » 1999-2001.

[10] The honeynet project ³Pot de miel-Pham QuyetThang, Victor Moraro 2002-2003 GenI et des honeynets virtuels » 1999-2001.

[11] N. Provos and T. Holz. Virtual Honeypots: From Botnet Tracking to Intrusion Detection. Addison-Wesley Professional, GenIIet des honeynets virtuels.2007.

[12] Karthik, S., Samudrala, B. and Yang, A.T. Design of Network Security Projects Using Honeypots. Journal of Computing Sciences in Colleges, 20 (4).

[13] Gubbels, Kecia. "Hands in the Honeypot." GIAC Security Essentials Certification (GSEC) (2002).

[14] Spitzner, Lance. "The value of honeypots, part one: Definitions and values of honeypots." Security Focus (2001).

[15] R. Berthier, D. Korman, M. Cukier, M. Hiltunen, G. Vesonder, and D. Sheleheda,ÃOn the Comparison of Network Attack Datasets: An Empirical Analysis," 11th IEEE High Assurance Systems Engineering Symposium, 2008. HASE 2008, 2008,pp. 39-48.

[16] Mokube, Iyatiti, and Michele Adams. "Honeypots: concepts, approaches, and challenges." Proceedings of the 45th annual southeast regional conference. ACM, 2007.

[17] Chawda, Kartik, and Ankit D. Patel. "Dynamic & hybrid honeypot model for scalable network monitoring." Information Communication and Embedded Systems (ICICES), 2014 International Conference on. IEEE, 2014.

[18] E. Cooke, M. Bailey, Z.M. Mao, D. Watson, F. Jahanian, and D. McPherson,ÃToward understanding distributed blackhole placement," Proceedings of the 2004ACM workshop on Rapid malcode, ACM New York, NY, USA, 2004, pp. 54-64.

[19] Niels Provos. Open Source honeyd. http://www.citi. umich.edu/u/provos/oneyd/

[20] L. Rist, S. Vetsch, M. Kobin, and M. Mauer, "Glastopf: A dynamic, low-interaction web application honeypot"", The Honeynet Project, 2010.

[21] M. Muter, F. Freiling, T. Holz, andl Matthews, "A generic toolkit for converting web applications into high-interactionhoneypots", Clarkson University, New York, 2007.

[22] A. Sintsov, "Hon eypotth at can bite: Reverse penetration", Black Hat Europe Conference, 2013.

[23] J. Riden, R. McGeehan, B. Engert, andM. Mueter, "Using Honeypots to learn about HTTP-based attacks", Honeynet Project, 2008.

[24] R. Barnett, "W ASC Distributed Open Proxy Honeypot Project'·, OW ASP and W ASC AppSec Conference, 2007.