Learning Probe Attack Patterns with Honeypots

Kanchan Shendre, Santosh Kumar Sahu, Ratnakar Dash and Sanjay Kumar Jena

Abstract The rapid growth of internet and internet based applications has given rise to the number of attacks on the network. The way the attacker attacks the system differs from one attacker to the other. The sequence of attack or the signature of an attacker should be stored, analyzed and used to generate rules for mitigating future attack attempts. In this paper, we have deployed honeypot to record the activities of the attacker. While the attacker prepares for an attack, the IDS redirects him to the honeypot. We make the attacker to believe that he is working with the actual system. The activities related to the attack are recorded by the honeypot by interacting with the intruder. The recorded activities are analyzed by the network administrator and the rule database is updated. As a result, we improve the detection accuracy and security of the system using honeypot without any loss or damage to the original system.

Keywords Honeypot · Virtual honeypot · Intrusion detection system · Honeyd

1 Introduction

Honeypot is the system to deceive the attacker by providing the decoy system which seems to be highly valuable, but badly secured so that the attacker can interact with that system. The administrator is able to analyze the attacker's

K. Shendre (✉) · S.K. Sahu · R. Dash · S.K. Jena
National Institute of Technology, Rourkela, India
e-mail: kanchanshendre19@gmail.com

S.K. Sahu
e-mail: santoshsahu@hotmail.co.in

R. Dash
e-mail: ratnakar@nitrkl.ac.in

S.K. Jena
e-mail: skjena@nitrkl.ac.in

A. Nagar et al. (eds.), *Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics*, Smart Innovation, Systems and Technologies 44, DOI 10.1007/978-81-322-2529-4_38

interaction with the system and categorize that attack by which the intent of the attackers can be known as discussed in [1, 2]. If a honeypot successfully interacts with the intruder, the intruder will never know that she/he is being monitored and tricked. Most of the honeypots are installed inside firewalls through which it can be controlled in a better way, although it can also be installed outside the firewalls. A firewall restricts the traffic coming from the Internet, whereas honeypot allows the traffic from the Internet, and restricts the traffic sent back from the system [1].

The parameters that are used to know the value fetched from a honeypot are given by [3]: (i) Type of deployment of honeypot and (ii) Scenario of deployment (location of deployment i.e. behind firewall inside DMZ, in front of firewall etc.). On the basis of these parameters a honeypot can act in the same way as bulgur alarm for detection of attacks, Prevention of attacks by deception and deterrence, responding to attacks by providing valuable logs regarding attack [3].

1.1 Areas of Deployment

There are two areas of deployment of honeypot: physical honeypots and virtual honeypots. In case of physical honeypots, the original system is allowed to completely compromise by the intruder. There is a risk to the system to be damaged by the intruder. So, another approach called as a virtual honeypot which provides the attacker with a vulnerable system which is not actually the real system is used, but the attacker never knows that he is dealing with the virtual system.

1.2 Types of Honeypot

There are two types of honeypot: High Interaction honeypot and Low Interaction honeypot. In a high-interaction honeypot the attacker can interact with a real system. While a low-interaction honeypots provides only some parts such as the network stack. The high interaction honeypot allows the adversary to fully compromise the system to launch the network attack. There is a higher risk in deploying high interaction honeypot. It takes more time for analyzing the events; it may take several days to know the intent of the attacker. It needs high maintenance so it is very hard to deploy. These are the drawbacks of high interaction honeypot.

Due to the drawbacks and risk in deployment of high-interaction honeypot, we have used the low interaction honeypot. Low-interaction honeypots are used to collect the statistical data and high-level information about attack patterns. Since an attacker interacts just with a simulation, he cannot fully compromise the system. A controlled environment is constructed by Low-interaction honeypots and thus the limited risk is involved: As the attacker cannot completely compromise the system, we do not need to worry about abuses of our low-interaction honeypots.

2 Related Work

The different types of honeypot can be used to detect different types attack by using different honeypot tools. Some previously known attacks and work done in honeypot is summarized as shown below (Table 1).

3 Objective

The objective of this paper is to learn the Probe attack patterns and generate rules for unknown probe attacks and update new rules into snort rule set. We not only trap the attacker but also try to know the motives and tactics used by the attacker.

