Interested in learning more about security?

SANS Institute InfoSec Reading Room

This paper is from the SANS Institute Reading Room site. Reposting is not permitted without express written permission.

Honey Pots and Honey Nets - Security through Deception

This article describes a security tool and concept known as a Honey Pot and Honeynet. What makes this security tool different is that Honey Pots and Honeynets are digital network bait, and through deception, they are designed to actually attract intruders. This paper expands on the work of two SANS GSEC research papers: 'Honey Pot Systems Explained'

Copyright SANS Institute

Author Retains Full Rights

AD

Honey Pots and Honey Nets - Security through Deception

William W. Martin, CISSP

May 25, 2001

Overview

This article describes a security tool and concept known as a Honey Pot and Honeynet. What makes this security tool different is that Honey Pots and Honeynets are digital network bait, and through deception, they are designed to actually attract intruders.

This paper expands on the work of two SANS GSEC research papers: 'Honey Pot Systems Explained' - by Loras Even and 'Honey Pots and Intrusion' - by David Klug.

What is a Honey Pot?

Remember:
"There can never be enough deception."
- *Sun Tzu*

Honey Pots are fake computer systems, setup as a "decoy", that are used to collect data on intruders.

This "decoy" appears to contain operating system vulnerabilities that make it an attractive target for hackers. A Honey Pot, loaded with fake information, appears to the hacker to be a legitimate machine. While it appears vulnerable to attack, it actually prevents access to valuable data, administrative controls and other computers. Deception defenses can add an unrecognizable layer of protection.

As long as the hacker is not scared away, system administrators can now collect data on the identity, access, and compromise methods used by the intruder. The Honey Pot must mimic real systems or the intruder will quickly discover the 'decoy'. Honey Pots are set up to monitor the intruder without risk to production systems or data. If the Honey Pot works as intended, how the intruder probes and exploits the system can now be assessed without detection.

The concept of a Honey Pot is to learn from the intruder's actions. This knowledge can now be used to prevent attacks on the "real", or production systems, as well as diverting the resources of the attacker to a the 'decoy' system.

Advantages of Honey Pots:

- **Deter Attacks - Fewer intruders will invade a network that know is designed to monitor and capture their activity in detail.**
- **Divert Attackers Efforts - A intruder will spend energy on a system that causes no harm to production servers.**
- **Educate - The properly designed and configured Honey Pot provides data on the methods used to attack systems.**
- **Detect Insider Attacks - Since most IDS systems have difficulty detecting insider attacks, Honey Pots can provide valuable information on the patterns used by insiders.**
- **Create Confusion for Attackers - The bogus data Honey Pots provide to attackers, can confuse and confound.**

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Integrating and Installing Honey Pots

The better the integration of Honey Pot into your system, the more effective it will be. This must be balanced by the ability to maintain control of the installation. We don't want a compromised system to become a platform from which to launch attacks on our system or others.

Experts suggest placing the Honey Pot machine on its own network and behind a firewall or router.

The advantages include:

- **The first goal is to track the intruder's moves by gathering forensic information. Secure firewall and router logs can provide detailed information on the probes and ports of interest to the intruder.**

- **Many firewalls and routers have the ability to alert the operator whenever someone connects to the Honey Pot.**

- **Firewall and router rules can be established to protect the real network should the Honey Pot become compromised.**

Start by giving the Honey Pot an attractive name. Systems named mail, name_server, finance, archive or human resources (hr), make enticing targets for intruders. We want to integrate the Honey Pot into our actual system without placing production servers at risk.

The Honey Pot should not be normally be accessed by anyone, since it provides no legitimate services. Any connections to the Honey Pot should alert the operator. Logging showing data flowing out of the Honey Pot machine can also indicate it has been compromised.

