

How to build and use a Honeypot

By

Ralph Edward Sutton, Jr

**DTEC 6873
Section 01**

Abstract

Everybody has gotten hacked one way or another when dealing with computers. When I ran across the idea of a honeypot and what exactly it was I became intrigued with the idea of actually getting back at these mysterious hackers. I want to build a honeypot, put it on my home network, and see what I can attract. I will build a honeypot and put it out for business. I researched what the ideal computer set up would be and built one. I will discuss the computer, how to build one, and what my results were.

Introduction

In the technology driven world we live in, the value guarding of information is crucial. The ability to guard this information has become of the highest importance and an art form. With that said, as a network administrator you have to be prepared to protect your network and the information on your network with extreme and sometimes diverse measures. One of these measures is a honeypot. With a honeypot, hackers are actually allowed in to your network to a certain degree and then the ability to block them out becomes a reality by checking your logs to see who and how they are doing it.

What is a Honeypot

Honeypots are a highly flexible security tool with different applications for security. They don't fix a single problem. Instead they have multiple uses, such as prevention, detection, or information gathering. Honeypots all share the same concept, a security resource that should not have any production or authorized activity. In other words, deployment of honeypots in a network should not affect critical network services

and applications. A honeypot is a security resource whose value lies in being probed, attacked, or compromised. There are two general types of honeypots: Production honeypots are easy to use, capture only limited information, and are used primarily by companies or corporations; and Research honeypots are complex to deploy and maintain, capture extensive information, and are used primarily by research, military, or government organizations. (<http://www.honeypots.net/>) Honeypots are increasingly used to provide early warning of potential intruders, identify flaws in security strategies, and improve an organization's overall security awareness. "Honeypots can simulate a variety of internal and external devices, including Web servers, mail servers, database servers, application servers, and even firewalls. As a software development manager, I regularly use honeypots to gain insight into vulnerabilities in both the software my team writes and the OS upon which we depend."

(<http://www.windowsitpro.com/Windows/Article/ArticleID/44711/44711.html>)

A honeypot is a security resource whose value lies in being probed, attacked, or compromised. This means that whatever we designate as a honeypot, it is our expectation and goal to have the system probed, attacked, and potentially exploited. A honeypot also is a detection and response tool, rather than prevention which it has a little value in. (<http://www.securitydocs.com/library/2692>) A better way to think of a honeypot is as an Internet-attached server that acts as a decoy, luring in potential hackers in order to study their activities and monitor how they are able to break into a system. Honeypots are designed to mimic systems that an intruder would like to break into but limit the intruder from having access to an entire network. If a honeypot is successful, the intruder will have no idea that s/he is being tricked and monitored. Most honeypots are installed

inside firewalls so that they can better be controlled, though it is possible to install them outside of firewalls. A firewall in a honeypot works in the opposite way that a normal firewall works: instead of restricting what comes into a system from the Internet, the honeypot firewall allows all traffic to come in from the Internet and restricts what the system sends back out. By luring a hacker into a system, a honeypot serves several purposes: The administrator can watch the hacker exploit the vulnerabilities of the system, thereby learning where the system has weaknesses that need to be redesigned. The hacker can be caught and stopped while trying to obtain root access to the system. By studying the activities of hackers, designers can better create more secure systems that are potentially invulnerable to future hackers.

<http://www.securitydocs.com/library/2692>

Building the Honeypot

Building a honeypot is not difficult at all. I have a network set up at my house and it has been successfully intruded a few times. I am currently finishing a Masters degree in Network Management and Information Security so I have the knowledge to combat this problem. But, you do not need a Masters degree to build a honeypot. Setting up and operating a honeypot involves legal considerations as well as some expertise with networking tools and computer forensics.

I chose Windows 2003 Professional system with a 1GHz processor and a CD-ROM drive. Windows 2003 Professional was the best choice to since it can be secured the most from the operating systems I had available to chose from, Windows XP,

