

Honeypot: A Survey

¹Neha Sahu, ²Vineet Richhariya

^{1,2}Dept. of Computer Science, LNCT Bhopal, MP, India

Abstract

Information security is a growing concern today for organizations and individuals alike. This has led to growing interest in more aggressive forms of defense to supplement the existing methods. One of these methods involves the use of honeypots. A honeypot is a security resource whose value lies in being probed, attacked or compromised. It is an outstanding technology that helps us to learn new hacking techniques from attackers and intruders. The much information we collect from multiple Honeypot servers, the more exact attack patterns we can generate and we can also try to find out the location of attacker. In this paper we present an overview of honeypots and provide a starting point for persons who are interested in this technology. We examine different kinds of honeypots, honeypot concepts, and approaches in order to determine how we can recommend measures to enhance security using these technologies.

Keywords

Classification of Honeypot, Distributed Honeynet, Generations of Honeypot, Honeynet, Honeypot Tools

I. Introduction

In recent years, there has been a growing interest in information protection and security for large organizations. This has led to a growing demand for more aggressive forms of security to complement the existing techniques. One of these security methods involves the use of honeypots and honeynets.

Honeypot is an outstanding technology that helps us learn new hacking techniques from attackers and intruders. The much information from multiple Honeypot servers, the more appropriate signatures or attack patterns we can generate. To ease the administrator to manage and monitor trace files from multiple points, Honeypot servers are distributed in various locations.

A. What Is Honeypot

According to Lance Spitzner [1], founder of the Honeynet Project, a honeypot is a system designed to learn how “black-hats” probe for and exploit weaknesses in an IT system. It can also be defined as “an information system resource whose value lies in unauthorised or illicit use of that resource”. In other words, a honeypot is a decoy, put out on a network as bait to lure attackers. Honeypots are typically virtual machines, designed to emulate real machines, acting or 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. The fig. 1, below shows the Honeypot colored darker.

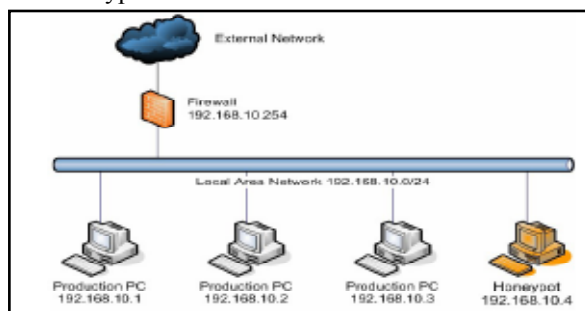


Fig. 1: Deployment Scenario of a Single Honeypot

It is not registered in any naming servers or any other production systems, i.e. domain controller to hide its strength. This is important, because only within an accurately configured network, one can suppose that every packet sent to the Honeypot, is suspect for an attack. If disarranged packets arrive, the amount of false alerts will increase and the value of the Honeypot drops.

B. Working of Honeypot

A honeypot [2], works by fooling attackers into believing it is a legitimate system; they attack the system without knowing that they are being observed completely. When an attacker tries to compromise a honeypot, attack-related information, such as the IP address of the attacker, will be collected.

A critical section to any honeypot is data capture, the ability to keep log, alert, and capture everything the attacker is doing. It can also capture the packets and packet payloads involved in the attack. This information can prove valuable in analyzing the attackers' activities.

The rest of the paper is organized as follows. In the section II classification of honeypot have been explained. Advantages and disadvantages of honeypot have been discussed in section III. In section IV, we have discussed the basic three generations of honeypot. In section V, we have explained related work and some popular honeypots with their advantages and disadvantages. Finally section VI, describes the concluding remarks.

II. The Specific Classification of Honeypot

Since the Internet come to arrive, many security experts have been studying the honeypot technology of application and deployment. And they have classified the honeypot of technology. According to the Design Deployment of the Honeypot, Honeypot can be divided into Production and Research Honeypot.

A. Production Honeypot

These honeypots are used to protect the organization network, recognize attacks from external intruders and possibly delay or stop the intrusion trials on the production systems to diminish risks in organization. It captures only limited information. It is placed inside the production network with other production servers like firewall to improve their security. The purpose of a production Honeypot is to reduce risks, and add values to the security measures of an organization.