How do we track the intruder without them knowing it? The establishment of multiple logging, or layers, provide the best solution. Logging needs to be as 'stealthy' as possible. We do not want to depend on a single layer of logging, since this could be altered or erased. Different logging views will also provide better understanding of exactly what the intruder was attempting. Most important to remember is that logs can only be trusted if their integrity can be guaranteed.

Establishment of logging on the Honey Pot itself creates a risk that the intruder will learn our logging scheme through the system configuration files. These logs and configurations could also be altered or erased if the machine is compromised. The best logging method is to create logs on a system the intruder cannot access, as well as the Honey Pot itself. A firewall or router can provide this capability.

Since logs created on the Honey Pot itself are at risk, logging should also be sent to a dedicated server using a cryptographic protocol, to mask the actual logging methods used. The logging server should be highly secured with all services turned off, and port 514 UDP blocked to prevent un-authorized logging of information from the Internet. A free open source encrypted solution is the program 'ssyslog' from Core-SDI or 'syslog-ng' from BalaBit software. Alternate logging methods for NT include 'slogger' and 'EventReporter'. A strong commercial product is the 'Secure Log Repository' product from NFR Security. Whenever possible, bogus logging configuration files should also be established on the local Honey Pot. This will help insure we capture valid information on how the system was attacked or compromised, and reduce the possibility of the intruder becoming aware of our decoy.

Another layer of logging includes using a network sniffer on the Honey Pot wire to capture all data in or out of the machine. This allows capturing the keystrokes of the intruder. The sniffer can also perform screen captures to see exactly what the intruder sees. Several different sniffers and/or IDS monitors can be used. They include Real Secure, NFR, Dragon and Snort.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

To help determine if the system has been compromised, capture an image of the original system program binaries using a tool such as TripWire and save this data remotely. Freeware tools similar to Tripwire can quickly create a database, which includes MD5 checksums, of system files for many system platforms. Use these tools to create a baseline of the system.

Remember that 'bad' things can happen on a compromised system by a knowledgeable intruder who becomes aware he/she is on a Honey Pot. Be ready to pull-the-plug, especially after all has been learned within reason. The goal is to learn how intruders' compromise a system, not to let the intruder use the Honey Pot as his/her tool and cause further damage. Part of the responsibility in establishing a Honey Pot, is to carefully monitor the activity on the decoy. A system that begins to launch attacks on Friday night at 11 PM must be addressed immediately. No system administrator wants to explain to his boss on Monday morning how this device, implemented and sold to management as a product to increase security, was then used by some hacker against them all weekend. Use the e-mail or pager alert feature contained in many firewalls.

To limit the scope of attacks that could be launched from a compromised Honey Pot, establish rules on the firewall for outbound traffic. Allow any type of traffic inbound from the Internet, but only allow outbound traffic such as ICMP, DNS (UCP) and FTP. The intruder may become wary, but this prevents many of the nastier hacker tools from working.

Consider making a disk image backup of the original 'clean' system install with a disk utility such as Norton 'Ghost'. This can be used to 'reset' the Honey Pot to a known state after the data is collected on the compromised system, or if the administrator completely loses control of the machine. The down side is that the intruder will know something is wrong and avoid the decoy in the future.

Once a compromised Honey Pot is 'reset', consider fixing the vulnerabilities that were used by the intruder. You can then learn new attack methods.

Honeynet Project

A group of security professionals has expanded on the Honey Pot concept and created a project dedicated to learning the tactics, tools, and motives of the blackhat (hacker) community and sharing the knowledge they learn. The project is called 'The Honeynet Project, and can found on the web at URL http://project.honeynet.org.

While a Honey Pot can be a single machine, the Honeynet is a network, where all inbound and outbound data is analyzed and collected. Within this network, a wide variety of standard production systems are established. These systems provide real services, so they more closely match the actual conditions found in many organizations today. This can make a Honeynet harder to detect, since it does not just mimic services like Honey Pots. Future plans include mixing the Honeynet into live production systems, making the Honeynet even harder to detect.

