Global Information Assurance Certification Paper

David Klug
Custom-Edge Corp
13831 Chalco Valley Pkwy
Omaha, NE 68154
402-758-7715

Honey pots and Intrusion Detection

Overview

This Paper is written on the subject of honey pots. It will cover many aspects of a honey pot including, what are they, how they work, how to build a honey pot, several types of commercial honey pots, are they worth it, and some legal issues that are involved in using them.

What are Honey pots?

Honey pots are one of the newest methods used in intrusion detection. The idea behind a honey pot is to setup a "decoy" system that has a non-hardened operating system or one that appears to have several vulnerabilities for easy access to its resources. The decoy system should be set up in a similar manner to those of the production servers in the corporation and should be loaded with numerous fake files, directories, and other information that may look real. By making the honey pot appear to be a legitimate machine with legitimate files, it leads the hacker to believe that they have gained access to important information. With a little luck the intruder will stay around in an attempt to collect data while the honey pot collects information about the intruder and the source of his or her attack. Ideally honey pots provide an environment where intruders can be trapped or vulnerabilities accessed before an attack is made on real assets. Decoys are setup not to capture the bad guy but to monitor and learn from their moves, find how they probe and exploit the system and how those exploitations can be prevented in production systems and doing this all without detection from the hacker.

How do honey pots work?

Honey pots work on the idea that all traffic to a honey pot should be deemed suspicious. As stated before honey pots are generally based on a real server, real operating system, and with data that appears to be real. One of the main differences is the location of the machine in relation to the actual servers. The decoy machine are usually placed somewhere in the DMZ. This ensures that the internal network is not exposed to the hacker. Honey pots work by monitoring and\or sometimes controlling the intruder during their use of the honey pot. This can done whether the attack came from the outside or the inside of the network, depending on the location of the decoy system. Honey pots are generally designed to audit the activity of an intruder, save log files, and record such events as the processes started, compiles, file adds, deletes, changes, and even key strokes. By collecting such data the honey pots work to improve a

corporation's overall security system. If enough data is collected it may be used to prosecute in serious situations. In cases where you do not wish to prosecute the data collected can be used to measure the skill level of hackers, their intent, and in some cases, even their identity. All in all the honey pot helps a company prepare for attacks and respond to those attacks by learning from information gathered.

Integrating honey pots

The integration of your honey pot into your network is a great determining factor into how effective it will be. You should position the decoy system close to your production servers to tempt intruders that are targeting production servers. One such possibility is to emulate nonproduction services on production servers. By using port redirection on an upstream router or firewall, it will appear that honey pot services are running on production systems. This would require an upstream router or firewall capable of performing port/service redirection; in this case the upstream device is responsible for transparently handling the address translation of the honey pot in order to help conceal its real destination IP address. One example of this is if you run a production web server (Port 80), telnet (port 23) and SMTP (Port 25) could then be redirected to a honey pot. Because these services should not be accessed on a production system, the honey pot should send off an immediate alert or at the very least, log the incident. In the scenario listed above, you can detect probing and tampering on production systems but only on nonproduction services so you would not be alerted to tampering on the production server because the service is not redirected to the honey pot. It is also important to realize the limitations of service emulation. Intrusion detection systems must know about the vulnerability prior the exploitation in order for it to emulate properly.

Another way to deploy a honey pot is to place it between production servers. If production servers are addressed as .9, .10, .11, and .13 it is ideal to address the honey pot as .12. The idea behind this is to catch intruders that "sweep scan" entire network ranges looking for vulnerable services. This method is not always the most affective as the intruder might just as easily focus on a production system thus making the decoy machine useless.

Building a Honey pot

When building a honey pot you should begin by loading an operation system whether it is NT, Linux, Solaris, etc as you would any other system. Do not do anything special to the system, as you want it to be easily compromised so you can collect information about the intruder. The Idea is, the fewer modifications made to the machine, the less chance the intruder will find something suspicious about the box. Throughout the process of setting up a honey pot a few points must be considered, how to track the intruder's moves, how to be alerted of a compromise, and how to stop a hacker from compromising production servers on your network. One simple solution to this is to place the honey pot on its own subnet behind a firewall. By doing this you can log traffic, this becoming a first layer in tracking an intruders moves. Most firewall come with an alerting capability so you can setup an alert system when you decoy system is

compromised. Lastly, you can control incoming and outgoing traffic. In this case you could allow all Internet traffic in but limit outgoing traffic so the intruder cannot attack other businesses or machines from your location.