Windows 2000 Server and Windows 2003 Professional. I beefed up the computer to 512 Mb of RAM, from 256 RAM and it had a 10/100 network card already. If this study was for a company I would suggest a DVD/RW drive so the company could archive the findings for evidence if needed and also to see if a pattern would develop over time to the probability of an attack and what type of attack. I then installed a program called Snort. This program is an open source network intrusion prevention and detection system utilizing a rule-driven language, which combines the benefits of signature, protocol and anomaly based inspection methods. Snort is the most widely deployed intrusion detection and prevention technology worldwide and has become the de facto standard for the industry. (<http://www.snort.org/>) Snort is a free program which is extremely powerful in what it does. This is part of an intrusion detection system. I also found a good Windows based Honeybot. Honeybot works by opening over 1000 udp and tcp listening sockets on your computer and these sockets are designed to mimic vulnerable services. When an attacker connects to these services they are fooled into thinking they are attacking a real server. The honeypot safely captures all communications with the attacker and logs these results for future analysis. Should an attacker attempt an exploit or upload a rootkit or trojan to the server the honeypot environment will safely store these files on your computer for analysis and submission to antivirus vendors. (<http://www.atomicsoftwaresolutions.com/honeybot.php>) It can be found at the address listed. Honeybot is very easy to use and meshes with Microsoft Windows very well. As you can see on the screen shot at the end of my paper it scans your ports for any unusual activities.

Types of Honeypots

There are two main categories of honeypots – production honeypots and research honeypots, but what matters most is the kind of involvement and interaction of these honeypots with the attackers. It actually depends on what the organization wants to achieve when they choose the level of interaction for a honeypot. Honeypots can be classified into three different levels; Low-Interaction Honeypots, Medium-Interaction Honeypots and High-Interaction Honeypots. In terms of installation, configuration, deployment and maintenance, the low-interaction honeypots are the easiest to implement. Basic services such as Telnet and FTP are emulated on low interaction honeypots. It limits the hacker to interact with only these few pre-configured services. For example, a honeypot could emulate a Windows 2000 server running with several services such as Telnet and FTP. A hacker could first telnet to the honeypot to get a banner which would indicate what operating system the honeypot is running on. The hacker will then be brought to a login screen. The hacker have two options to gain access to the honeypot; either by brute force or password guessing. The honeypot would capture and collect all the hacker's attempts to the honeypot. But remember, there is no real or legitimate operating system for the hacker to interact with, but instead the hacker's involvements and interactions are limited to only login attempts. The main objective of low-interaction honeypot is only to detect, such as unauthorized probes or login attempts.

<http://www.securitydocs.com/library/2692>)

Low-interaction honeypots can be easily installed on the system and configured to any of the services specified above. This low-interaction honeypot is both easy to

deploy and maintain. But to prevent the system from being fully exploited by hackers, the administrator needs to ensure patch management on the host system and to conscientiously monitor the alert mechanisms. Low-interaction honeypots have the lowest level of risk. The honeypot cannot be used as a launch pad to attack other systems as there is no legitimate operating system for the hacker to interact with. The low-interaction honeypot is only good at capturing known attack patterns, but is worthless at interacting or discovering unknown attack signatures.

<http://www.securitydocs.com/library/2692>)

Another type of honeypot is the Medium-Interaction Honeypots. In terms of interaction, this is a little more advanced than low-interaction honeypots, but a little less advanced than high-interaction honeypots. Medium-Interaction honeypots still do not have a real operating system, but the bogus services provided are more sophisticated technically. (<http://www.securitydocs.com/library/2692>)

The final and most advanced of honeypots are the high-interaction honeypots. These kinds of honeypots are really time-consuming to design, manage and maintain.

Among the three types of honeypots, this honeypot possess a huge risk. But, the information and evidence gathered for analysis are bountiful. The goal of a high interaction honeypot is to give the attacker access to a real operating system where nothing is emulated or restricted. (<http://www.securitydocs.com/library/2692>)

Do to time constraints, I chose to build a low interaction Honeypot. It would give me time to set it up correctly and learn to operate it with some success.

Honeypot... Ready, Set, Go!

Placement of the honeypot is crucial. An organization may place a honeypot inside their internal network, secured by their perimeter defenses where it should never to be attacked. Any traffic captured on the honeypot in this situation would indicate that another computer inside the network is already infected with a virus or worm, or even that a company employee is attempting to break into the computer.

Another method is to attach the honeypot directly to the internet which normally results in captured malicious network traffic in minutes. A direct connection is the most basic setup for honeypot users and in this scenario the honeypot computer is placed external to your production systems and allocated a public IP address.

The most popular choice of honeypot placement for internet users is to place the honeypot in your network DMZ where all unsolicited internet probes are forwarded to your honeypot computer. (<http://www.atomicsoftwaresolutions.com/honeybot.php>) I placed mine on the outside of my router and started it. I felt this placement would get the most attention and action. Only five minutes in to the active placement and I had some serious action already, as the screen shots indicate. It is quite amazing how fast the Honeybot picked up on UDP traffic trying to scan ports on my computer. I was really surprised that my computer, at a residence, not a business, was getting the action of my ports being scanned.

