RESEARCH ARTICLE OPEN ACCESS

Advanced Honey pot Architecture for Network Threats Quantification

Karthikeyan R[1], Dr.T.Geetha[2] , Shyamamol K.S[3] , Sivagami G[4]
[1,2] Asst.Prof, Dept of MCA, Gnanamani college of Technolgy, Namakkal, INDIA
[3,4]P.G.Scholar, Dept of MCA, Gnanamani college of Technolgy, Namakkal, INDIA.

Abstract:

Today internet security is a serious problem. For every consumer and business that is on the Internet, viruses, worms and crackers are a few security threats. There are the obvious tools that aid information security professionals against these problems such as anti-virus software, firewalls and intrusion detection systems, but these systems can only react to or prevent attacks-they cannot give us information about the attacker, the tools used or even the methods employed. Given all of these security questions honeypots are a novel approach to network security and security research alike. It is a resource, which is intended to be attacked and compromised to gain more information about the attacker and the used tools. It can also be deployed to attract and divert an attacker from their real targets. Honeypots is an additional layer of security. Honeypots have the big advantage that they do not generate false alerts as each observed traffic is suspicious, because no productive components are running on the system. The levels of interaction determines the amount of functionality a honeypots provides that is low and high interactions.

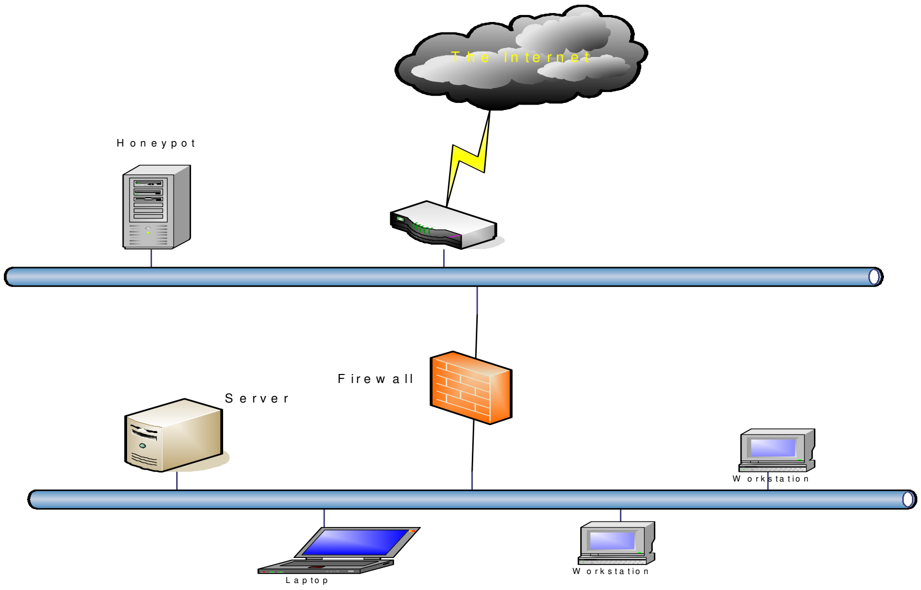
INTRODUCTION

Today's world increasingly relies on computer networks. The use of network resources is growing and network infrastructures are gaining in size and complexity refer by paper (3, 4, 6, 7, and 8). This increase is followed by a rising volume of security problems. New threats and vulnerabilities are found every day, and computers are far from being secure. In the first half of 2008, 3,534 vulnerabilities were disclosed by vendors, researchers and independents. Between 8 and 16% of these vulnerabilities were exploited the day they were released by malicious programs. The consequences affect users and companies at critical levels, from privacy issues to financial losses. To address this concern, network operators and security researchers have developed and deployed a variety of solutions refer by paper (9, 11). The goal of these solutions is two-fold: first to monitor, and second to protect network assets. Monitoring allows researchers to understand the different threats. Data are being collected to better characterize and quantify malicious activity. The goal of this dissertation is to introduce an innovative framework to better measure malicious threats in the organization network. The framework is based on a flexible hybrid honeypot architecture that we integrate with the organization network using network flows refer this paper (13, 15, 16). A honeypot is a deception trap, designed to entice an attacker into attempting to compromise the information systems in an organisation. If deployed correctly, a honeypot can serve as an early-warning and advanced security surveillance tool, minimising the risks from attacks on IT systems and networks. Honeypots can also analyse the ways in which attackers try to compromise an information system, providing valuable insight into potential system loopholes.