4 Proposed Work

The honeypot is configured on the virtual system like Vmware. In low interaction honeypot, there are certain fingerprint files which contain the information about how the particular operating system will respond. For example, if we want to show the attacker that we are running Windows XP operating system, it will respond with certain characteristics, which will be used by the honeypot to respond to the attacker. The attacker will think that he is actually working with the Windows XP operating system but he will never know that he is actually dealing with the virtual operating system. The few of the important features of honeyd are creation, setting, binding and adding. In the creation process, we are going to create a template with some name or default. The structure of the template is as follows:

```
create<template-name>
create default
dynamic<template-name>
{Then we set the personality of the honeypot, i.e, the
operating system and mention certain protocol or action
such as reset, block or open.}
set<template name>personality<personality-name>
set<template name>default<proto>action<action>
We are adding the particular template along with protocol
name, port number and action.
add<template-name><proto>port<port-number><action>
```

Figure 1 shows the working model of IDS and honeypot together. The intrusion detection system redirects the attacker to the honeypot, when the malicious activity

Table 1 Related works in honeypot

Year	Author	Type	Attack type	Work done
2011	Saurabh et al. [3]	*l*	NA	Due to the lack of capabilities of existing security devices, there is a need to study honeypot deployment and analyze tools, methods and targets of the attacker
2006	Nguyen et al. [2]	*h*	NA	The purpose of this paper is to deploy a honeypot in such a way that it is well concealed from the intruder. The honeypot is deployed on Xen virtual machine with system Xebek
2004	Kuwatly et al. [4]	*h* + l	NA	The dynamic honeypot approach integrates passive or active probing and virtual honeypot
2006	Alata et al. [5]	*h* + l	U2R	The results based on a 6 months period using high interaction honeypot concludes that if the password is found to be weak it is replaced by strong one
2009	Vinu et al. [6]	NA	DOS	This paper has proposed the effective honeypot model for secured communication of authorized client and server
2009	Shujun et al. [7]	NA	Phishing	Honeypot is used to collect important information regarding attackers' activity
2009	Jianwei et al. [8]	*h*	malware	This paper has introduced a high interaction toolkit called HoneyBow containing three tools MwFetcher, MwWatcher, MwHunter
2003	Lance et al. [9]		The advance insider	This paper detects the threats done by the authorised insider
2009	Almotairi et al. [10]	*l*	NA	The technique for detecting new attacks using PCA with low interaction honeypot is presented in this paper

Note l Low interaction honeypot and *h* High interaction honeypot

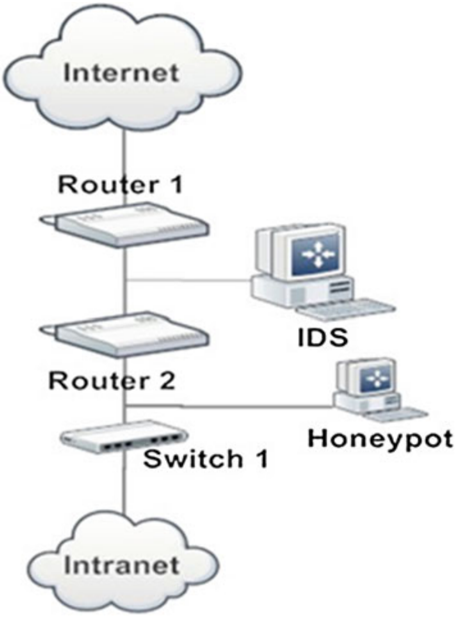

Fig. 1 The working model of IDS and honeypot

is detected. The intruder interacts with the honeypot and tries to know its vulnerabilities and open ports. The honeypot allows to gain access to the limited resources of the system so that it should not make any harm to the important files and resources. The attack activities of the particular intruder is logged by the honeypot. This log file is then used to create new rules which are further added to the list of already generated rules. Once this is done, when the same type of behavior occurs next time, this is directly considered as attack and there is no need to redirect that intruder to the honeypot. In this way, the novel attacks can be detected by the intrusion detection system.

5 Result and Discussion

We have studied the probe attack patterns and represented the number of instances of each type as follows (Table 2):

Table 2 Number of instances for each type of probe attack

Sl. no.	Name of attack	No. of instances
1	nmap	11,609
2	portsweep	1,915
3	ipsweep	2,177
4	satan	2,013