I suggest that you install Honeybot on a dedicated computer with no valuable information or resources required of it. In fact, you want your honeypot to be as free as possible from any legitimate traffic so in broad terms it can consider any traffic to the honeypot to be malicious in nature. Honeybot requires minimum operating system of Windows 2000 and at least 128MB RAM is recommended.

<http://www.atomicsoftwaresolutions.com/honeybot.php>

Results

The results were impressive. Not the actual fact of what the honeypot caught/stopped, but in the fact that how fast my computer was assaulted as fast as it was. It was amazing how fast the computer was attacked. It seemed like it was bait for sharks in a feeding frenzy. I let the honeypot run for six hours and came back to see what I would find. Port 162 got quite a bit of scanning while port 67 and 68 were occasionally hit as well. Port 162 is commonly known as snmptrap. My higher ports were also being scanned to find an open one. It seemed every 3 to 5 seconds my ports were being scanned to see if they were open or closed. This was my first time using a honeypot on my system. After seeing the results, I feel you must have one for your business if you plan to survive in the computer world. Even if you have a network at your home, I believe you should have one. It is not hard to set up and the benefits you gain from having a honeypot are amazing. The cost is minimal. If you have an extra computer at your home office this could easily be turned into a higher end honeypot. If you do not have an extra computer you could certainly use your own computer and turn it in to a

lower active honeypot to learn what it does and become proficient at the art of intrusion detection. The biggest problem you will run into will be to learn the software and what it means when an alert comes up or you check your logs to see the activity. The only real expense is your time in putting one together.

Conclusion

In this growing IT arena, there is also a need to strengthen its security. Preventive, Detective and Responsive measures have to be undertaken in order to improve IT Security. To improve our Security and to fight the enemy, we must know them, befriend them and learn from them. Hackers can find our computers in just 30mins. Malicious attackers are constantly scanning our network looking for vulnerable loopholes and open ports. But without the knowledge of the enemy, how can we defend ourselves? We have to think like a hacker in order to stop a hacker. Honeypots can be used simply to confuse and deflect attacks or to collect evidence. There are many free Windows based and Linux based honeypot programs available to individuals and companies. There is no reason you should not have one to use. The low interactive honeypot can be run on your own system to learn how they work and to see just what really is going on with your network. (<http://www.securitydocs.com/library/2692>) Either way, they're a cost-effective tool you should add to your security arsenal.

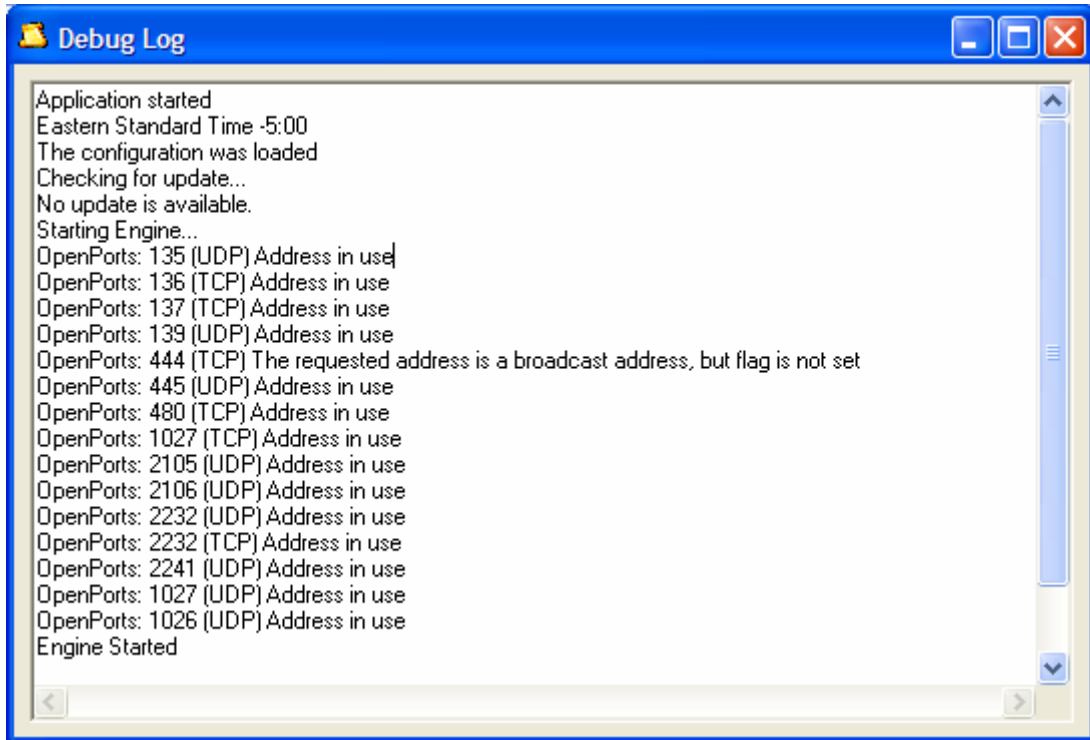
No Author Listed. (2005). Honeypots and Honeynets. Retrieved November 5, 2005, from Intrusion Detection, Honeypots and Incident Handling Resources Website: <http://www.honeypots.net/>

