HONEYPOTS – The new-way Security Analysis

By – D.R.Esesve
B.Tech (ECE), MPIT (Networking Technology)
dresesve@hotmail.com
http://www.geocities.com/dresesve

Symbiosis Center for Information Technology, Pune (www.scit.edu)

Abstract

Security – the most hackneyed word in the digital world. To build an effective security system we first need to know what the security flaws are. One such tool which helps in identifying the Black Hat community and their work is a HoneyPot. A honeypot is used in the area of computer and Internet security. 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. Main goal of this paper is to show the possibilities of HoneyPots and their use in a research as well as productive environment.

Introduction

Global communication is getting more important everyday. At the same time, computer crimes are increasing. Countermeasures are developed to detect and prevent attacks - most of these measures are based on known facts, known attack patterns. As in the military, it is important to know, who your enemy is, what kind of strategy he uses, what tools he utilizes and what he is aiming for. Gathering this kind of information is not easy but important. By knowing attack strategies, countermeasures can be improved and vulnerabilities can be fixed. To gather as much information as possible is one main goal of a honeypot [4].

Generally, such information gathering should be done silently, without alarming an attacker. All the gathered information leads to an advantage on the defending side and can therefore be used on productive systems to prevent attacks. A honeypot is primarily an instrument for information gathering and learning. Its primary purpose is not to be an ambush for the blackhat[1] community to catch them in action and to press charges against them. The focus

[1] Blackhat community in Network security parlance means intruders, crackers etc.

lies on a silent collection of as much information as possible about their attack patterns, used programs, purpose of attack and the blackhat community itself. All this information is used to learn more about the blackhat proceedings and motives, as well as their technical knowledge and abilities. This is just a primary purpose of a honeypot. There are a lot of other possibilities for a honeypot - divert hackers from productive systems or catch a hacker while conducting an attack are just two possible examples [1].

NOTE: Honeypots are not the perfect solution for solving or preventing computer crimes. Honeypots are hard to maintain and they need operators with good knowledge about operating systems and network security. In the right hands, a honeypot can be an effective tool for information gathering. In the wrong, inexperienced hands, a honeypot can become another infiltrated machine and an instrument for the blackhat community.

HoneyPots

"A honeypot is a resource whose value is being in attacked or compromised. This means, that a Honeypot is expected to get probed, attacked and potentially exploited. Honeypots do not fix anything. They provide us with additional, valuable information."

- Lance Spitzner[2]

Honeypots do not help directly in increasing a computer network's security. On the contrary, they do attract intruders and can therefore attract some interest from the Blackhat community on the network where the honeypot is located.

Elaboration:

A honeypot can not be used to fix anything. Moreover a honeypot can attract more interest in a specific network than one would like. So what can a honeypot provide, what can it be used for? There are two categories of HoneyPots – ***Production HoneyPots*** and ***Research HoneyPots***. A production honeypot is used to migrate the risk in an organization, while the second category, research, is meant to gather as much information as possible. These HoneyPots

[2] Lance Spitzner is the founder Honeynet Project and moderator for HoneyPot mailing list; He is the senior security architect at Sun Microsystems.

do not add any security value to an organization, but they can help to understand the blackhat community and their attacks as well as to build some better defenses against security threats.

A honeypot is a resource which is intended to get compromised. Every traffic from and to a honeypot is suspicious because no productive systems are located on this resource. In general, every traffic from and to a honeypot is unauthorized activity. All data collected by a honeypot is therefore interesting data.