I.HONEYPOTS

According to Lance Spitzner, founder of the Honeynet Project, a honeypot is a system designed to learn how "black-hats" probe for and exploit weaknesses in an IT system1. 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, feigning 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. A honeypot works by fooling attackers into believing it is a legitimate system; they attack the system without knowing that they are being observed covertly. When an attacker attempts to compromise a honeypot, attack-related information, such as the IP address of the attacker, will be collected. This activity done by the attacker provides valuable information and analysis on attacking techniques, allowing system administrators to "trace back" to the source of attack if required. Honeypots can be used for production or research purposes. A production honeypot is used for risk mitigation. Most production honeypots are emulations of specific operating systems or services. They help to protect a network and systems against attacks generated by automated tools used to randomly look for and take over vulnerable systems. By running a production honeypot, the scanning process from these attack tools can beslowed right down, thereby wasting their time. Some production honeypots can even shut down attacks altogether by, for example, sending the attackers an acknowledgement packet with a window size of zero. This puts the attack into a “wait” status in which it could only send data when the window size increases3. In this way, production honeypots are often used as reconnaissance or deterrence tools.Research honeypots are real operating systems and services that attackers can interact with, and therefore involve higher risk. They collect extensive information and intelligence on new attack techniques and methods, and hence provide a more accurate picture of the types of attacks being perpetrated. They also provide improved attack prevention, detection and reaction information, drawn from the log files and other information captured in the process. In general, honeypot research institutions such as universities and military departments will run research honeypots to gather intelligence on new attack methods. Some of the research results are published for the benefit of the whole community.

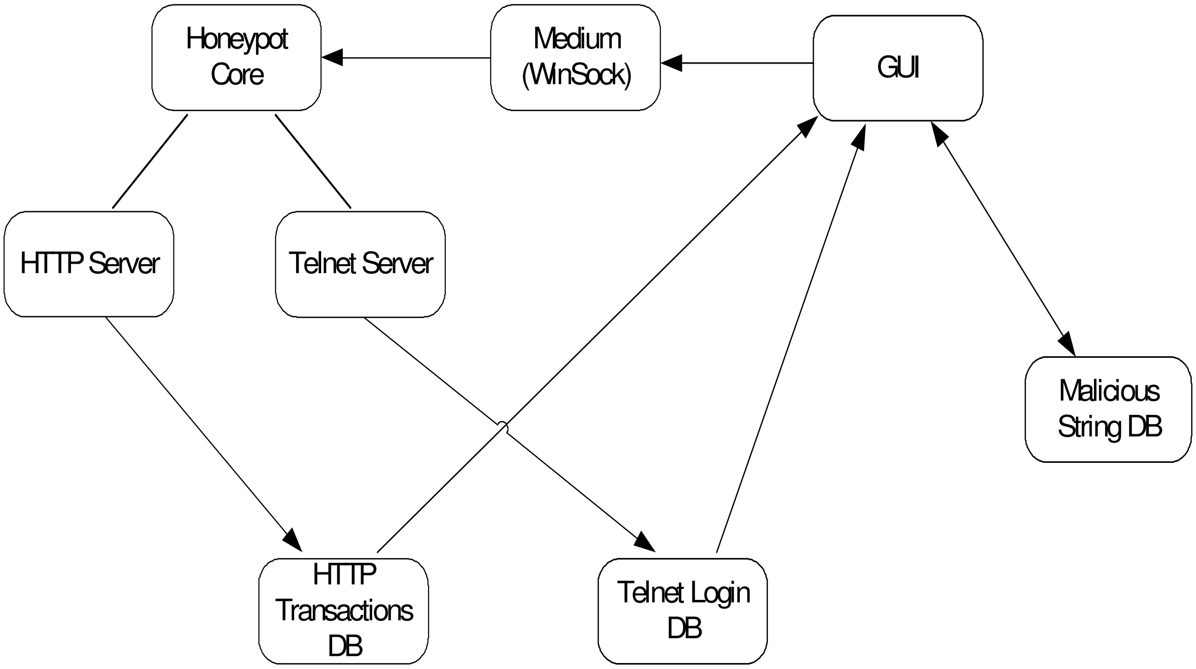
II.ARCHITECTURE OF HONEYPOTS



The program is divided into two main applications.

- GUI – Allows an easy way of starting and stopping the servers, searching through collected data and displaying statistics.
- Honeypot_Core – Creates and maintains the servers. Collects the data from the users and updates the databases.

A.BLOCK DIAGRAM



III. LEVELS OF HONEYPOTS

Honeypots can be classified into two general categories: low-interaction honeypots that are often used for production purposes, and high interaction honeypots that are used for research purposes.

