Honeyd – A OS Fingerprinting Artifice

Craig Valli
School of Information and Computer Science
Edith Cowan University
Western Australia
e-mail:c.valli@ecu.edu.au

Abstract

This research looks at the efficiency of the honeyd honeypot system to reliably deceive intruders. Honeypots are being used as frontline network intelligence and forensic analysis tools. A honeypots ability to reliably deceive intruders is a key factor in gathering reliable and forensically sound data. Honeyd's primary deceptive mechanism is the use of the NMAP fingerprint database to provide bogus OS fingerprints to would be intruders. Tests conducted by the author on honeyd's ability to provide bogus fingerprints sees 78% of 704 signatures invalidated under heavy probing. However, the tests have left 152 viable signatures for producing hardened honeypot designs.

Keywords

honeypot, deception, honeyd, NMAP, fingerprinting, forensics, network forensics

INTRODUCTION

The use of honeypot systems as primary intelligence gathering tools is starting to increase. The development and utilisation of honeypots is a vital preliminary stage in advanced network forensics allowing administrators and researchers to examine behaviours of intruders. This ability to research behaviours and *modus operandi* within systems designed to be probed and attacked should allow for the development of better countermeasures and techniques to combat intruder activity.

Honeypots are deployed typically as a deceptive network for hackers to explore or attack. The prime objective of a honyepot is simply to be compromised or at least make the intruder believe they have compromised the system (Spitzner, 2002). One such tool for the deployment of honeypot systems is the honeyd honeypot. This honeypot allows for the emulation of particular server and device types via the use of operating system fingerprints derived from the NMAP fingerprint database. This research builds upon work conducted by (Gupta, 2002) and looks at the ability of honeyd to effectively deceive the NMAP OS fingerprinting program.

Intruders are employing more sophisticated techniques for gathering attack intelligence on target systems. The traditional images of an intruder using keyboard style attacks is becoming less common. The methods used by intruders are becoming more automated and stealthy with them utilising freely available scanning tools such as Nessus and NMAP to achieve probes and forward intelligence gathering for attacks (Valli, 2002).

What Is An Os Fingerprint?

Nmap uses a technique called fingerprinting that was first proposed and implemented by (Fyodor, 1998). Nmap is designed to allow individuals to scan large networks to determine which hosts are running and what services they appear to be offering. Nmap supports a large number of scanning techniques such as: UDP, TCP connect(), TCP SYN (half open), ftp proxy (bounce attack), Reverse-ident, ICMP (ping sweep), FIN, ACK sweep, SYN sweep IP protocol and Null scan. Nmap also offers a number of advanced features such as remote OS detection via TCP/IP fingerprinting, stealth scanning, dynamic delay and retransmission calculations, parallel scanning, detection of down hosts via parallel pings, decoy scanning, port filtering detection, direct (non-portmapper) RPC scanning, fragmentation scanning and flexible target and port specification.

Every operating system has a unique TCP/IP stack fingerprint when compared to other operating systems, therefore each stack will generate a different response in a TCP/IP conversation. This is because each TCP/IP stack produces different sequencing of the TCP/IP packet and will have different TCP/IP flags set. Nmap interrogates the target machine's TCP/IP stack by sending it eight different packets and observing the response from these packets. The packets that are sent follow:

> "Tseq is the TCP sequenceability test
> T1 is a SYN packet with a bunch of TCP options to open port
> T2 is a NULL packet w/options to open port

T3 is a SYN|FIN|URG|PSH packet w/options to open port
T4 is an ACK to open port w/options
T5 is a SYN to closed port w/options
T6 is an ACK to closed port w/options
T7 is a FIN|PSH|URG to a closed port w/options
PU is a UDP packet to a closed port" (Fyodor, 1998)

This test is specifically crafted to put the target machine in a position where there is a high probability that two things will occur. Firstly, that the target operating system's TCP/IP stack will respond uniquely in comparison to another operating system's TCP/IP stack and that the target operating system's TCP/IP stack will respond in a consistent manner.

The granularity achievable for this type of test is reasonably high, as an example here are 2 fingerprints both from NT4 machines that are taken from the NMAP fingerprint database. They show NMAP's ability to even discriminate between varying Service Pack Levels of the underlying operating system.