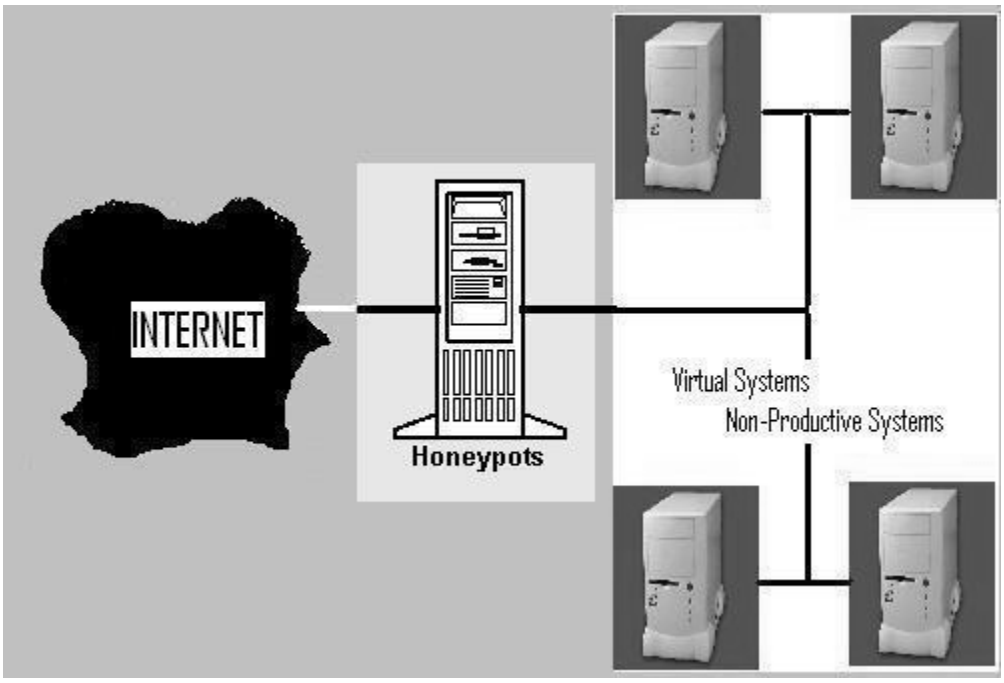

Figure 1

The advantage is a honeypot will in general not produce an awful lot of logs because no productive systems are running on that machine which makes analyzing this data much easier. Data collected by a honeypot is of high value and can lead to a better understanding and knowledge which in turn can help to increase overall network security. One can also argue that a honeypot can be used for prevention because it can deter attackers from attacking other systems by occupying them long enough and bind their resources. Against most attacks nowadays (which are based on automated scripts) a honeypot does not help deceiving individuals as there are no persons to deceive.

If a honeypot does not get attacked, it is worthless. Honeypots are normally located at a single point and the probability can be quite small that an attacker will find the honeypot. A

honeypot does also introduce a certain risk - blackhats could get attracted to the whole network or a honeypot may get silently compromised.

Classifications: [3]

One chief characteristic of a honeypot is its level of involvement. The level of involvement does measure the degree an attacker can interact with the operating system.

Low-Involvement Honeypot

A low-involvement honeypot typically only provides certain fake services. In a basic form, these services could be implemented by having a listener on a specific port. In such a way, all incoming traffic can easily be recognized and stored. With such a simple solution it is not possible to catch communication of complex protocols.

On a low-involvement honeypot there is no real operating system that an attacker can operate on. This will minimize the risk significantly because the complexity of an operating system is eliminated. On the other hand, this is also a disadvantage. It is not possible to watch an attacker interacting with the operating system, which could be really interesting. A low-involvement honeypot is like a one-way connection. We only listen, but we do not ask questions ourselves. The role of this approach is very passive.

A low involvement honeypot does reduce risk to a minimum through minimizing interaction with the attacker. A low-involvement honeypot can be compared to a passive IDS5 [1], as both do not alter any traffic or interact with the attacker or the traffic flow. They are used to generate logs and alerts if incoming packets match certain patterns. Examples of Low Level Involvement HoneyPots include Honeyd, Specter, and KFSensor [3].

High-Involvement Honeypot

A high-involvement honeypot has a real under laying operating system. This leads to a much higher risk as the complexity increases rapidly. At the same time, the possibilities to gather information, the possible attacks as well as the attractiveness increase a lot. One goal of a hacker is to gain root and to have access to a machine, which is connected to the Internet 24/7. A high involvement honeypot does offer such an environment.