A.LOW-INTERACTION HONEYPOTS

Low-interaction honeypots work by emulating certain services and operating systems and have limited interaction. The attacker's activities are limited to the level of emulation provided by the honeypot. For example, an emulated FTP service listening on a particular port may only emulate an FTP login, or it may further support a variety of additional FTP commands. The advantages of low-interaction honeypots are that they are simple and easy to deploy and maintain. In addition, the limited emulation available and/or allowed on low- interaction honeypots reduces the potential risks brought about using them in the field. However, with low-interaction honeypots, only limited information can be obtained, and it is possible that experienced attackers will easily recognise a honeypot when they come across one.

Example: Façades

A façade is a software emulation of a target service or application that provides a false image of a target host. When a façade is probed or attacked, it gathers information about the attacker. Some façades only provide partial application-level behaviour (e.g. banner presentation), while others will actually simulate the target service down to the network stack behaviour. The value of a façade is defined primarily by what systems and applications it can simulate, and how easy it is to deploy and administer. Façades offer simple, easy deployment as they often require minimal installation effort and equipment, and they can emulate a large variety of systems. Since they are not real systems, they do not have any real vulnerabilities themselves, and cannot be used as a jumping-off point by attackers. However, because they provide only basic information about a potential threat, they are typically used by small to medium-sized enterprises, or by large enterprises in conjunction with other security technology.

B.HIGH-INTERACTION HONEYPOTS

High-interaction honeypots are more complex, as they involve real operating systems and applications. For example, a real FTP server will be built if the aim is to collect information about attacks on a particular FTP server or service.By giving attackers real systems to interact with, no restrictions are imposed on attack behaviour, and this allows administrators to capture extensive details about the full extent of an attacker's methods. However, it is not impossible that attackers might take over a high-interaction honeypot system and use it as a stepping-stone to attack other systems within the organisation. Therefore, sufficient protection measures need to be implemented accordingly. In the worst case, the network connection to the honeypot may need to be disconnected to prevent attackers from further penetrating the network and machines beyond the honeypot system itself.

Example one: Sacrificial Lambs

A sacrificial lamb is a system intentionally left vulnerable to attack. The administrator will examine the honeypot periodically to determine if it has been compromised, and if so, what was done to it. Additional data, such as a detailed trace of commands sent to the honeypot, can be collected by a network sniffer deployed near the honeypot. However, the honeypots themselves are "live" and thus present a possible jumping-off point for an attacker. Additional deployment considerations must be made in order to isolate and control the honeypot, such as by means of firewalls or other network control devices, or by completely disconnecting the honeypot from the internal network. Because sacrificial lambs are themselves real systems, all results generated are exactly as they would be for a real system. However, sacrificial lambs require considerable administrative overhead, such as the installation of a full operating system, and manual application configuration or system hardening. The analysis is also conducted manually and may require additional tools. They also require additional deployment considerations as explained above, and will likely require a dedicated security expert to manage, support, and to analyse the resulting data from the honeypot system.

Example two: Instrumented Systems

An instrumented system honeypot is an off-the-shelf system with an installed operating system and kernel level modification to provide information, containment, or control. The operating system and kernel have been modified by professional security engineers, unlike the sacrificial lamb model. After modifying the operating system and kernel, they will leave the system running in the network as a real target. Instrumented systems combine the strengths of both sacrificial lambs and façades. Like the sacrificial lamb system, they provide a complete copy of a real system, ready for attackers to compromise, while at the same time (like façades) they are easily accessible and difficult to evade. Furthermore, the operating system and kernel in these systems have been modified to prevent attackers from using them as a stepping-stone for further attacks on other parts of the network.

Example three: Spam Honeypots

Honeypot technology is also used for studying spam and email harvesting activities. Honeypots have been deployed to study how spammers detect open mail relays. Machines run as simulated mail servers, proxies and web servers. Spam email is received and analysed to ascertain the reasons why they were received4. In addition, an email trap can be set up, using an email address dedicated to just receiving spam emails.

IV.HYBRID HONEYPOTS

The need to collect detailed attack processes on large IP spaces has pushed researchers to invent more scalable and intelligent architectures. Collapsar simplifies the deployment and administration of high interaction honeypots on large IP spaces by using GRE tunnels to route traffic from distributed networks into a centralized farm of honeypots. The limitation of Collapsar is to not provide any filtering mechanism that can prevent high interaction honeypots from being overloaded. Another project called Potemkin is based on the idea that idle high interaction honeypots do not even need to run. As a result, the architecture saves resources by starting a new virtual machine for each active IP address. As soon as an IP address becomes inactive, the virtual machine is destroyed to save physical memory and CPU resources. Such a system allows hundreds of virtual machines to run on a single physical host.