Kinsella, J. (January 2005). Build a PC Honeypot. Retrieved November 5, 2005, from Windows IT Pros Website: <http://www.windowsitpro.com/Windows/Article/ArticleID/44711/44711.html>

Noordin, M. (November 5, 2004). Honeypots Revealed. Retrieved November 5, 2005, from Security Docs.com Website: <http://www.securitydocs.com/library/2692>

No Author Listed. (No Date Listed). What is Snort? Retrieved on November 5, 2005, from snort.org Website: <http://www.snort.org/>

No Author Listed. (2005). Honeybot Retrieved on November 5, 2005, from atomicsoftwareresolutions.com Website: <http://www.atomicsoftwaresolutions.com/honeybot.php>



HoneyBOT - Log_20051106.bin
 File View Help

Date	Time	Remote IP	Remote Port	Local IP	Local Port	Protocol
11/6/2005	11:20:30 AM	192.168.1.1	18293	192.168.1.100	162	UDP
11/6/2005	11:20:30 AM	192.168.1.1	18294	192.168.1.100	162	UDP
11/6/2005	11:21:27 AM	192.168.1.1	18295	192.168.1.100	162	UDP
11/6/2005	11:21:28 AM	192.168.1.1	18296	192.168.1.100	162	UDP
11/6/2005	11:22:06 AM	192.168.1.1	18297	192.168.1.100	162	UDP
11/6/2005	11:22:06 AM	192.168.1.1	18298	192.168.1.100	162	UDP
11/6/2005	11:22:19 AM	192.168.1.1	18299	192.168.1.100	162	UDP
11/6/2005	11:23:24 AM	192.168.1.1	18300	192.168.1.100	162	UDP
11/6/2005	11:31:30 AM	192.168.1.1	67	192.168.1.100	68	UDP
11/6/2005	11:31:33 AM	192.168.1.1	67	192.168.1.100	68	UDP
11/6/2005	11:31:44 AM	192.168.1.1	18301	192.168.1.100	162	UDP
11/6/2005	11:32:05 AM	192.168.1.1	18302	192.168.1.100	162	UDP
11/6/2005	11:32:05 AM	192.168.1.1	18303	192.168.1.100	162	UDP
11/6/2005	11:32:40 AM	192.168.1.1	18304	192.168.1.100	162	UDP
11/6/2005	11:34:45 AM	192.168.1.1	18305	192.168.1.100	162	UDP
11/6/2005	11:38:07 AM	192.168.1.1	18306	192.168.1.100	162	UDP
11/6/2005	11:38:35 AM	192.168.1.1	18307	192.168.1.100	162	UDP
11/6/2005	11:41:08 AM	192.168.1.1	18308	192.168.1.100	162	UDP
11/6/2005	11:41:09 AM	192.168.1.1	18309	192.168.1.100	162	UDP
11/6/2005	11:41:12 AM	192.168.1.1	18310	192.168.1.100	162	UDP
11/6/2005	11:41:12 AM	192.168.1.1	18311	192.168.1.100	162	UDP
11/6/2005	11:41:19 AM	192.168.1.1	18312	192.168.1.100	162	UDP
11/6/2005	11:41:26 AM	192.168.1.1	18313	192.168.1.100	162	UDP
11/6/2005	11:41:26 AM	192.168.1.1	18314	192.168.1.100	162	UDP
11/6/2005	11:41:36 AM	192.168.1.1	18315	192.168.1.100	162	UDP
11/6/2005	11:42:47 AM	192.168.1.1	18316	192.168.1.100	162	UDP
11/6/2005	11:46:04 AM	192.168.1.1	18317	192.168.1.100	162	UDP
11/6/2005	11:46:04 AM	192.168.1.1	18318	192.168.1.100	162	UDP
11/6/2005	11:46:11 AM	192.168.1.1	18319	192.168.1.100	162	UDP
11/6/2005	11:46:59 AM	192.168.1.1	18320	192.168.1.100	162	UDP
11/6/2005	11:47:38 AM	192.168.1.1	18321	192.168.1.100	162	UDP
11/6/2005	11:47:38 AM	192.168.1.1	18322	192.168.1.100	162	UDP