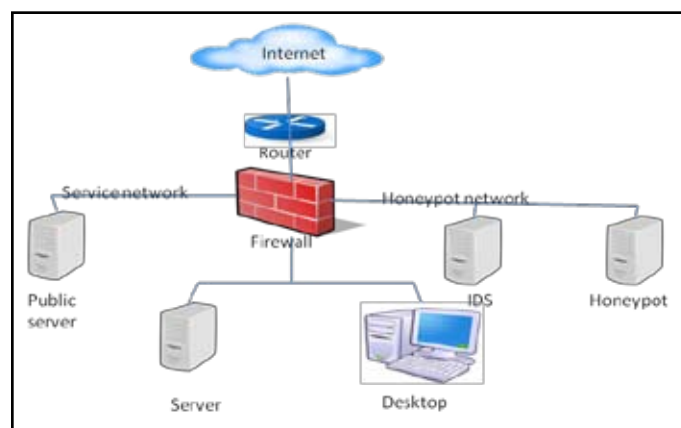


Fig. 2: Production Honeypot

B. Research Honeypot

This Honeypot is run by a volunteer who gathers important information about the motives and methods of intruders and attackers. It is used to research the threats organizations face, and to learn how to provide better protection against those threats. Research Honeypot is more complex to deploy and maintain, capture extensive information, and is used primarily by research, military, or government organizations.

“The Worm Propagation Model and Control Strategy Based on Distributed Honeynet” and “Detecting and Defending against Worm Attacks Using Bot-honeynet” are examples of research Honeypot.

According to the Honeypot with Different Attacker Interaction Level, we may divide honeypots into three major classes: low-interaction, medium interaction, and high-interaction. The interaction level of a honeypot is directly related to the amount of data that can be collected from intrusions.

C. Low Interaction Honeypot

Low interactive honeypot has lower interaction performance, and it is used for simulating the specific function or service which is running in the existing system, attackers can only have movement in this controlled range. Of course, in a good design of honeypot, attacker should not know that he is tracked. But in such low interactive honeypot range, it is very difficult to really hide themselves and not let attackers found himself having entered a well designed trap. Example of this type of honeypot is Honeyd [3] and BackOfficer Friendly [1].

D. Medium Interaction Honeypot

The middle interactive honeypots also carry out a certain degree of simulation; they use some Script or small application programs to mimic some services or functions. They can obtain more information, and have stronger concealment than low interactive honeypots. They offer more ability to interact with intruder than do low-interaction honeypots but less functionality than high-interaction honeypots. Examples – mwcollect, nepenthes [4], honeytrap [5].

E. High Interaction Honeypot

High interactive honeypots are constructed by real operating system, and provide a real operating system for attackers. High interactive honeypot allows attackers running all the instructions in the real operating systems, and make attackers rarely feel himself in a honeypot system. This kind of honeypot, is easier to gather useful information to the benefit of analysis attackers behavior. They can obtain and store a great amount of data since the attacker has a great deal of interaction with the honeypot. However, the more flexibility the attacker has, the more risk is involved. Based on the assessments and evaluations of different studies.

Table 1: Low, Medium and High Interaction Honeypot Comparison

	Low interaction honeypot	Medium interaction honeypot	High interaction honeypot
Installation and configuration	Easy	Involved	Difficult
Redesign and development	Easy	Involved	Difficult
Maintenance	Easy	Involved	Difficult

Information gathering	limited	Variable	extensive
Level of risk	Low	Medium	High
Resource utilization	Low	Medium	High
Compromised wished	No	No	Yes
Knowledge to develop	Low	High	Medium high
Knowledge to run	Low	Low	High
Implementation cost (hardware,licenses.... etc)	Low	Low medium	Medium high
Real OS	No	No	Yes

Table 1, summarizes all the benefits and shortcomings of each level of interaction in honeypots. An example of this type of honeypot is Honeynet [6], and is typically used for research purposes.

