HoneyComb : Enhancement to Honeypot Log Management

Lavenya
Department of Computer Science
Lovely Professional University
Punjab , India

Karanvir Kaur
Department of Computer Science
Lovely Professional University
Punjab , India

Abstract

Honeypot is a technology that helps us to discover the new types of hacking techniques from hackers and intruders or capture malicious content. With the aim to ease the administrator to handle files from multiple honeypot servers that are distributed in various locations at same time. Therefore this paper discusses the design and implementation of a prototype log management server for collection of log files from them. Information from honeypot servers will be sent in secure manner to log management server. The log management server provides the information to the administrator which runs an automated process to parse the information into database server. The result of analysis of information can be viewed by users through web interface.

1. Introduction

Organizations are nowadays dependent on their network infrastructures for providing necessary services. Also added growth to cyber crimes on the Internet has lead these organization to focus on security of their network. Firewall and Intrusion Detection Systems (IDS) are already in use for the security purposes. However they have certain limitations , such as being based on predefined signatures hence cannot capture new hacking techniques.

To overcome these limitations , Honeypots can be used instead. As per Wikipedia "honeypot is a trap that is set to detect , deflect or in some manner counteract attempts at unauthorized use of information system". It consists of a computer , a network site or data which appears to be part of the network but is actually isolated and monitored with the ability to simulate services (e.g:FTP , HTTP or POP3 etc). Honeypots are means of creating an inviatation to lure attackers with the purpose of studying the attackers and their attack patterns. Understanding these attack strategies , patterns and trends can be helpful in determining the vulnerabilities of a system. This requires the system to capture and log large amounts of data which are very difficult to process manually.

Traffic from different honeypot servers may share some common characteristics. With such large amount of data , comes the difficulty of handling and processing it. Firstly the collection of log files from honeypot servers deployed in different locations is a tedious job since administrator

has to manually login into each machine. Secondly , the analysis and the interpretation of information from numerous log files. In this paper we focus on low-interactive honeypot architecture . We choose to deploy the Honeypot server as it is easy to deploy. Our objectives are: a) To collect log files from from various Honeyd servers to a Honeypot log management server (web mail server). b) To provide a web-based interface to access the reports.

2. Classification of Honeypots

Based on the purpose of deployment Honeypots can be classified into two types : Production Honeypot and Research Honeypot. Only limited information can be captured by honeypots. Honeypots are placed inside a production network with other production servers to improve security is called Production Honeypot.There are two types of production honeypots : low-interactive honeypots and high-interaction honeypots. Example of them are Honeyd[2] developed by Niels Provos and Honeynet[3] respectively.

On the otherhand , Research Honeypot is run by a volunteer who gathers information about motives and tactics of the intruders and attackers. These are more complex to deploy and maintain , capture extensive information and is used primarily by research , military or government organizations. “Detecting and defending against Worm Attacks Using Bot-honeynet” is an example of this type.

Currently there are different categories of tools for Honeypots such as data collection tools , data analysis tools and the visualization tools. Reference[3] briefly introduces various honeynet data collection tools , techniques and research published in Honeynet project. The related work that needs to be mentioned here for the purpose of log analysis are : Honeyview and Honeydsum.

Honeydsum[6] is a log analyzer written in Perl by Brazillian Honeynet team. It can generate summaries from honeyd logs. Honeyview[7] is also a log analyzer whih can summarize log files in text or graphic mode. Moreover Honeydsum and Honeyview doesnot provide facility to automatically collect log files from multiple Honeypots distributed in different geographical locations.

3. Proposed Architecture

Collection of malicious activities from multiple honeypots , we design and implement a web mail server which acts as a central log management server. In this paper we consider using honeyd server as it’s a low-interactive honeypot implementation and easy to deploy. Our distributed honeypot architecture consists of two or more honeyd servers installed on computers as shown in figure 1.

Honeyd server functions to collect log files and send it to web mail server which is under the control of Honeypot Log Management server Administrator. Honeyd

collects two types of log files : Traffic Monitoring log and Service monitoring log.

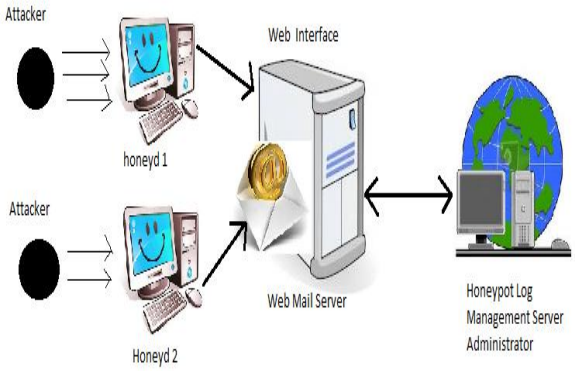

Fig 1. Proposed System

Honeyd server has to run python script in order to send the log files to web mail server. Honeypot Log Management server Administrator is responsible for fetching these log files from web mail server and parsing of these log files via BeeViewer tool that is deployed an Administrator system. In this paper we assume that all honyd servers that join our log management server use the same service script to generate the same format of log files[5].