Fingerprint Windows NT 4 SP3
TSeq(Class=TD|RI%gcd=<18%SI=<2A00DA&>6B73)
T1(DF=Y%W=7FFF|2017%ACK=S++%Flags=AS%Ops=M|MNWNNT)
T2(Resp=Y%DF=N%W=0%ACK=S%Flags=AR%Ops=)
T3(Resp=Y%DF=Y%W=7FFF|2017%ACK=S++|O%Flags=AS|A%Ops=M|NNT)
T4(DF=N%W=0%ACK=O|S%Flags=R%Ops=)
T5(DF=N%W=0%ACK=S++%Flags=AR%Ops=)
T6(DF=N%W=0%ACK=O|S++%Flags=R%Ops=)
T7(DF=N%W=0%ACK=S++%Flags=AR%Ops=)
PU(TOS=0%IPLEN=38%RIPTL=148%RID=E%RIPCK=E%UCK=E%ULEN=134%DAT=E)

Fingerprint Windows NT 4.0 SP 6a + hotfixes
TSeq(Class=RI%gcd=<6%SI=<40132&>290%IPID=BI|RPI%TS=U)
T1(DF=Y%W=2017%ACK=S++%Flags=AS%Ops=M)
T2(Resp=Y%DF=N%W=0%ACK=S%Flags=AR%Ops=)
T3(Resp=Y%DF=Y%W=2017%ACK=S++%Flags=AS%Ops=M)
T4(DF=N%W=0%ACK=O%Flags=R%Ops=)
T5(DF=N%W=C00|800%ACK=S++%Flags=AR%Ops=WNMETL)
T6(DF=N%W=0%ACK=O%Flags=R%Ops=WNMETL)
T7(DF=N%W=0%ACK=O%Flags=R%Ops=WNMETL)
PU(Resp=N|Y)

NMAP also produces a port scan of available ports/services that are running on the victim device. This combined intelligence allows an attacker to quickly narrow the choices of effective attack type dependent upon the information that NMAP reports back. This forward intelligence gives an attacker several key advantages with regards to stealth. Firstly, the intruder can use fewer exploits or probes against the system as they possess a reasonable supposal of the installed operating system. Secondly, as a result of having knowledge of the operating system and by using fewer attacks the intruder may reduce the probability of detection by defensive systems such as intrusion detection systems or log file reporting systems.

METHODOLOGY

A Red Hat 7.3 server had the honeyd daemon installed upon it. The honeyd configuration was then populated with all of the OS fingerprints found in the *nmap.fingerprint* file. The NMAP fingerprint file used contained 704 unique OS fingerprints and each fingerprint was configured with a unique IP number. This allowed for all fingerprints to be uniquely identified making subsequent analysis and cross-referencing easier to achieve.

The honeyd configurations were probed using the NMAP fingerprinting program. There was an attempt to fingerprint each configured host using NMAP's six main modes. Each mode made three passes over the honeyd network to ensure consistency. The six main scan modes of NMAP used in this research are outlined below.

-sS TCP SYN scan

This technique is often referred to as "half-open" scanning as it does not open a full TCP connection. A SYN packet is sent to the victim device by the intruder as if it is going to open a real connection and waits for response. A SYN|ACK indicates that the port is listening. As RST is indicative of a non-listening port. If a SYN ACK is received from the victim a RST is sent by the intruder to close the connection. The primary advantage of this scanning technique is that fewer sites will log it.

-sT TCP connect() scan

This is the most basic form of TCP scanning. The connect() system call provided by your operating system is used to open a connection to every interesting port on the machine. If the port is listening, connect() will succeed, otherwise the port isn't reachable.

-sF -sX –sN Stealth FIN, Xmas Tree, or Null scan modes

The idea is that closed ports are required to reply to your probe packet with an RST, while open ports must ignore the packets in question. The FIN scan uses a bare (surprise) FIN packet as the probe, while the Xmas tree scan turns on the FIN, URG, and PUSH flags. The Null scan turns off all flags. If a -sF,-sX,or -sN scan shows all ports closed, yet a SYN (-sS) scan shows ports being opened, you are probably looking at a Windows box due to inconsistencies in Microsoft TCP/IP implementation. There are also a few other systems that are broken in the same way as Windows and they include Cisco, BSDI, HP/UX, MVS, and IRIX. All of the above send resets from the open ports when they should just drop the packet.

-sU UDP scans

This method is to send 0 byte UDP packets to each port on the target machine to determine which UDP ports are open on a host. If an ICMP port unreachable message is received, then the port is closed, otherwise NMAP assumes it is open.

The other NMAP command line switches used to scan the honeyd network were

-PO Don't Ping

NMAP does not ping hosts at all before scanning them which is a technique used by hackers to enumerate networks. This allows for the scanning of networks that don't allow ICMP echo requests (or responses) through their firewall or have devices/machines that block ICMP traffic.

-O OS Fingerprint

This option activates remote host identification via TCP/IP fingerprinting. It gathers information by probing the OS TCP/IP stack with specially formulated probes. The information gleaned from these probes generates a fingerprint that is then compared with the database of known OS fingerprints (the *Nmap.prints* file).