As soon as a hacker has gained access, his real work, the interesting part, begins. Unfortunately the attacker has to compromise the system to get this level of freedom. He will then have root rights on the system and can do everything at any moment on the compromised

system. As per se, this system is no longer secure. Even the whole machine can not be considered as secure.

A high-involvement honeypot is very time consuming. The system should be constantly under surveillance. A honeypot which is not under control is not of much help and can even become a danger or security hole itself. It is very important to limit a honeypot's access to the local intranet, as the honeypot can be used by the blackhats as if it was a real compromised system. Limiting outbound traffic is also an important point to consider, as the danger once a system is fully compromised can be reduced.

By providing a full operating system to the attacker, he has the possibilities to upload and install new files. This is where a high-involvement honeypot can show its strength, as all actions can be recorded and analyzed. Gathering new information about the blackhat community is one main goal of a high-involvement honeypot and legitimates the higher risk.

The below table depicts the major differences between low level involvement and high level involvement HoneyPots.

Degree of involvement	**low**	**high**
Real Operating system	-	Used
Risk	Low	High
Information gathering	Connections	All
Compromise wished	-	Yes
Knowledge to run	Low	High
Knowledge to develop	Low	High
Maintenance time	Low	High

Table 1: [1]

Honeypot Location

A honeypot does not need a certain surrounding environment as it is a standard server with no special needs. A honeypot can be placed anywhere a server could be placed. But certainly, some places are better for certain approaches as others. A honeypot can be used on the Internet as well as the intranet, based on the needed service. Placing a honeypot on the intranet can be useful if the detection of some bad guys inside a private network is wished. It is especially important to set the internal thrust for a honeypot as low as possible as this system could be compromised, probably without immediate knowledge. If the main concern is the Internet, a honeypot can be placed at two locations:

- In front of the firewall
- DMZ[3]
- Behind the firewall (Intranet)

Each approach has its advantages as well as disadvantages.

Sometimes it is even impossible to choose freely as placing a server in front of a firewall is simply not possible or not wished. By placing the honeypot in front of a firewall, the risk for the internal network does not increase. The danger of having a compromised system behind the firewall is eliminated.

A honeypot will attract and generate a lot of unwished traffic like portscans or attack patterns. By placing a honeypot outside the firewall, such events do not get logged by the firewall and an internal IDS will not generate alerts. Otherwise, a lot of alerts would be generated on the firewall or IDS. Probably the biggest advantage is that the firewall or IDS, as well as any other resources, have not to be adjusted as the honeypot is outside the firewall and viewed as any other machine on the external network. Running a honeypot does therefore not increase the dangers for the internal network nor does it introduce new risks.

The disadvantage of placing a honeypot in front of the firewall is that internal attackers can not be located or trapped that easy, especially if the firewall limits outbound traffic and therefore limits the traffic to the honeypot. Placing a honeypot inside a DMZ seems a good solution as long as the other systems inside the DMZ can be secured against the honeypot.

Most DMZs are not fully accessible as only needed services are allowed to pass the firewall. In such a case, placing the honeypot in front of the firewall should be favored as opening all corresponding ports on the firewall is too time consuming and risky.

A honeypot behind a firewall can introduce new security risks to the internal network, especially if the internal network is not secured against the honeypot through additional firewalls. This could be a special problem if the IPs are used for authentication. It is important to distinguish between a setup where the firewall enables access to the honeypot or where access from the Internet is denied. A honeypot does often provide a lot of services. Probably most of them are not used as exported services to the Internet and are therefore not forwarded to the honeypot by the firewall. By placing the honeypot behind a firewall, it is inevitable to adjust the firewall rules if access from the Internet should be permitted. The biggest problem arises as soon

[3] DMZ means a demilitarized zone which is a network segment only partly accessible from the Internet.

as the internal honeypot is compromised by an external attacker. He gains the possibility to access the internal network through the honeypot. This traffic will be unstopped by the firewall as it is regarded as traffic to the honeypot only, which in turn is granted. Securing an internal honeypot is therefore mandatory, especially if it is a high-involvement honeypot. With an internal honeypot it is also possible to detect a wrongly configured firewall which forwards unwanted traffic from the Internet to the internal network. The main reason for placing a honeypot behind a firewall could be to detect internal attackers.