The goals of this project are twofold:

1) **To raise awareness of threats and vulnerabilities that exist on the Internet.**

2) **To teach and inform security professionals.**

The site contains a wealth of information including a library of white papers on security topics, forensic data collection and passive fingerprinting data analysis. Also included is information on the decoding and makeup of various network scans used by intruders. This information library can be found at URL: http://project.honeynet.org/papers/.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Commercial Honey Pots

CyberCop Sting by Network Associates

Simulates MS Windows NT, Sun Solaris, and Cisco routers.

"Available as a standalone product, as part of the CyberCop Intrusion Protection suite, and as part of the groundbreaking Active Security solution, integrating our best-in-class firewall, intrusion protection, antivirus, and helpdesk products around a secure Event Orchestrator."

Network Associates

See URL: http://www.cybercop.co.uk/cybercop/sting/default.htm

ManTrap by Recourse Technologies

This product uses the Honeynet concept by creating an entire network of deception hosts. Uses the Sun Solaris 2.7 or 2.7 OS to simulate a Solaris environment to intruders.

"Recourse Technologies' ManTrap may be the best-known commercial example of this tool, which is ideal for collecting evidence to prosecute system crackers while keeping your systems running at the same time."

InfoWorld Test Center
By P.J. Connolly

See URL: http://www.mantrap.com

Deception Tool Kit (DTK) - Fred Cohen and Associates

A freeware product for Linux platforms. Requires a C compiler and a PERL interpreter. DTK also requires TCP wrappers, found at URL ' http://www.porcupine.org/', for the "Generic.pl" program.

From the DTK web site URL ' http://www.all.net/dtk/faq.html', DTK uses the following components:

- Generic.pl - a generic interface that works via TCP wrappers to service incoming requests.
- listen.pl - a port listener that listens to a port and forks slave processes to handle each inbound attempt.
- logging.pl - the subroutines and initialization for logging what happens.
- respond.pl - the subroutine for responding based on 'response' file content.
- notify.pl - a sample program to notify administrators of known attacks by email.
- coredump.c - produces a coredump message on a port (what a fakeout).
- deception.c - working on a C version of the program - don't even think about compiling it yet.
- makefile - makes the C programs into executables - truly trivial.
- [nn].response - the responder finite state machine for each port. This takes some understanding of finite state machines and will be detailed later in this document.
- @[nn].[something] - a response file for non-trivial outputs.
- @fake.passwd - a fake password file that nobody will ever be able to decode.
- expandlog.pl - expands compressed logfiles into more readable form

DTK can be found at URL: http://www.all.net/dtk/dtk.tar

Integrity Tools for Honey Pot and Honeynet Administration

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

The following section describes software tools that can be used to verify the file integrity on the system used as a Honey Pot. These utilities assist the administrators in spotting a Honey Pot systems comprise.

Tripwire - Tripwire Inc.

Originally written in 1992 by Dr. Eugene Spafford and Tripwire CTO, Gene Kim. This is not a Honey Pot or Honeynet application, but a commercial software product used to verify the integrity of system binaries and inform the operator of changes. It is now available for all major operating system platforms, including Windows NT, Windows 2000, UNIX, and Linux.

Tripwire version 2.2.1 for Linux is available as freeware at URL:
http://www.tripwire.com/products/linux/221.cfml' or as Unix freeware at URL:
http://www.tripwire.com/downloads/tripwire_221/.

See URL: http://www.tripwire.com

INTACT - Pedestal Software

Detects changes in systems in real time. Changes can trigger used defined actions, such as executing batch files, reloading system files, sending alerts or performing a shutdown. The enterprise version of INTACT uses the ODBC protocol to log change detection records to Oracle and MS SQL data base servers.

See URL: http://www.pedestalsoftware.com/intact/index.htm

INTEGRIT - SourceForge Project by Edward Cashin

An alternative to file integrity verification programs like tripwire for the POSIX (Unix) operating system. An Open Source development project. A tool to detect compromised POSIX (Unix) system Honey Pots.