34 records 1229 sockets

HoneyBOT - Log_20051106.bin

File View Help

Ports

- 162
- 67
- 68

Remotes

- 192.168.1.1
- 192.168.1.100

Date	Time	Remote IP	Remote Port	Local IP	Local Port	Protocol
11/6/2005	4:23:52 PM	192.168.1.1	18988	192.168.1.100	162	UDP
11/6/2005	4:23:52 PM	192.168.1.1	18989	192.168.1.100	162	UDP
11/6/2005	4:23:54 PM	192.168.1.1	18990	192.168.1.100	162	UDP
11/6/2005	4:23:57 PM	192.168.1.1	18991	192.168.1.100	162	UDP
11/6/2005	4:24:04 PM	192.168.1.1	18992	192.168.1.100	162	UDP
11/6/2005	4:24:05 PM	192.168.1.1	18993	192.168.1.100	162	UDP
11/6/2005	4:24:05 PM	192.168.1.1	18994	192.168.1.100	162	UDP
11/6/2005	4:24:05 PM	192.168.1.1	18995	192.168.1.100	162	UDP
11/6/2005	4:24:05 PM	192.168.1.1	18996	192.168.1.100	162	UDP
11/6/2005	4:24:06 PM	192.168.1.1	18997	192.168.1.100	162	UDP
11/6/2005	4:24:06 PM	192.168.1.1	18998	192.168.1.100	162	UDP
11/6/2005	4:24:40 PM	192.168.1.1	18999	192.168.1.100	162	UDP
11/6/2005	4:24:40 PM	192.168.1.1	19000	192.168.1.100	162	UDP
11/6/2005	4:25:07 PM	192.168.1.1	19001	192.168.1.100	162	UDP
11/6/2005	4:26:01 PM	192.168.1.1	19002	192.168.1.100	162	UDP
11/6/2005	4:26:01 PM	192.168.1.1	19003	192.168.1.100	162	UDP
11/6/2005	4:26:04 PM	192.168.1.1	19004	192.168.1.100	162	UDP
11/6/2005	4:29:08 PM	192.168.1.1	19005	192.168.1.100	162	UDP
11/6/2005	4:29:08 PM	192.168.1.1	19006	192.168.1.100	162	UDP
11/6/2005	4:29:42 PM	192.168.1.1	19007	192.168.1.100	162	UDP
11/6/2005	4:29:43 PM	192.168.1.1	19008	192.168.1.100	162	UDP
11/6/2005	4:29:44 PM	192.168.1.1	19009	192.168.1.100	162	UDP
11/6/2005	4:29:44 PM	192.168.1.1	19010	192.168.1.100	162	UDP
11/6/2005	4:29:44 PM	192.168.1.1	19011	192.168.1.100	162	UDP
11/6/2005	4:29:44 PM	192.168.1.1	19012	192.168.1.100	162	UDP
11/6/2005	4:29:44 PM	192.168.1.1	19013	192.168.1.100	162	UDP
11/6/2005	4:29:45 PM	192.168.1.1	19014	192.168.1.100	162	UDP
11/6/2005	4:29:45 PM	192.168.1.1	19015	192.168.1.100	162	UDP
11/6/2005	4:29:45 PM	192.168.1.1	19016	192.168.1.100	162	UDP
11/6/2005	4:29:45 PM	192.168.1.1	19017	192.168.1.100	162	UDP
11/6/2005	4:29:47 PM	192.168.1.1	19018	192.168.1.100	162	UDP
11/6/2005	4:29:48 PM	192.168.1.1	19019	192.168.1.100	162	UDP

731 records | 1229 sockets