An Analysis of Honeypots with special reference to Honeyd

Ebenezer A. Laryea[#1], Haridas Acharya[*2], Winfred Larkotey[#3], Joseph Abandoh-Sam[#3]

[#1,3,4]*Institute of Computer Science,* [#2]*Symbiosis Institute of Computer Studies & Research,* [#1,3,4]*Valley View University,*
[#2]*Symbiosis International University*
Oyibi, Accra-Ghana, Pune, India

[1]afotey@vvu.edu.gh
[2]haridas.undri@gmail.com
[3]larkotey@vvu.edu.gh
[4]abandoh@vvu.edu.gh

***Abstract*— It's a herculean task for system administrators and normal web users to patch up their computers and keep all the software up to date. And in the case of Zero Day [5] attacks there is very little which he can do to protect his system. One way to get early warnings of new vulnerabilities is to setup monitoring computer system on a network where we expect that the tendency of breaking in is high. There are various methods of doing this. One popular way is by using IDS and IPS. But there is another method, which is gradually becoming quite popular in the security domain. This new systems are called Honeypots. Honeypots are systems, which are intentionally, setup to bait hackers and learn their modus operandi. This paper takes a look at the setup of such a system and the type of information that can be gleaned from such a system.**

***Keywords*— Honeypot, Honeyd, Cyberoam, Put your keywords here, keywords are separated by comma.**

I. INTRODUCTION

"According to the Computer Security Institute and the FBI's joint survey, 90% of 643 computer security practitioners from government agencies, private corporations, and universities detected cyber attacks last year. Over **$265,589,940** in financial losses was reported by 273 organizations."[1]. People are now more interested in breaking into computers and networks in order to get some financial benefits. Over the last 3 years almost all the major leading banks around the world have had to face a successful break-in by an intruder!

Security on the Internet is becoming quite important nowadays as more and more business start moving online. With this current trend, ensuring the security of the online business is a herculean task! Each day hundreds or thousands of worms and viruses are released into the cyber world. At the same time thousands of attacks are launched on computers worldwide by "hackers". Even the FBI States, "Cyber attacks pose the greatest threat to the United States after nuclear war and weapons of mass destruction - and they are increasingly hard to prevent"[2].

II. LITERATURE REVIEW

"A Honeypot is a computer system on the internet that is expressly set up to attract and "trap" people who attempt to penetrate other peoples computer systems"[6] We have basically two types of Honeypots, they are physical Honeypots and virtual Honeypots.

A. Types of Honeypots

Physical Honeypot: This is a physical computer with its own IP address and running various services.
Virtual Honeypot: Is a simulated machine with has been assigned different behaviours and one of these behaviours is its ability to respond to network traffic. Multiple virtual Honeypots can be simulated on a single system. Generally virtual Honeypots are preferred because they require fewer physical systems, which directly reduce maintenance cost and also make it easier to manage in the long run.

Apart from the physical Honeypots and virtual Honeypots, we can further classify them into two other groups. This classification is based on their level of interaction. They are Low interaction Honeypots, Medium interaction Honeypots and High Interaction Honeypots.

B. Benefits of Honeypots

Honeypots have several advantages, which are unique. Some of them are discussed below.

Data Value: One of the main challenges of the most corporations is gaining value from data, They collect large amounts of Data everyday and store them, But most of them time they don't interpret them to see what value they hold. But in the case of Honeypots, this changes. Since a Honeypot logs the entire information coming to it. Honeypots generate a large amount of date. This data tells the user the various types of attacks the Honeypot experienced during a given time period. Also unlike most IPS or IDS, Honeypots give you the precise information of what you want in a very quick and simple format which you will understand.

Simplicity: An Honest survey of Honeypots show that it's very simple to use. There are no complicated algorithms, no signature databases to update, no rule bases to mis-configure. All you have to do is to take the Honeypot, and then just connect to a part of your network and relax. What you have to bear in mind is that you only have to check the Honeypot when any data comes to it because it is only malicious activities which it captures. Most security professionals would say the more simple a system is the more reliable it is because

with complexity comes mis-configurations which leads to failure.