Praise for INTEGRIT on Freshmeat.net:

> by Karellen - Jan 6th 2001 17:15:58
>
> "This tool is pretty nice and it has most of the things I wanted from a file integrity verification system: constant databases, file attributes like inode, permissions, number of links, uid, gid, file size, access and modification times, and of course SHA checksums. It's statically linked with OpenSSL and CDB, so things don't get messed up if someone poisons your libs. Very simple config file syntax (syslog.conf like) and checksum generation for the current/known state database so you know if it's been tampered with. See the homepage for more info. Keep up the good work, I'd like to see this included in Debian ;*)"

See: http://integrit.sourceforge.net/
and
http://sourceforge.net/projects/integrit/

SAMHAIN - Samhain Labs

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

An open source file integrity and intrusion detection system that uses cryptographic checksums of files to detect modifications. Samhain a signed audit trail and signed database to provide a high level of tamper resistance, multiple logging facilities, and the ability to run as a daemon process. On networks, Samhain supports centralized monitoring of multiple hosts using a central log server. Network client/server connections are authenticated, signed and encrypted. Client status is provided via HTML pages.

Samhain has been tested on Linux, FreeBSD, AIX 4.x, HP-UX 10.20, UnixWare 7.1.0, Solaris 2.6, and Alpha/True64. On Linux, Samhain can detect *kernel module rootkits*, i.e. rootkits implemented as loadable kernel modules.

See URL: http://www.la-samhna.de/samhain/index.html

SIDEKICK - Sun Microsystems

A free toll designed to automate the collection of MD5 signatures on Solaris systems. Uses several file collection methods to catalog special files such as set-UID and set-GID file types. Can assist in rootkit detection on Solaris Honey Pots.

See URL: http://www.sun.com/blueprints/tools/fingerprint_license.html

Honey Pots and the Law

Opinions vary with regard to the legal worth of data Honey Pots and Honeynets collect. Many laws require one to show financial loss. Since they are not production systems, it is difficult to show a financial loss due to intruders accessing such systems. One can argue that the investment in the Honey Pot setup and monitoring is a security asset, and as such, is a financial asset. It can also be argued that since they contain no 'real' data, and are not 'real' systems, no financial loss can be shown from the intrusion.

Things change for everyone if a compromised Honey Pot or Honeynet becomes a springboard to launch additional system attacks. In this scenario, the intruder who compromised the Honey Pot certainly assumes additional legal responsibility. It can also be argued that the establishment of a 'vulnerable' system contributed to the problem and could constitute 'gross negligence' by the establishers of the decoy systems. Security 'experts' should have known and addressed the risks.

The possibility of an out-of-control Honey Pot or Honeynet dictates that operators closely watch their decoy systems. Do not simply establish this system and then ignore it. Be sure the resources are in place to monitor and control the setup.

Summary

Honey Pots and Honeynets are tools to acquire knowledge. The education they provide is their most important contribution. They also require substantial resources to operate correctly. If the operators understand what is demanded, Honey Pots and Honeynets can provide a fantastic learning tool in computer security.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Bibliography

The Honeynet Project, "Know Your Enemy ; Honeynets" 21 April 2001
URL: http://project.honeynet.org/papers/honeynet/

Verton, Dan "Inside Monitoring" 19 March 2001
URL: http://www.computerworld.com/cwi/

Forristal, Jeff "Luring Killer Bees With Honey " 21 August 2000
Email: jeff@neohapsis.com

Global Integrity :Security Study Validates Honeypot Effectiveness" 24 October 2000
URL: http://www.recourse.com/news/press/releases/r102400.html

Cohen, Fred "Deception Toolkit FAQ" 25 May 2001
URL: http://www.all.net/dtk/faq.html