F. Honeynets

Groups of honeypots are called Honeynets [6]. A honeynet is simply a network that has one or more honeypots, where each honeypot can be considered as vigilance and early-warning unit, and can take many forms, such as data records or files, or even unused IP addresses. These honeypots may be physically located in the environment, or virtually situated using different virtualization techniques, such as VMware, XEN ...etc.

To successfully design a honeynet, you must correctly arrange the honeynet architecture [6]. The key to the honeynet architecture is called a honeywall. This is a gateway device that detaches your honeypots from the rest of the world. Each traffic going to or from the honeypots has to go through the honeywall. This gateway is generally a layer 2 bridging device, meaning the device should be invisible to anyone interacting with the honeypots. Below we can see in fig. 3, which describes this architecture. Our honeywall has 3 interfaces. The first 2 interfaces (eth0 and eth1) are what dispatch our honeypots from everything else, these are bridged interfaces that have no IP stack. The 3rd interface (eth2, which is optional) has an IP stack allowing for remote administration.

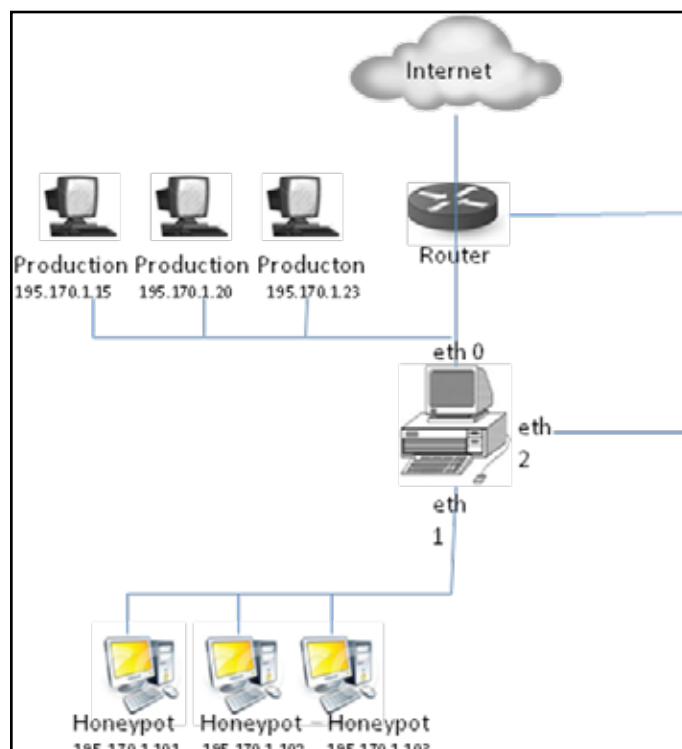


Fig. 3: Deployment of Honeynet

G. Distributed Honeypot

One major limitation of honeypots is that must take up a large segment of the address space in order to be effective and useful (since malware and attackers must target the honeypots). Aiming at the defects of the existing honeypot system, the distributed honeypot [8] system is put forward. Yang et al. provide a distributed framework for grid computing in which legitimate hosts redirect suspicious users to a single honeypot. For example consider a distributed system shown in figure 6 based on [8]. The System works under the unified management by central control system, and various agents distribute in several network area to realize the efficient network area coverage. Various agents are mainly responsible for a message interception of network intrusion; while control center is responsible for analyzing the information stored by the agents.

1. Honeytokens

A honeytokens is a data entity whose value lies in the inherent use of that data. Simply put, a honeytokens is a fake digital entity that can have many different applications. The term “honeytokens” was coined in 2003 by Augusto Paes de Barros [9], honeytokens are entities such as false medical records, incorrect credit card numbers and invalid social security numbers. How honeytokens work can be found in [10].