Resources: Another challenge most security mechanisms face is resource limitations, or even resource exhaustion. Resource exhaustion is when a security resource can no longer continue to function because its resources are overwhelmed. For example, a firewall may fail because its connections table is full, it has run out of resources, or it can no longer monitor connections. This forces the firewall to block all connections instead of just blocking unauthorized activity. An Intrusion Detection System may have too much network activity to monitor, perhaps hundreds of megabytes of data per second. When this happens, the IDS sensor's buffers become full, and it begins dropping packets. Its resources have been exhausted, and it can no longer effectively monitor network activity, potentially missing attacks. Another example is centralized log servers. They may not be able to collect all the events from remote systems, potentially dropping and failing to log critical events.

TABLE I
TYPES OF HONEYPOT SOFTWARE

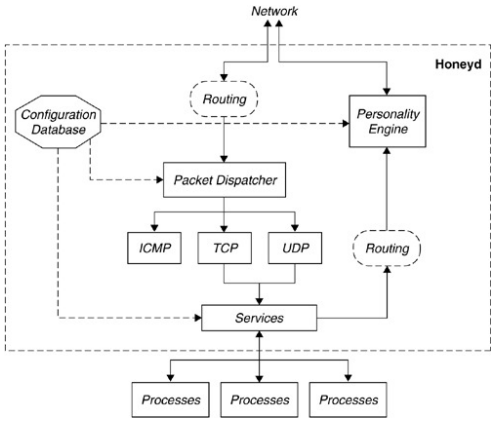
NS	D	L	P
Argos	Argos is a full and secure system emulator designed for use in honeypots.	Open Source	Linux
Back Officer Friendly	BOF is a useful little burglar alarm It identifies attacks from Back Orifice, one of the nastier hacking applications, as well as other sorts of scans.	Proprietary	Linux
Bait N Switch Honeypot	BNSH is a multifaceted attempt to take honeypots out of the shadows of the network security model and to make them an active participant in system defense	Open Source	Linux
FakeAP	Fake AP generates thousands of counterfeit 802.11b access points. As part of a Honeypot. Fake AP confuses Wardrivers, NetStumblers,	Open Source	Linux
GHH - The "Google Hack" Honeypot	It is designed to provide reconnaissance against attackers that use search engines as a hacking tool against your resources	Open Source	Linux
HoneyBot	HoneyBOT is a Windows based medium interaction honeypot solution. HoneyBOT works by opening over 1000 udp and tcp listening sockets on your computer and these sockets are designed to mimic vulnerable services	Open Source	Windows
Honeyd*	Honeyd is a small daemon that creates virtual hosts on a network. The hosts can be configured to run arbitrary services, and their personality can be adapted so that they appear to be running certain operating systems	Open Source	Linux
HoneyPearl	Honeypot software based on perl with many plugins like fakehttp, fakesmtp, fakesquid, faketelnet, etc.	Open Source	Linux
Honeypoint	HoneyPoints are flexible pseudo-server applications that are able to emulate thousands of real services such as web, email, database systems and others.	Proprietry	Linux, windows, Mac

Table Key: NS- Name of Software
D- Description
L- License
P- Platform

III. WORKING OF HONEYD

Honeyd is an Open source Honeypot designed by Niels Provos. It was initially designed to be used on the linux operating system. Currently it's been used on both Linux and Windows. The Primary purpose of Honeyd is for detection. It's main job is to detect unauthorized activity within your organization or network. It does this by monitoring all the unused IP's in your network. Therefore any attempt to connect to an unused IP is considered to be a malicious activity. It's very likely that when there is any activity on these unused IP's it could me that there's an attack on your network, IP scan or some worm activity going on in your network.