Schwartz, Mathew "To Trap a Thief" 2 April 2001
URL: http://www.computerworld.com/cwi/story/0,1199,NAV63_STO59072,00.html?RCKEY=73

Messmer, Ellen "Decoy Nets" 5 March 2001
URL: http://www.nwfusion.com/news/2001/0305honeypot.html

Piscitello, David "Honeypots: Sweet Idea, Sticky Business" TISC Insight, Volume 3, Issue 2
URL: http://www.tisc2001.com/newsletters/32.html

Raikow, David "Building your own Honeypot" ZDNET, 25 May 2001
URL: http://www.zdnet.com/zdhelp/stories/main/0,5594,2650750-1,00.html

Schultz, Dr. Gene "Honeypots Trap Trespassers" 26 October 2000
URL: http://www.telekomnet.com/writer_annes/10-26-00_honeypots.asp

Forristal, Jeff "Honey Pots for Sale" 21 August 2000
URL: http://www.networkcomputing.com/1116/1116ws3side1.html

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Last Updated: May 22nd, 2013

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS Malaysia @ MCMC 2013	**Cyberjaya, MY**	**Jun 03, 2013 - Jun 08, 2013**	**Live Event**
SANS Pen Test Berlin 2013	**Berlin, DE**	**Jun 03, 2013 - Jun 08, 2013**	**Live Event**
Industial Control Systems Security Training - Houston	**Houston, TXUS**	**Jun 10, 2013 - Jun 15, 2013**	**Live Event**
Security Impact of IPv6 Summit 2013	**Washington, DCUS**	**Jun 14, 2013 - Jun 16, 2013**	**Live Event**
SANSFIRE 2013	**Washington, DCUS**	**Jun 14, 2013 - Jun 22, 2013**	**Live Event**
SANS Canberra 2013	**Canberra, AU**	**Jul 01, 2013 - Jul 13, 2013**	**Live Event**
Digital Forensics & Incident Response Summit 2013	**Austin, TXUS**	**Jul 09, 2013 - Jul 16, 2013**	**Live Event**
SANS London Summer 2013	**London, GB**	**Jul 09, 2013 - Jul 16, 2013**	**Live Event**
SANS Rocky Mountain 2013	**Denver, COUS**	**Jul 14, 2013 - Jul 20, 2013**	**Live Event**
SANS Mumbai 2013	**Mumbai, IN**	**Jul 22, 2013 - Jul 27, 2013**	**Live Event**
SEC528 SANS Training Program for the CompTIA® New Advanced Security Practitioner (CASP) Certification	**Washington, DCUS**	**Jul 22, 2013 - Jul 26, 2013**	**Live Event**
SANS San Francisco 2013	**San Francisco, CAUS**	**Jul 29, 2013 - Aug 03, 2013**	**Live Event**
SANS SEC 560: Network Penetration Testing @ Bangalore 2013	**Bangalore, IN**	**Aug 05, 2013 - Aug 10, 2013**	**Live Event**
SANS Boston 2013	**Boston, MAUS**	**Aug 05, 2013 - Aug 10, 2013**	**Live Event**
Critical Security Controls Summit	**Washington, DCUS**	**Aug 12, 2013 - Aug 18, 2013**	**Live Event**
Industrial Control Systems Security Training - DC	**Washington, DCUS**	**Aug 12, 2013 - Aug 16, 2013**	**Live Event**
SANS Thailand 2013	**Bangkok, TH**	**Aug 19, 2013 - Aug 31, 2013**	**Live Event**
SANS Virginia Beach 2013	**Virginia Beach, VAUS**	**Aug 19, 2013 - Aug 30, 2013**	**Live Event**
Mobile Device Security Summit 2013	**OnlineCAUS**	**May 30, 2013 - Jun 06, 2013**	**Live Event**
SANS OnDemand	**Books & MP3s OnlyUS**	**Anytime**	**Self Paced**