The best solution would be to run a honeypot in its own DMZ, therefore with a preliminary firewall. The firewall could be connected directly to the Internet or intranet, depending on the goal. This attempt enables tight control as well as a flexible environment with maximal security.

Active Information Gathering

Gathering information on a honeypot is mostly passive. Information is gathered out of the network stream or the bits and bytes on the machine itself. No information is retrieved by inquiring third parties for specific information about a certain identity. But information gathering does not only have to be passive. It is possible to get more information about a person, an IP address or an attack by querying specific services or machines. This can be very powerful as valuable information can be found. However this attempt is also dangerous as the attacker could take notice and vanish.

The following services are available:

- Who is
- fingerprinting network traffic
- port scan

Some of these methods can be detected by the attacker and are therefore a little bit more dangerous than others.

Honeynets

A honeypot is physically a single machine, probably running multiple virtual operating systems. Controlling outbound traffic is often not possible, as the traffic goes directly onto the network. In this case the only possibility to limit outbound traffic is to use a preliminary firewall.

Such a more complex environment is often referenced as Honeynet. A typical Honeynet consists of multiple HoneyPots and a firewall (or firewalled-bridge) to limit and log network traffic. An IDS is often used to watch for potential attacks and decode and store network traffic on the preliminary system.

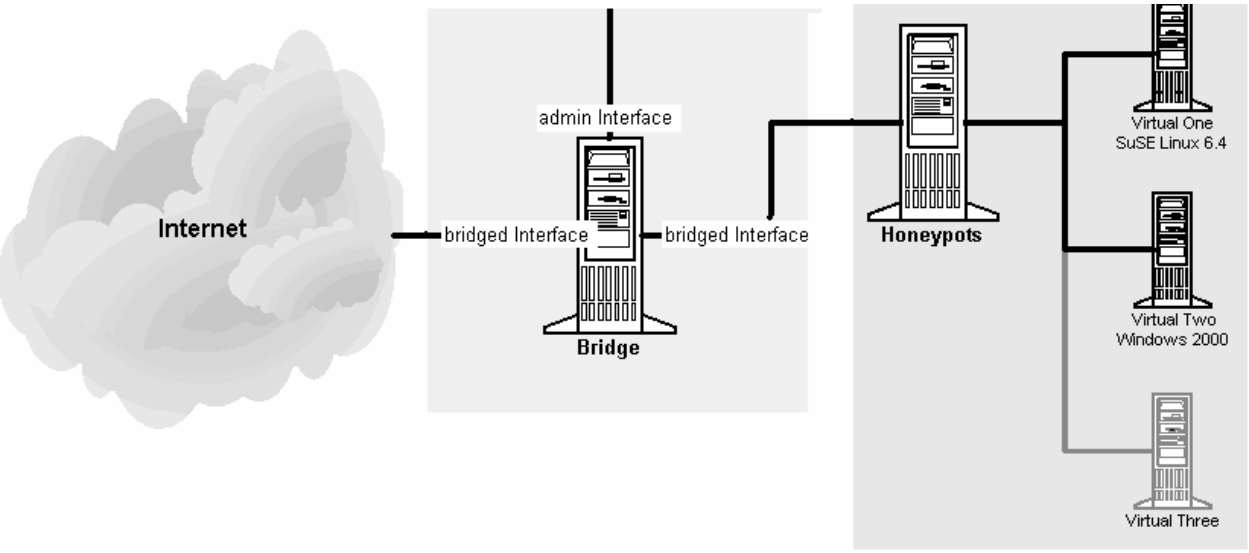

Figure 2

By placing a firewall in front of a honeypot (or multiple HoneyPots) the risk based on the honeypot can be reduced. It is possible to control the network flow, the inbound as well as the outbound connections. Logging of network traffic is much easier as this can be done on one centralized location for all HoneyPots. The captured data does not have to be placed on the honeypot itself and the risk of detecting this data by an attacker is eliminated.