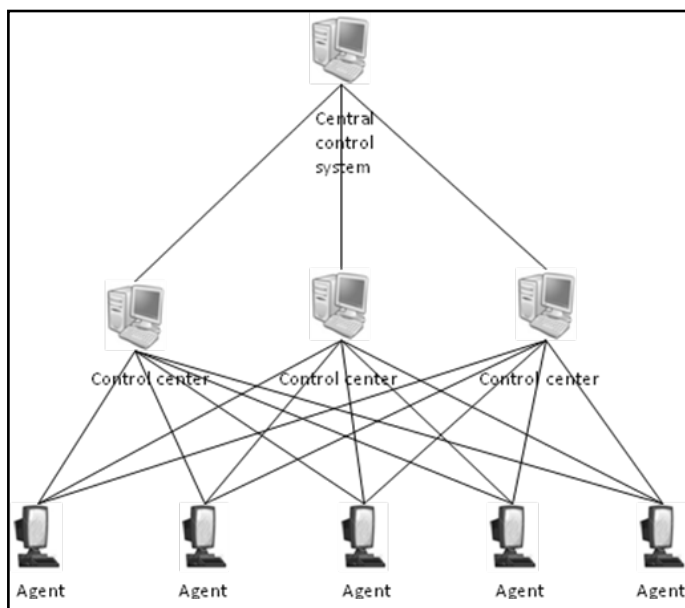


Fig. 4: Distributed Honeypot

III. Advantages and Disadvantages of Honeypot

A. Advantages

Honeypots are enormously simply concept, which gives them some very powerful advantages when compared to the currently used security mechanism:

1. Small Data Sets of High Value

Honeypots gather small amounts of information. Instead of logging a one GB of data per day, they can log only one MB of data per day. Instead of producing 10,000 alerts a day, they can produce only 10 alerts a day. Remember, honeypots only capture bad activity, any interaction with a honeypot is most likely unauthorized or malicious activity. As such, honeypots reduce ‘noise’ by collecting only small data sets, but information of high value.

2. New Tools and Techniques

Honeypots are made to capture anything thrown at them, including tools or techniques never seen before.

3. Minimal Resources

Honeypots require minimal resources because they only catch bad activity. For example an old Pentium computer with 128MB of RAM can easily handle an entire class B network sitting off an OC-12 network.

4. Encryption or IPv6

Unlike most security technologies (such as IDS systems) honeypots can work fine in encrypted or IPv6 environments. It does not matter what the attackers throw at a honeypot, the honeypot will detect and analyze it.

5. Information

Honeypots can collect in-depth only valuable information that is few, and any other technologies cannot match.

6. Simplicity

Finally, honeypots are basically very simple. There are no fancy algorithms to establish, state tables to keep up, or signatures to update. The simpler a technology, the less chances of mistakes.

7. Cost

Honeypots are fairly not expensive. Some simple versions can be downloaded for free.

B. Disadvantages of Honeypot

Like any other technology, honeypots also have some weaknesses. Because of this they do not replace any current technology, but can easily work with existing technologies.

1. Narrow Field of View

The greatest challenge honeypots face is their narrow field of view. Honeypots can only watch activities directed against them, so if an attacker is able to crack into a network and attack multiple systems, the honeypot would not be able to recognise it unless it was directly attacked.

2. Fingerprinting

It occurs when an attacker detects a honeypot by identifying certain behaviors that they particularly would be looking for. If a honeypot is set up wrongly, an attacker could identify this and avoid it as well as others. Something as simple as a misspelled word could trigger a fingerprint to the attacker.

3. Risk

The third disadvantage of honeypots is risk. If an attacker identifies a honeypot, they could gain control and attack the organization. Also if a honeypot is compromised, it could be set up to attack other systems.

IV. Generations of Honeypot

The idea of honeypots began in 1991 with two publications, “The Cuckoos Egg” and “An Evening with Breford”. “The Cuckoos Egg” by Clifford Stoll was about his experience catching a computer hacker that was in his corporation searching for secrets. The next publication, “An Evening with Berferd” by Bill Chewick is about a computer hacker’s moves through traps that he and his colleagues used to catch him. In both of these publications were

the beginnings of what has now become honeypots. The first type of honeypot was released in 1997 called the Deceptive Toolkit. This kit was used for deception to attack back. In 1999 the HoneyNet Project was founded. The non-profit group dedicated itself to research the blackhat community and to share the results with others. HoneyNet project gave three generations of Honeypot as follows:

A. Gen I Honeypot

It was developed in 1999. These honeypots served as a proof of concept and were very easy to setup and deploy. They had only basic mechanisms for satisfying data control and capture requirements. The architecture of GenI honeypots is shown in Figure 5. The data control requirement in GenI honeypots is delivered by a reverse firewall. The data capture component in GenI honeypots is done by an Intrusion Detection System (IDS) which has two major roles.