Keeping system logs is an important part of using a honey pot because of they are full of valuable data. Generally speaking it is not a good idea to keep log information on the honey pot, as sooner or later the intruder will most likely have the ability to change the data. This is why, if you want to track the intruder's moves, it is important to logging his or her activity on a different server.

Using a sniffer is also a good idea as it allows you to monitor the traffic going to and coming from the honey pot. The big advantage of a sniffer is it will pick up keystrokes and even screen shots, making it easier to see what the intruder is doing and how they are doing it.

Once the intruder has gone as far as you are comfortable with you should then kick them off and fix the vulnerability that was used. It is a good idea to fix the vulnerability as opposed to rebuilding the machine; this allows you to learn more about new and different vulnerabilities.

Some Commercial Honey Pots and helpful software

CyberCop Sting by Network Associates

> This product is designed to run on Windows NT and is able to emulate several different systems including Linux, Solaris, Cisco IOS, and NT. It is made to appeal to hackers for looking as if it has several well-known vulnerabilities.

BackOfficer Friendly by NFR

> This product is designed to emulate a Back Orifice server.

Tripwire

> This product is for use on NT and Unix machines and is designed to compare binaries and inform the server operator which have been altered. This helps protect machines from would be hackers and is an excellent way to determine if a system has been compromised.

Legal Aspects of Honey Pots

At the time of this writing there has been little legal precedence in regard to honey pots. Some feel that they are unfair and used as entrapment tools while others feel that they are a respectable way of catching the bad guy. There are some repercussions that should be noted as well. If you intentionally lure intruders to a honey pot it may be seen as an open invitation, even permission for intruders to access the decoy system, therefore you should have the same minimum security restrictions on your honey pot as on your production servers. After access to a honey pot is gained, it can be used as a stepping-stone to compromise other systems. In this case if you knew that the machine had been compromised and had not handled it in the appropriate manner (i.e. waited to see how far the hacker would go) it could be considered gross negligence because by setting up a "vulnerable" system you indirectly allowed and promoted the hacker's actions. Honey

pots should generally be used as defensive detection tools, not an offensive approach to luring intruders.

Is it worth all the trouble?

In order to answer the above question you must first ask yourself a few other questions.

1. Does your corporation have enough resources to dedicate a system or two as a honey pot host?
2. Do you monitor system and intrusion-detection system logs?
3. Is there any intention to prosecuting or tracking intruders?
4. Is there a proper incident response capability?

Honey pots are highly useful if the time and resources are available. They are highly educational as far as learning about your corporation's vulnerabilities and for seeing into hacker's tools and tricks used to compromise systems. If the resources can be had a honey pot is a great addition to any intrusion detection setup.

Bibliography

Hartley, Bruce. "Honeypots: A New Dimension to Intrusion Detection" August 2000
URL: http://www.advisor.com/MIS

Schwartau, Winn. "Lying to hackers is okay by me: Part 9 of 9" 7 July 1999.
URL: http://www.nwfusion.com/newsletters/sec/0705sec2.html?nf

Spitzner, Lance "To Build a Honey pot" 7 June 2000
URL: http://www.enteract.com/~lspitz/honeypot.html

Bandy, Phil "What is a honey pot? Why do I need one? 1999
URL: http://www.hackzone.ru/koi8/nsp/info/ids/IDFAQ/honeypot2.htm

Forristal, Jeff "Luring Killer Bees With Honey" 21 August 2000
URL: http://www.networkcomputing.com/1116/1116ws3.html

Last Updated: May 23rd, 2017

Upcoming Training

SANS Stockholm 2017	Stockholm, Sweden	May 29, 2017 - Jun 03, 2017	Live Event
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event