By introducing new machines to the honeypot itself, more hardware is required. A solution with only one machine is thinkable. By using VMWare[4], setting up multiple virtual systems on one physical machine is possible. Through this attempt, it is even possible to place the firewall on the same machine as all virtual HoneyPots however the security of this solution isn't as good as having different physical machines. As soon as the Honeynet is a virtual environment, the system could be compromised and the attacker could be able to break out of the virtual machines. Placing a bridge with firewall capabilities in front of a honeypot is much safer as the attacker can not see the bridge. Even attacking the bridge is not possible as the bridge has no IP address and therefore no attack point exists.

[4] VMWare – a tool to run multiple virtual systems on a single physical system; further reference can be had from www.vmware.com

Introducing additional hardware also raises the complexity of the environment. Understanding networking and associated tools is important as long as the best security has to be provided.

How to make the HoneyPot Attractive?

Being the owner of a honeypot can be an interesting experience, but what if the members of the blackhat community do not find their way to the honeypot or, even more dramatically, are not interested in the honeypot at all? This section discusses some possibilities to attract the blackhat community.

Normally a honeypot is put into one network segment. But if an administrator has more than one network segment at hand (for example a class B net and a class C net, or a collection of non-connected class C networks) then he can ”widen” the ”surface” of his honeypot installation. A straight way would be to put one or more HoneyPots into each segment, with the penalty of increased administrative and monetary expenses. A more advanced technique would be the installation of IP-tunnels from different networks to a central honeypot installation. There are some considerable advantages over a bunch of single HoneyPots.

First, there is only one honeypot installation to administrate. Second, often no new machines must be installed in the monitored segments, since one can use existing computers to do the tunneling. Third, new networks can easily be integrated in the existent honeypot environment by just installing a new tunnel. Another approach to lure attackers is the offering of interesting services on the honeypot. Of course the question arises, what an interesting service is or what it should look like. For instance, [1]a so called ”script-kiddie” will never bother about an open database port on a machine, the advanced attacker will notice that there is obviously a database running and that the machine could hold some interesting data about the company the network belongs to.

Dangers and Threats

Running a Honeypot or Honeynet is not something that should be underestimated - there are some dangers one must be aware of which basically are:

- Unnoticed takeover of the honeypot by an attacker
- Lost control over the honeypot installation

- Damage done to third parties

Unnoticed takeover of a honeypot is surely a bad thing which has to be avoided in any case - otherwise the benefits of a honeypot could be rather questionable. If there is the possibility an attacker can infiltrate the system without being noticed by the operator then there is obviously a flaw in the design of the honeypot monitoring.

The loss of control over the honeypot and the attacker is also a serious issue. A honeypot should be designed in a way that the operator can on one hand safely disrupt any communications from the honeypot with its environment and on the other hand do backups of system states at any time for later investigation. The operator should never rely on any machine correlated with the honeypot - any administrative action must be applicable even if the honeypot is under total control by an attacker. In this context one has to point out the possible dangers of virtual machines. It is never guaranteed that the virtualization software is perfect and the attacker has no way to break out from the virtual machine into the host operating system. The host operating system of a virtualized honeypot is therefore not trustworthy and relying on its proper functioning should be avoided.

Another aspect of losing control is the deception of the operator by an attacker. If an attacker generates that much traffic and unfiltered log events that the operator is not able to keep track of all the on goings, the attacker has good chances that the real purpose of his attack is never discovered.