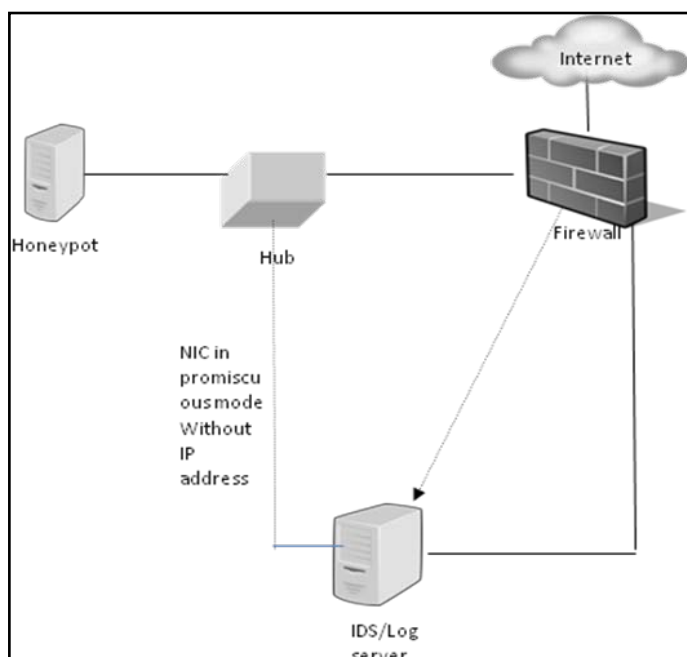


Fig. 5: Generation I Honeypot

The first role is to capture all network traffic traversing through the firewall, so that later analysis can be made. The second role is the standard IDS operation, which is to parse network traffic to detect malicious activities and alert the honeypot administrator accordingly.

B. Gen II Honeypot

GenII honeypots [11] released in 2002. After the concept of GenI honeypots was successful, the HoneyNet Project started work on the second generation, which improves various honeypot features. GenII honeypots target to provide a high level of interaction with a malicious user. This level of interaction rises the overall risk, so advanced methods of data control and capturing must be available. Figure 6 shows GenII honeypots architecture. Data control is a critical requirement for GenII honeypots. Once the honeypot is compromised, a malicious user may try to attack remote systems from the honeypot. The key component of Gen II honeypot is firewall. It has following capabilities:

Data capture, data control, data analysis, data alerting.