Honeyd will monitor all the unused IP addresses in your network and whenever it notices an activity on a particular IP addresses It assumes the identity of that IP address and then tries to interact with the hacker. Therefore anytime honeyd generates and alert you know that it's a real attack. Generally Honeyd can detect and log TCP,UDP and some ICMP packets activity. You do not necessarily need to create a port listener or specific service. Honeyd does this automatically. Also it receives the requests been sent to the unassigned IP address and starts responding to the hacker. All the interactions with the intruder during this time are logged for later analysis. Honeyd generates fewer alerts as compared to other software. With Honeyd you can create emulated scripts to interact with the hacker so that you can try to learn what they are attempting to do. The emulated services are limited because they usually act in a predetermined behavior. The scripts which are supposed to interact with the attacker can be written in any language such as Perl, shell. Honeyd also has about seven emulated services you can choose from. An example is a service which acts like a Cisco router.



IV. WORKING OF HONEYD

Honeyd is designed to respond to packets which are sent to network addresses to which it is configured. For honeyd to receive any packets sent to it, it should be properly configured. There are several ways of doing this but one of the most common is creating special routes to the virtual IP. This illustration will explain how to do this.

Assume A is the IP address of the router, and B is IP address of the host running honeyd. In this case the IP address of the Honeypot lies within our local network. We denote them as V_1 …., V_n. Hence when an attacker sends a packet from the internet to Honeypot V_i, the router A receives and attempts to forward the packer. The router searches its routing table to find the address to which it should forward the packet to. Basically there are three possible outcomes,

The router drops the packet because there is no root to the destination

The router forwards the packet to another router

Or V_i lies in the same local network range of the router therefore it can be directly reached by A which is the router.

To direct the packets meant for Vi to b, we can configure routing entries for V_i, with 1< I < n that points to B/ In that case the router will just forward packets to our virtual Honeypot.

V. ROUTING OF HONEYD

Honeyd is successfully capable to simulate arbitrary virtual routing topologies to deceive attackers and network mapping tools. The virtual routing topologies have an entry point for packets and depending on the topology there can be several entry points. When the framework receives a packet it finds the current entry in the routing tree and traverses it, starting from the root until it finds a node that contains the destination IP address of the packet. Also the Honeyd framework decrements the TTL *(Time to Live)* field of the packet. If the TTL of the packet reaches zero, then an ICMP message is sent back with the message "*time exceeded" With* the IP address of the router that caused the TTL to go to zero.

VI. PLACEMENT OF HONEYD ON UNIVERSITY NETWORK

Cyberoam is Unified Threat Management system used by the University. It serves as a firewall, anti-spam, and gateway level antivirus, for the university. We decided to design how the honeypot should be placed so that it doesn't interfere with the working of Cyberaom and vice versa

VII. SETTING UP HONEYD

For the experiment honeyd wad downloaded from the ubuntu 8.10 repository. The command for downloading and installing honeyd in Ubuntu is,

sudo apt-get install honeyd

This command will download and install honeyd (Note: sometimes the version available in the Ubuntu Repository may not be the most current version).

Configuration: Once honeyd has been successfully installed the next task is to configure the various honeypots which you want to setup.. Honeyd uses a simple text-based configuration file to specify which IP addresses to run virtual honeypots. The configuration also specifies which services are available for each host. Some samples of configuration files are given below.

```
create routerone
set routerone personality "Cisco 7206 running IOS 11.1(24)"
set routerone default tcp action reset
add routerone tcp port 21 "webserver/ftp.sh"
bind 10.1.0.1 routerone
```

Above is an example of a honeyd configuration file. The above line is called a template. A template refers to a completely configured computer system. One of the first steps in creating virtual honeypots is to configure a template for each different computer system. Once you have designed a template you assign it an IP address to bring it up. A virtual honeypot has all the characteristics of the assigned template, including operating system behaviour and which services should run on each port

Reasons for selection of Operating systems and services:

A little research was done before setting up the operating systems for the virtual Honeypots. One key factor which a Honeypot should have is that it, should be configured in such a way that it's quite enticing for any hacker. It was realized that a virtual Honeypot can be setup which will be running three different operating systems. Windows, Linux and Macintosh were chosen so that all the popular brands off operating system could be covered.. Also since the Honeypot was being setup in a college environment, it would be a surprise to a hacker to find these three different operating systems in the college's network.