V.BENEFITS

Based on how honeypots conceptually work, they have several advantages.

- Reduce False Positives and False Negatives
- Data Value
- Resources
- Simplicity

VI.DRAWBACKS

- Limited View
- Specifically, Honeypots have the risk of being taken over by the bad guy and begin used to harm other system this risk various for different honeypots .

CONCLUSION

Honeypots have their advantages and disadvantages. They are clearly a useful tool for luring and trapping attackers, capturing information and generating alerts when someone is interacting with them. The activities of attackers provides valuable information for analysing their attacking techniques and methods. Because honeypots only capture and archive data and requests coming in to them, they do not add extra burden to existing network bandwidth. However, honeypots do have their drawbacks. Because they only track and capture activity that directly interacts with them, they cannot detect attacks against other systems in the network. Furthermore, deploying honeypots without enough planning and consideration may introduce more risks to an existing network, because honeypots are designed to be exploited, and there is always a risk of them being taken over by attackers, using them as a stepping-stone to gain entry to other systems within the network. This is perhaps the most controversial drawback of honeypots.

REFERENCE:

1. Srivathsa S Rao#1,Vinay Hegde#2 , Boruthalupula Maneesh#3, Jyothi Prasad N M#4, Suhas Suresh#5, August 2013, International Journal of Scientific and Research Publications, Volume 3, Issue 8
2. Abhishek Sharma,Nov-Dec 2013,International Journal of Technical Research and Applications e-ISSN: 2320-8163, www.ijtra.com Volume 1, Issue 5, PP. 07-12
3. R.Karthikeyan,” Improved Apriori Algorithm for Mining Rules” in the International Journal of Advanced Research in biology Engineering science and Technology Volume 11, Issue 4, April 2016, Page No:71-77.
4. R.Karthikeyan,Dr.T.Geetha “Honeypots for Network Security”, International journal for Research & Development in Technology.Volume 7.Issue 2 ,Jan 2017,Page No.:62-66 ISSN:2349-3585
5. https://www.client-honeynet.org/honeyc.html
6. R.Karthikeyan,”A Survey on Position Based Routing in Mobile Adhoc Networks” in the international journal of P2P Network Trends and Technology, Volume 3 Issue 7 2013, ISSN:2249-2615
7. R.Karthikeyan,”A Survey on Sensor Networks” in the International Journal for Research & Development in Technology Volume 7, Issue 1, Jan 2017, Page No:71-77
8. R.Karthikeyan,Dr.T.Geetha “Web Based Honeypots Network”,in the International journal for Research & Development in Technology.Volume 7.Issue 2 ,Jan 2017,Page No.:67-73 ISSN:2349-3585.
9. R.Karthikeyan,Dr.T.Geetha,“A Simple Transmit Diversity Technique for Wireless Communication”,in the International journal for Engineering and Techniques. Volume 3. Issue 1, Feb 2017, Page No.:56-61 ISSN:2395-1303.
10. C.Ganesh,B.Sathyabhama,Dr.T.Geetha “ Fast Frequent Pattern Mining using Vertical Data Format for Knowledge Discovery “International Journal of Engineering Research in Management & Technology. Vol.5,Issue-5,Pages:141-149.
11. R.Karthikeyan,Dr.T.Geetha “Strategy of Trible – E on Solving Trojan Defense in Cyber Crime Cases”, International journal for Research & Development in Technology.Volume 7.Issue 1 ,Jan 2017,Page No.:167-171
12. http://www.honeynet.org.pt/index.php/HoneyMole
13. R.Karthikeyan,”A Survey on Position Based Routing in Mobile Adhoc Networks” in the international journal of P2P Network Trends and Technology, Volume 3 Issue 7 2013, ISSN:2249-2615.
14. K.Ramya and K.Pavithradevi “Effective Wireless Communication”,International journal of Advanced Research, Vol 4(12), pp.1599-1562 dec 2016.
15. R.Karthikeyan,Dr.T.Geetha ”FLIP-OFDM for Optical Wireless Communications” in the international journal of Engineering and Techniques, Volume 3 Issue 1, Jan - Feb 2017, ISSN:2395-1303,PP No.:115-120.
16. R.Karthikeyan,Dr.T.Geetha”Application Optimization in Mobile Cloud Computing” in the international journal of Engineering and Techniques, Volume 3 Issue 1, Jan - Feb 2017, ISSN:2395-1303,PP No.:121-125.