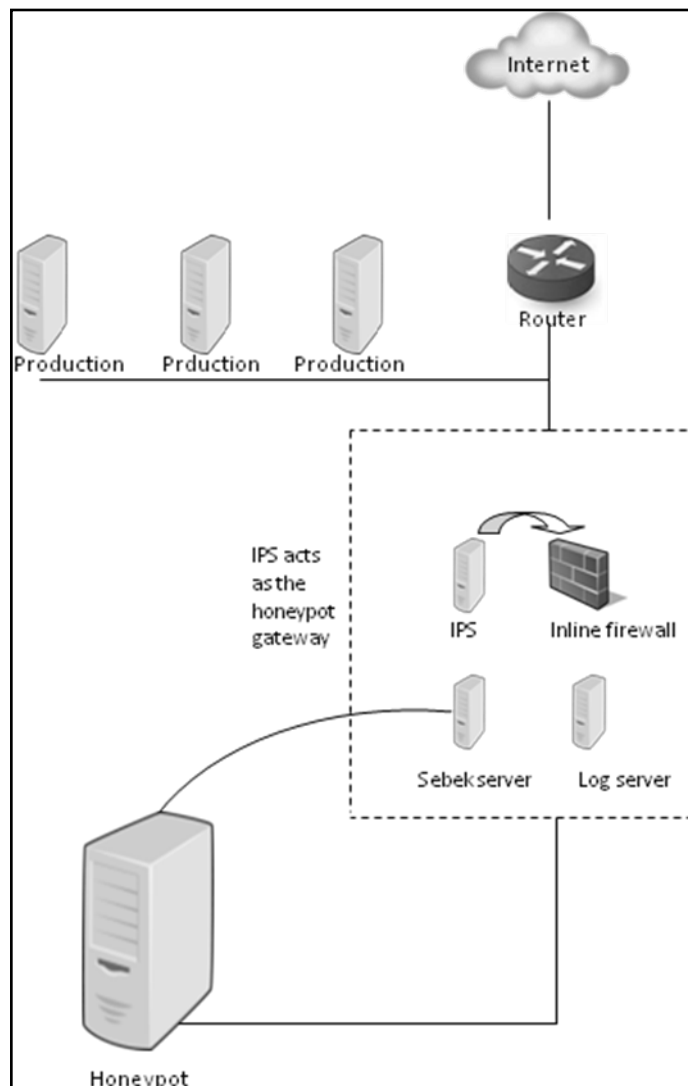


Fig. 6: Generation II Honeypot

C. Gen III honeypot

Gen III Honeypots were released at the end of year 2004. Gen II HoneyNets were released in order to address the deficiencies in Gen I Honeypots. Gen II and Gen III Honeypots have the same architecture, with the only difference being improvements in deployment and management mechanism in Gen III Honeypots along with the involvement of a Sebek server in the gateway – this is known as the Honeywall. This architecture incorporates 3 interfaces on the Honeywall. Two interfaces acted as a bridge between the internal network and the external Honeypot network; while the third interface is used for management and configuration tasks. Gen III honeypot has following improvements over Gen II honeypots:

1. Improved installation, operation and customization.
2. Improved data capture capability by introducing a new database schema.
3. Improved data analysis capability by introducing a new web based analysis tool.
4. Improved user interface and online documentation.

Further details of Gen III of honeypot can be found in [12].

V. Related Work and Some Popular Honeypots

Currently, there are many tools for Honeypots. We can divide them in three categories as data collection tools, data analysis tools, and visualization tools. For data collection infrastructure, there are two main deployments: the Brazilian Distributed Honeypots

project and the Matrix Chinese Distributed Honeynet deployment with CNCERT/CC.

As a log management server and log analyzer, there are some related works need to be mentioned:

A. Honeyd

Honeyd [3] is an open source computer program created by Niels Provos that allows a user to set up and run multiple virtual hosts on a computer network. These virtual hosts can be configured to mimic several different types of servers, allowing the user to simulate an infinite number of computer network configurations.

Advantages

- It can monitor any UDP or TCP port and entire networks.
- As an OpenSource solution, it is free and will develop easily and quickly with the input and development of others in the security association.
- It resists fingerprinting efforts by emulating operating systems at both the IP stack and the application level.

Disadvantages

- As it is a low-interaction solution, it cannot provide real operating system for hackers to interact with.
- It does not provide support for maintenance and troubleshooting of attacks.
- It has no built-in mechanism for alerting, and capturing extensive information.

B. Honeyview

Honeyview [13] is also a Honeyd log analyzer which can summarize the log file in text or graphic mode. Honeyd is an excellent tool to collect data from hackers and script-kiddies but it can be difficult to get an overview of what really happens, to overcome it honeyview is designed. Honeyview can save our time when watching and analyzing activities in the honeypot.

C. Honeydsum

Honeydsum [14] is basically a log analyzer written in Perl language designed by the Brazilian Honeynet team. It can produce summaries from Honeyd logs by specifying IP addresses, protocols, ports or networks. It shows the top source and port access and the no of connections per hour. It also supports input from various log files and can combine events from multiple honeypots. However, contrary to this work, their summaries are in text format only.

D. DTK (Deception Tool Kit)

The Deception ToolKit (DTK) [15] is a toolkit designed by Fred Cohen, is one of the original honeypots. It is designed to give defenders a couple of orders of magnitude advantage over attackers. It can be characterized as low to mid involvement honeypot. The basic idea is not new. We use deception to counter attacks. In the case of DTK, the deception is intended to make it appear to attackers as if the system running DTK has a large number of widely known vulnerabilities.

Advantages

- It provides more information than Specter.
- This toolkit is free and user has the source.
- Users can modify the scripts to emulate any vulnerability they want.