To make it a little bit easier for the hacker, the Honeypots setup were running the most vulnerable versions of the operating system and with the most popular ports, Open. We got the list of operating systems and Port vulnerability from the respective websites. For apple the information was available at [9], for Microsoft. It was available at [10] and for Linux It was also available at [11]. The table below gives the operating system and their respective vulnerabilities.

VIII. ISSUES ARISING IN THE DEPLOYMENT OF HONEYPOTS

Entrapment: Entrapment can be legally defined as "The act of government agents or officials that induces a person to commit a crime he or she is not previously disposed to commit."[18] Since the main functionality of a Honeypot is to engage hackers and people with malicious intent the issue of Entrapment arises. Well for this law to be used against you, there are two conditions, Firstly, you the owner of the Honeypot should be a government agency and also you should have promoted your Honeypot in such a way it attracted the person. So ensure that you don't promote it, naturally the hackers will come

Privacy: This is something you actually have to worry about a little because there are a lot of legal issues regarding privacy and it's easy to be entangled in at least one of them. The laws concerning privacy are somehow contradictory so you have to be very cautious. But so far there have been no cases filed against Honeypot owners.

Attacks on Third Parties: There is a potential of your Honeypot being compromised and then used a launch point to attack a third party. To ensure that this doesn't happen to your Honeypot, ensure that you have configured your Honeypot well. As far as possible limit outgoing traffic from your Honeypot. Also regularly monitor your honeyd logs and your system logs to ensure there is no "malicious" activity going on unless what you have already permitted. Also ensure that your base is secure. That is the operating system you are running the Honeypot is well updated and patched with the latest security and antivirus updates. You can also setup a firewall which will block outgoing traffic from your Honeypot and inform you whenever it receives any outbound data from the Honeypot.

IX. ISSUES ARISING IN THE DEPLOYMENT OF HONEYPOTS

During the process of setting up and configuring Honeyd, several issues were encountered The teething ones are listed and explained below.

Insufficient Documentation & Support: Since the Concepts of Honeypots is not really common or popular it was really difficult to get queries and doubts solved. There is forum for honeyd where users can post their queries. But the form is not active. Hence it takes several weeks or in some cases a month before anybody responds to the queries posted on the forum.

Also there are very few books which talk about the Honeypot concepts therefore trying to get adequate literature for my research was really tough. Since the Internet is not a reliable or respected source of information. The little information gleaned from it could not be very reliable.

Difficult to use and configure: Honeyd is mainly configured through the command line hence to be able to work with it you should have knowledge of how to use the command line and also be conversant with most linux commands. Also if a Graphical User Interface is designed for Honeyd then it would make it easier for a beginner to configure and monitor and analyze how it works. Also since

Operating System	Version	Service	Port	Vulnerability
Mac OS X Server	10.3.9	Apache 2	80 TCP	The htdigest program contains a buffer overflow, which, if used improperly in a CGI application, could allow a remote system compromise
Mac OS X Server		Directory Services	625 TCP	A buffer overflow in Directory Services could lead to remote execution of arbitrary code
Mac OS X Server		kerberos	749 TCP/ UDP	An authenticated user could execute arbitrary code on the KDC host, compromising a Kerberos realm.
Mac OS X Server	10.4.2	Mail	25 TCP	Using Kerberos Version 5 for SMTP authentication Mail.app may disclose sensitive information
Mac OS X Server	v10.3.9	Mysql	3306	Multiple vulnerabilities in MySQL, including arbitrary code execution by remote authenticated users
Windows Server	2003	SMB	445 TCP	An attacker can execute code with "SYSTEM" privileges
Windows Server	2003	IIS	80 TCP	A WebDAV vulnerability allows attackers to access password-protected directories and download and even upload arbitrary files
Ubuntu Linux	8.1	Apache	80 TCP	underflow flaw in apr-util as included in Apache. An attacker could cause a denial of service via application crash in Apache using a crafted SVNMasterURI directive, .htaccess file, or when using mod_apreq2
Ubuntu Linux	8.1	Tomcat6	8080 TCP	Tomcat did not properly normalise paths. A remote attacker could send specially crafted requests to the server and bypass security restrictions, gaining access to sensitive content
Ubuntu Linux	8.1	Cups	631 UDP	CUPS did not properly handle certain network operations. A remote attacker could exploit this flaw and cause the CUPS server to crash, resulting in a denial of service