Steps to use the underlying proposed system can be summarized as :-

1. The administrator of Honeyd Server installs and configures the Honeyd program and various service scripts.

2. The Honeyd server periodically itself sends the log files using smtp by running python script via crontab.

3. Web mail server receives logs in form of text files from various Honeyd servers.

4. Honeypot Log Management Server Administrator now downloads these logs for parsing.

4. Honeyd Server

Capturing incoming traffic is done by Honeyd Server. Types of logs generated are : traffic monitoring log and service monitoring log.

A. Traffic Monitoring Log

Record of Information of all incoming pakets is referred to as traffic monitoring log. Information includes the arrival time , protocol , source and destination IP address , source and destination ports and fingerprint. Fingerprints here represents the name and version of OS of the originated computer.

```
2012-07-18-10:22:24.2779 udp(17) - 178.125.184.50 14849 192.168.1.3 27458: 315
2012-07-18-10:22:24.9615 udp(17) - 192.168.1.3 27458 189.120.99.170 43611: 131 [windows 2000 RFC1323]
2012-07-18-10:22:24.9618 udp(17) - 192.168.1.3 27458 76.212.6.47 51413: 131 [windows 2000 RFC1323]
2012-07-18-10:22:25.3664 udp(17) - 76.212.6.47 51413 192.168.1.3 27458: 296
2012-07-18-10:22:25.9444 udp(17) - 192.168.1.3 27458 201.58.109.27 15152: 131 [windows 2000 RFC1323]
2012-07-18-10:22:25.9448 udp(17) - 192.168.1.3 27458 178.44.176.28 47842: 131 [windows 2000 RFC1323]
```

Fig 2. Example of Traffic monitoring Log.

B. Service Monitoring Log

It is used to keep activity logs for assigned services such as telnet log , HTTP log , FTP log and POP3 log. Once suspicious packets have been captured we will know which port and what information intruders have tried to access such as login name and password. Figure 3 illustrates example of service monitoring log file.

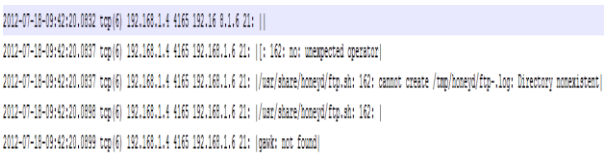

Fig 3. Example of Service Monitoring Log

Administrator can also emulate telnet , HTTP , FTP and POP3 services by running shell scripts which can be obtained from Honeyd webpage.More scripts for emulating services can be automatically generated by scriptgen tool[5]. Processes on Honey Server can be categorized as :

a) Transferring log files via smtp (emails).
b) Collection of log files.

C. Automatically transferring Log Files via Crontab

The Honeyd server will run the Cron file to send the log files to web mail server by running the python script, automatically at the specified times.

5. Web Mail Server

Function of web mail server is to receive the incoming log files in the form of text from the various Honeyd servers. The domain thehoney.net is written in ASP.NET and the administrator of the domain is responsible for the mails received and their mainatainance. The logs are received in the domain's email account of the administrator. Functioning of log management server is dependent on it.

6. Log Management Server

Logs are retrieved by administrator from the web mail server. The BeeViewer tool has been developed in C#.NET for admin's ease in handling logs. Processes on Honeypot Log Management Server are : a) Retrieval of Logs b) Reformatting of logs c) Database storage and d) Report generation.

A. Retrieval of Logs

The administrator of Honeypot Log Management server has to manually retrieve the log files from the web mail server. Also has to select the log type before the tool can begin its automated processing.

B. Reformatting of Logs

After the logs are retrieved from the web mail server , the raw log files are reformatted by BeeViewer tool that runs on Honeypot Log Management Server. BeeViewer translates the raw logs files to data which is then inserted on SQL database.

C. Database Storage

After reformatting the information on each column of raw log files is inserted into SQL database table based on log type. Different log types supported in this work are : a) Honeyd Traffic Log and b) Honeyd Service Log.

D. Generating Reports

BeeViewer is also responsible for the analysis of reformatted log files. The latest information is then displayed on the web interface which is prominent information for users.

7. Experiments and Results

This prototype system is implemented on Ubuntu 10.10 with honeyd version 1.5c , SQL Server 2008 , .NET3.5 and three libraries namely libevent,libdnet and libpcap. To check whether the system is functioning properly and can be viewed over the network, we use Nmap fingerprinting tool[*]. Nmap scans the targeted network or system IP and returns the information related to it such as open ports and services available.

Figure 4. shows the service log on honeyd server when an FTP connection attempt is received. This attack is redirected towards the FTP script emulated by honeyd server resulting in generation of fake FTP output.