Disadvantages

- It takes so much time and work to install.
- Its scripts can potentially be exploited to give an attacker access to the system.

E. BOF (Back Officer Friendly)

Back Officer Friendly (BOF) [16] was originally created to detect when anyone attempts a Back Orifice scan against your computer. It was not originally intended to be a honeypot. Instead, it was a tool designed as a response to a specific threat. It was first developed in 1998 by Marcus Ranum and the folks at Network Flight Recorder in response to the Cult of the Dead Cows. It has since evolved to detect attempted connections to other services, such as Telnet, FTP, SMTP, POP3 and IMAP2. When BOF receives a connection to one of these services, it will fake replies to the hopeful hacker, wasting the attacker's time, and giving you time to stop them from other mischief.

Advantages

- It can run on almost any Windows based-platform, making it an excellent solution for desktop systems, including home users.
- It can monitor up to 7 emulated services.
- It is simple to use and deploy.
- One of the best advantages of BOF is its cost, it's free.

Disadvantages

- It can identify attacks only on seven ports.
- By interacting with BOF's predefined emulated services, it can be very easy for the attacker to determine BOF's identity.
- It has no remote logging and alerting functionality, so it is not suitable for enterprise level.

F. Specter

SPECTER [17] is a low-interaction honeypot designed to run on Windows. It simulates a complete machine. It can emulate 13 different operating systems, monitor up to 14 TCP ports and offers common Internet services such as SMTP, FTP, POP3, HTTP and TELNET. Its deployment, information gathering and alerting capabilities can be described in [1] chapter 7.

Advantages

- It can detect unauthorized activity in the networks immediately.
- It stores detailed logs of the actions taken by the attacker.
- It can collect important information about the identity of an attacker automatically.
- The system is very easy to set up and configure.
- It has outstanding notification capabilities and remote management.
- It can also be used in the field of deception and deterrence.

Disadvantages

- As a low-interaction honeypot, Specter offers no real operating system for the attacker to control and interact with.
- It is limited in the type of information it can gather, it can detect the attack and its source, but it cannot capture details of the attackers' activities.
- It can monitor only 14 ports.

G. Mantrap

ManTrap [1] is a high interaction honeypot designed, maintained, and sold by Recourse Technologies. It is unique in that it provides complete operating systems for hackers to interact with, capturing their every access. It is designed to be attacked and then to be compromised. ManTrap can create a virtual minefield that an internal attacker must successfully navigate in order to reach his target. One step in the wrong direction and the attacker is exposed. Like the homemade jailed environment ManTrap creates a highly controlled operating environment that an attacker can interact with. ManTrap creates cages that are mirror images of the master operating system. Further details can be found in [1] chap. 10.

Advantages

- It can be used to collect extensive amount of information.
- It provides flexible solution to the problem, and can cope up with any honeypot requirements.
- It can be used as a production honeypot to detect, prevent, and response to attacks.
- It can also be used as a research honeypot to detect new attacks.

Disadvantages

- The cages give intruders access to a wide range of capabilities. The more the hacker can do, the more we can learn. However, the more a hacker can do the more damage he can cause to the system.
- If once compromised, a cage can be easily used to harm other systems.

H. Hihat

The High Interaction Honeypot Analysis Toolkit (HIHAT)[18] allows to transform random PHP applications into web-based high-interaction Honeypots. In addition with a graphical user interface is provided which supports the process of monitoring the Honeypot and analysing the collected data.

Features of HIHAT

- It provides detailed information about all the data combined with every interaction to the honeypot.
- For known attacks HIHAT performs automatic scans.
- It can easily detect SQL-Injections, Cross-Site Scripting (XSS), Download attempts for malicious files e.g. with WGET or CURL, Command-Injections, etc.
- It generates numerous statistics about all the traffic identified at the system.
- It provides but is not limited to HTTP-GET, HTTP-POST and COOKIE data.
- It secures multiple copies of awful tools in a safe place for later detail analysis of it.

According to above discussion Table 2 summarizes all the benefits and limitations of the popular honeypots.