the debug message s are displayed on the screen. If they could be captured and displayed appropriately on a screen it would make it easier for a system administrator to respond to the error messages or other messages which may appear on the screen.

Unable to Attain DHCP: One of the major setbacks in the research was that honeyd was un-able to get a DHCP from the DHCP Server on the university network allow my Honeyd setup to be integrated into the college network. Within the college network all the computers are assigned an IP address through a DHCP server. The setup of Honeyd so that it would be able to receive an IP address through the DHCP server. The plan was to make the IP address assigned to Honeyd permanent once it gets assigned. This would be done by configuring the DHCP server to always assign a particular IP address based on the MAC address.

One Honeyd was setup to get it's IP address through DHCP it didn't work, it gave an error that "after 12 tries dhclient is not able to obtain an IP address". A question was therefore posted on the Honeyd Forum but got no reply as at the time of writing this. An extensive search was done and found out that some people were also facing the same problem. There were two or three people who were successfully able to assign an IP address through DHCP but when the steps given were followed they had given it still got the same error message. Those who were not able to receive the IP address were all saying that there was a bug issue in the latest version of Honeyd which is Honeyd 1.5c There was no error log generated per say so an analysis could not be done.

X. CONCLUSIONS

Overall Honeypots is a very interesting and useful concept, which should be adopted and implemented more often and in more places. Also the data which is collected for the various Honeypots setups can be shared with other Honeypot groups so that useful analysis can be done on the data. If the data is collected by one organization is kept and not shared, then the real benefits of having a Honeypot may not be fully achieved. Once the data is shared around, different interpretations and analysis can be done on it and the modus operandi of attackers can be more understood and various organizations can secure themselves.

Personally I would encourage all major organizations to implement Honeypots in their networks and track out the activities which go on within their networks. Doing so would help them reduce costs in controlling security breaches from within and also when purchasing security equipment they would be more informed of the threats they face within their organization and thereby be able to make more useful and beneficial purchases for their organizations.

ACKNOWLEDGMENT

I would like to thank Prof Haridas Archarya for assistance in doing this research. I would also like to acknowledge Dr. Ronald Zane and Prof Seth A. Laryea

REFERENCES

[1] http://linux.omnipotent.net/article.php?article_id=11505
[2] http://www.thearynews.com/english/newsdetail.asp?nid=19868
[3] http://fcw.com/Articles/2009/02/17/CERT-cyber-incidents.aspx
[4] http://www.us-cert.gov/cas/techalerts/TA09-088A.html
[5] http://en.wikipedia.org/wiki/Zero_day_vulnerability
[6] http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci551721,00.html#
[7] Lance Spitzner. Honeypots: Tracking Hackers. Addison Wesley Professional, Sept, 2002.
[8] Addison.Wesley.Virtual.Honeypots.From.Botnet.Tracking.to.Intrusion.Detection.Jul.2007
[9] http://support.apple.com/kb/TA23465?viewlocale=en_US
[10] http://www.sans.org/top20/#s3
[11] http://www.sans.org/top20/#s2
[12] Table 1: List of various Honeypot softwware.
[13] Figure 1: Honeyd Design Overview
[14] Figure 2: General view of How honeyd works in a network
[15] Figure 3: A TCP Packet header
[16] [Figure 4: Honeypot placed before cyberoam
[17] Figure 5: Honeypot placed after cyberoam
[18] Compact Oxford Dictionary, Thesaurus and Word power Guide, Oxford University Press, 2006