We have estimated some snort rules by using honeypots and represented them in the form of pseudocode as follows:

```
If(protocol=“icmp”, duration=”0”,service=”eco_i” or “ecr_i”, flag=“SF”, src_byte= “8”, dest_byte= “0” count= “1” or “2” or “46”, srv_diff_host_rate = “1” to “17”, dst_host_srv_diff_host_rate = “0” to “0.4”) then Attack= “nmap”

If(protocol= “udp”, duration= “0”,service=“private”, flag= “SF”, src_byte=”100” or “207” or “215”, dest_byte=”0” or “100” or “207”) then Attack=”nmap”

If(protocol= “udp”, duration=”0”,service=“private”, flag= “SF”, src_byte= “100” or “207” or “215”, dest_byte= “0” or “100” or “207”) then Attack= “nmap”

If(protocol= “icmp”, duration=”0”,service=“eco_i” or “ecr_i” or “urp_i”, flag=”SF”, src_byte= “20” or “37”, dest_byte= “0”) then Attack= “satan”

If(protocol= “udp”, duration= “0” or “4”, service=“domain_u” or “other” or “private”, flag= “SF”, src_byte= “1” or “5” or “19” or “40”, dest_byte=”0” or “1” or “4” or “5” or “26” or “28” or “74” or “444”) then Attack= “satan”

If(protocol=”tcp”, duration=”0” to “9”, flag=all except ”SF” and “OTH”, src_byte= “0” or “5” or “6” or “7” or “9” or “10” or “30” or “31” or “39” or “44” or “54” or “103” or “1710” dest_byte= “0” or “4” or “15” or “19” or “23” or “25” or “26” or “28” or “31” or “32” or “34” or “35” or “40” or “43” or “44” or “53” or “54” or “60” or “75” or “77” or “109” or “112” or “114” or “121” or “131” or “143” or “144” or “147” or “151” or “164” or “178” or “186” or “192” or “196” or “292” or “375” or “536” or “556” or “672” or “1405” or “1886” or “18056”) then Attack= “satan”

If (protocol= “tcp”, duration=”0” to “7” or “12743”, service=“ctf” or “domain” or “ftp_data” or “gopher” or “http” or “link” or “mtp” or “name” or “private” or “remote_job” or “rje” or “smtp” or “ssh” or “telnet” or “time” or “whois”, flag= “REJ” or “RSTO” or “SF”, src_byte=“0” or “4113”, dest_byte= “0” or “3” or “4” or “12” or “15” or “61” or “77” or “79” or “82” or “83” or “84” or “85” or “89” or “90” or “91” or “96” or “132” or “133” or “142” or “51633”, dst_host_count= “1” to “72”, dst_host_srv_count= “1” to “194”, dst_host_diff_srv_rate= “0” or “0.99” or “1”, dst_host_serror_rate= “0”, dst_host_rerror_rate= “0.5” to “1”, dst_host_srv_rerror_rate= “0.01” to “0.07” or “0.13” or “0.25” or “0.29” or “0.5” “0.67” or “1”) then Attack= “ipsweep”

If(protocol= “icmp”, duration=”0”,service=“eco_i” or “ecr_i” or “urp_i”, flag= “SF”, src_byte= “8” or “18”, dest_byte= “0”) then Attack= “ipsweep”
```

If the attacker sends probe requests to multiple hosts using a specific port, then this attempt recorded as portsweep attack.

6 Conclusion

The primary objective of the honeypot is to collect intense attack patterns and decode it into human understandable format. In this paper, we have implemented a virtual honeypot using honeyd which is installed on Ubuntu 14 machine and the attack patterns are captured whenever recommended by the IDS. The well-known probe attacking tools are used for attacking the system by us. The packets captured by the honeypot is decoded and converted into csv format for subsequent analysis. Finally, the patterns are processed and the snort rule set is updated to detect these type of attacks that may take place in future. It helps the administrator to protect the system from probe attacks and to analyze the signatures of the attacks.

References

1. Provos, N., Holz, T.: Virtual Honeypots: from Botnet Tracking to Intrusion Detection. Pearson Education, New Delhi (2007)
2. Quynh, N.A., Takefuji, Y.: Towards an Invisible Honeypot Monitoring System, Information Security and Privacy. Springer, Berlin (2006)
3. Chamotra, S., et al.: Deployment of a low interaction honey pot in an organizational private network. In: International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), IEEE, 2011
4. Kuwatly, I., et al.: A dynamic honeypot design for intrusion detection. In: International Conference on Pervasive Services, ICPS 2004. IEEE/ACS, IEEE (2004)
5. Alata, E., et al.: Lessons learned from the deployment of a high-interaction honeypot. arXiv preprint arXiv:0704.0858 (2007)
6. Das, V.V.: Honeypot scheme for distributed denial-of-service. In: International Conference on Advanced Computer Control, ICACC'09. IEEE (2009)
7. Li, S., Schmitz, R.: A novel anti-phishing framework based on honeypots. IEEE (2009)
8. Zhuge, J., et al.: Collecting autonomous spreading malware using high-interaction honeypots. In: Information and Communications Security. Springer, Berlin, pp. 438–451 (2007)
9. Spitzner, L.: Honeypots: Catching the insider threat. In: Proceedings of 19th Annual Computer Security Applications Conference, IEEE (2003)
10. Almotairi, S., et al.: A technique for detecting new attacks in low-interaction honeypot traffic. In: Fourth International Conference on Internet Monitoring and Protection, ICIMP'09. IEEE (2009)