Table 2: Comparison of Various Honeypot Tools

	Man-Trap	BOF	DTK	Spec-ter	Honeyd	HIHAT
Interaction level	High	Low	Mid-dle	High	Low	High

Freely available	No	No	Yes	No	Yes	Yes
Log file support	Yes	No	Yes	Yes	Yes	Yes
OS simulation	Yes	No	Yes	Yes	Yes	Yes
Supported services	Unres- tricted	7	Limited	13	Unres- tricted	Unrestricted
Support Graphical user interface	Yes	Yes	No	Yes	Yes	Yes

VI. Conclusion

In this paper, we present the basic concept of honeypots and their use. We have discussed different classes of honeypot such as production, research honeypot, honeynet and distributed honeypot. On the basis of level of interaction we also classified low, medium and high level interactive honeypots. We examined the available honeypots, their advantages and disadvantages. One can choose suitable honeypot tool according to his need based on the given comparison. Honeypots are currently under great research and they provide us various advantages in the field of catching hackers. We examined that the resources like time, effort and money play very important role in the field. Although there are some disadvantages to using such devices, overall they are very useful and helpful tools for both to government officials and ordinary citizens. We suggested the suitability of different honeypots on various conditions and environment. They can be used to collect large information on attackers and threats, so we believe that they can be used in digital forensic investigation.

References

- [1] L Spitzner, "Honeypots : Tracking Hackers", Addison-Wesley Pearson Education, Boston, MA, 2002.
- [2] Lance Spitzner (2003), "Honeypots", Definitions and Value of Honeypots, [Online] Available: <http://www.tracking-hackers.com/papers/honeypots.html>.
- [3] Jun Wang, Jing Zeng, "Construction of large-scale honeynet Based on Honeyd", Procedia Engineering, Vol. 15, 2011, pp. 3260-3264.
- [4] Paul Baecher, Markus Koetter, Thorsten Holz, Maximilian Dornseif, Felix Freiling, "The nepenthes platform: an efficient approach to collect malware", In Proceedings of the 9th international conference on Recent Advances in Intrusion Detection, Springer 2006, pp. 165-184.
- [5] Iksu Kim, Yongyun Cho, "A security framework for blocking new types of internet worms in ubiquitous computing environments", Proceedings of the 12th international conference on Computational Science and Its Applications - Volume Part IV, (ICCSA'12), Springer 2012, pp. 622-631.
- [6] Honeynet Project (2006), "Know Your Enemy: Honeynets", What a honeynet is, its value, overview of how it works, and risk/issues involved, [Online] Available: <http://www.project.honeynet.org/papers/honeynet/>.
- [7] Lance Spitzner (2008), "Honeynet Project", [Online] Available: <https://www.projects.honeynet.org/sebek>.
- [8] Yun Yang, Hongli Yang, Jia Mi, "Design of Distributed Honeypot System Based on Intrusion Tracking", IEEE 2011, pp. 196-198.

- [9] Lance Spitzner (2003), Honeytokens: The Other Honeytrap", [Online] Available: <http://www.securityfocus.com/infocus/1713>, Security Focus.
- [10] Michele Adams, Iyatiti Mokube, "Honeytraps: Concepts, Approaches, and challenges", ACMSE March 2007, pp. 321-326.
- [11] Edward Balas, Camilo Viecco, "Towards a Third Generation Data Capture Architecture for Honeytraps", Proceedings of the IEEE Workshop on Information Assurance and Security June 2004, pp 1-9.
- [12] Jan Gerrit Göbel, "Advanced Honeytrap Based Intrusion Detection", Department of Computer Science, Diploma Thesis, July 2006.
- [13] Karl Hable (2003), "HoneyView a honeytrap Logfile Analyzer", [Online] Available: <http://www.honeyview.sourceforge.net/>.
- [14] Brazilian Honeytrap Project (2008), "honeydsum.pl". [Online] Available: <http://www.honeytrap.org.br/tools/>.
- [15] All.net, "Deception Toolkit", [Online] Available: <http://www.all.net/dtk/index.html>.
- [16] [Online] Available: <http://www.nfr.com/resource/backdoor.php>.
- [17] NETSEC (Network Security Software) (2012), [Online] Available: www.specter.com.
- [18] HIHAT (2007), [Online] Available: hihat.sourceforge.net/index.html.