Damage done to third parties can have a very high price. Legal consequences (above all, paying compensation) are not desirable. Assuring by all means that third parties are not caused any harm should have high priority. Being very careful with all the possible aspects of damage done to others is important. This does not only consist of direct attacks, but also infringes of copyrights can be taken into this category. A honeypot which is used by an attacker as a MP3 14Distributed denial of service server can lead to ungracious acquaintance with certain worldwide conglomerates [1]. Generally it can be said that the operator of a honeypot has a heavy responsibility. He must be very attentive, which is not easily achieved, particularly if a honeypot is running for a long time.

Different Products available in the Market

The major advantage of open source software can be seen in this field. There are a lot of such products available to experiment with. A few of them and their features are discussed below.

I. Symantec Decoy Server [7]

Key Features:

1. Detects unauthorized access and system misuse to provide enterprises with cost-effective prioritization of threats
2. New! Includes the improved ability to automatically create simulated email traffic between users to enhance the decoy environment
3. New! Improved response mechanisms include frequency-based policies and the ability to shut down systems based on attacker activity
4. New! Improved reporting and logging eases report creation and enhances prioritization efforts and incident resolution
5. Provides early detection of threats, supplying information crucial to maintaining a secure network infrastructure
6. Enables stealth monitoring and containment, plus live attack analysis
7. Detects both host- and network-based intrusions while eliminating the inefficiencies and time penalties of false positives
8. Offers centralized management, policy-based response, and comprehensive reporting and trend analysis for enterprise environments

II. LaBrea Tarpit [2]

1. This is a freeware honeypot created by Tom Liston that will run on any flavor of OpenBSD, Linux, Solaris or Windows.
2. Sticky HoneyPot
3. It borrows unassigned IP addresses on the network where it is and answers to connection requests

III. HoneyD [2]

1. Versatile HoneyPot that pretends to be other operating systems at the TCP/IP stack.
2. Simulates many virtual hosts.
3. Simulates various operating systems and different routing topologies.

Legal Issues

Implementation of HoneyPots also involves a lot of legal issues. Unfortunately, in addition to becoming popular, HoneyPots have also suffered a lot of criticism recently. The areas that may apply when implementing a honeypot could include [2]:

- Possible violation of the "standard of due care". A honeypot may be considered a bad neighbor type of behavior.
- Peripheral attack – where your compromised HoneyPot is used to attack others (i.e. denial of service or virus infection).
- HoneyPots can also be considered a means of entrapment

Conclusion

A honeypot is an illusion that is weaved for the attacker. The illusion can be as creative as we want it to be. A good illusion will get us zero day exploits, root kits, and loads of information on how attackers work. The key point here is only a best thief can become a best cop, just because, he knows how thefts are done and thus could recover. Same way it is very important to know how the patterns of attacks used by the blackhat community. This helps us design fool proof security systems.

HoneyPots are rapidly gaining a place in defense strategies, while they maintain an important status in the security research community. Before implementation, consideration of the goals for the honeypot must be thoroughly examined, and a cost/benefit analysis must be completed. Extreme care must also be taken when implementing a HoneyPot and it should be treated like any other security device or system implemented on your network, with constant care and feeding to ensure that its standards are kept up to date.

References

1. White Paper: Honeypots - Reto Baumann, Christian Plattner, February 26, 2002
2. How to build a HoneyPot? – Kristy Westphal – Sysadmin (Magazine)– September 2003
3. HoneyPots - Presentation by Lance Spitzner
4. Cryptography & Network Security – By William Stallings

5. Fighting Internet Worms With Honeypots - by Laurent Oudot - http://www.securityfocus.com/infocus/1740
6. http://project.honey.org
7. Symantec Decoy Server - http://enterprisesecurity.symantec.com/products/products.cfm?ProductID=157
8. www.honeypots.org
9. http://www.tracking-hackers.com/misc/faq.html