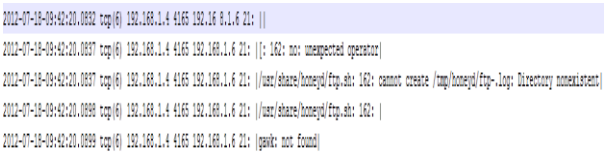

Fig 4. FTP attack redirected to FTP.sh script

Logs from many different Honeyd servers are sent to domain's administrator email account via python (smtp) script. Figure 5. Shows the list of logs received in the domain's account.

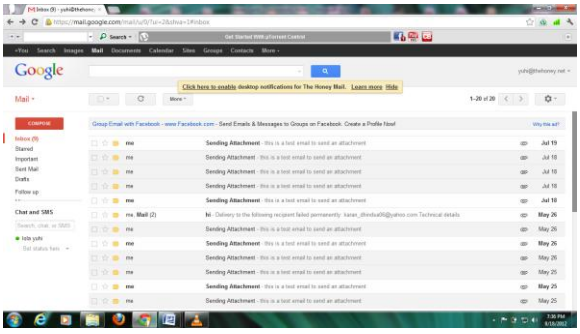

Fig 4. Honeyd logs received in the domain's account

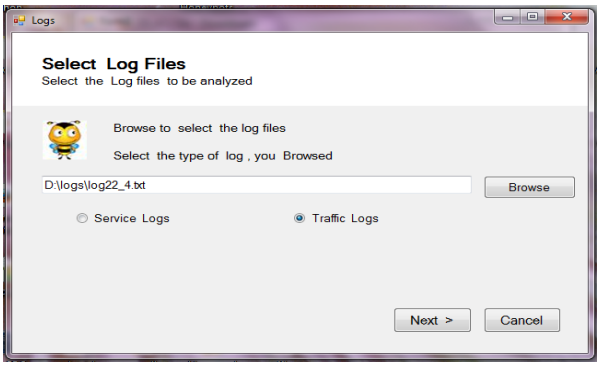

Fig 5. Bee Viewer-log parser interface

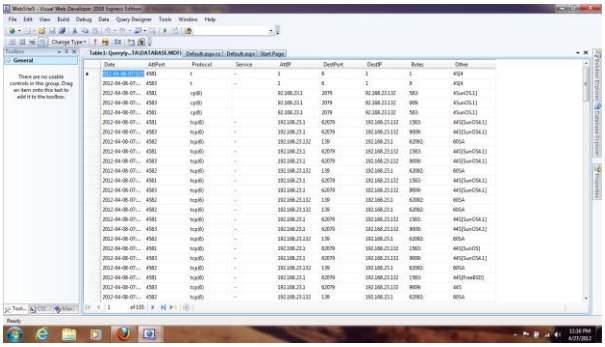

Fig 6. Tabular result generated by BeeViewer log parser.

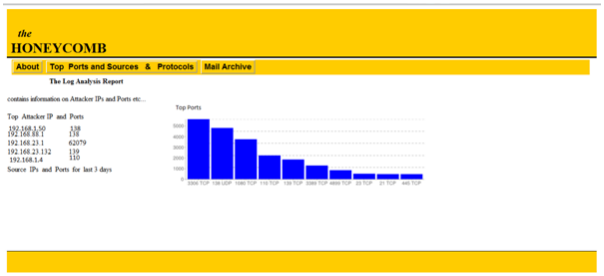

Fig 7. The result generated can be visualised on domain's site.

8. Conclusion

This paper aims to create a improvised prototype system for management of distributed Honeypot log management.Here, Logs are transferred automatically by the honeyd server in form of emails. These logs

are then processed by BeeViewer log parser on the Honeypot log management server. Here the tool BeeViewer has been developed in .NET and database has been maintained in SQL Server 2008.

9. References

[1] Vasaka Visoottiviseth, Uttapol Jaralrungroj, Ekkachai Phoomrungraungsuk and Pongpak Kultanon, "Distributed Honeypot Log Management and Visualization of Attacker Geographical Distribution", Eighth International Joint Conference on Computer Science and Software Engineering(JCSSE), 2011.

[2] N. Provos, "A virtual honeypot framework", In Proceedings of the 12th USENIX Security Symposium, August 2004.

[3] David Watson and Jamie Riden, "The Honeynet Project: Data Collection Tools, Infrastructure, Archives and Analysis", WOMBAT Workshop on Information Security Threats Data Collection and Sharing, April 2008.

[4] Distributed Honeypots Project, http://honeytarg.cert.br/honeypots/

[5] Script Contribution, http://www.honeyd.org/contrib.php

[6] Brazillian Honeynet Project, "honeydsum.pl". http://www.honeynet.org.br/tools/.

[7] HoneyView – a honeyd Logfile Analyzer, http://honeyview.sourceforge.net/