The –O option also enables several other tests. One is the "Uptime" measurement, which uses the TCP timestamp option to guess when a machine was last rebooted. This is only reported for machines, which provide this information.

The resulting NMAP log files were then edited into comma separated values for importation into various analysis tools. Table 1 shows the NMAP output, a descritpion what this means and the value that was encoded as a replacement for analysis. The outputs were imported into Excel and also into Notebook Analyst for analysis. This allowed the author to perform basic statistical analysis and also to use Notebook Analyst for pattern mathcing and grouping of outputs.

NMAP Output	Meaning	Encoded
Remote operating system guess	This typically indicated a singular operating system e.g. AIX Version 4.3	OS
Remote OS guesses	This indicated multiple operating system possibilities and listed them e.g Allied Telesyn AT-3726 Ethernet Switch: 2.1cycleA, Cisco Catalyst 1900 switch or Netopia DSL/ISDN router or Bay 350-450	MOS
No exact OS matches for host	NMAP reliably tested and was unable to create a match. It also displayed the resulting fingerprint that was discovered.	NOS
No OS matches for host (test conditions non-ideal)	Due to the test conditions NMAP was unable to ascertain the hosts operating system	NOSNI
Too many fingerprints match this host for me to give an accurate OS guess	NMAP reliably tested and was unable to narrow choices	MFP

Table 1: NMAP outputs

The results from the probes on the bogus configured hosts are in Figure 1. The results are based on the OS signature results returned by NMAP. This figure shows the number of Fingerprints reliably identified for bogus honeyd hosts tested, that is the initial configured bogus host fingerprint returned a result as OS or was found in a MOS (as per encoding schema Table 1). Hence, six fingerprints means all six probe types reported a "correct" signature back to the NMAP scanner.

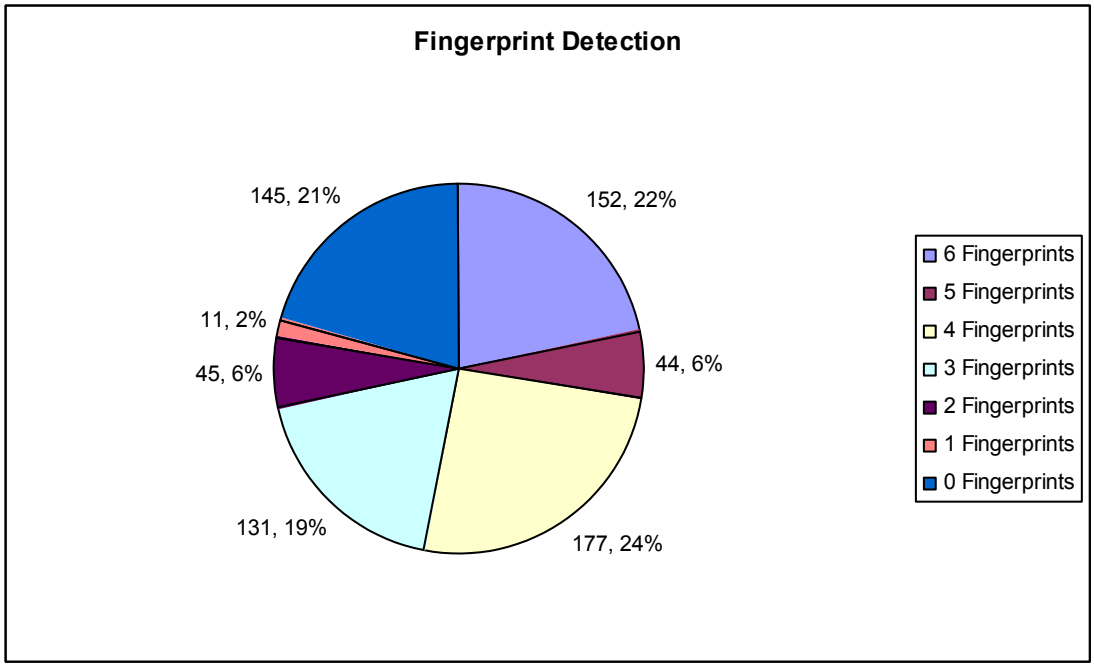

Figure 1: Fingerprint signature detection by host

It would be fair to take the premise that the faked host signatures must validate across all six major NMAP scan types to deceive a hacker. Based on this premise 78% of the fingerprints responses from the honeyd proved ineffective when subjected to NMAP scans, that is, they scored less than 6 fingerprints per host. This apparent lack of reliability in fingerprint recognition would restrict the design possibilities for a truly hardened honeynet. This lowered ability to effectively deceive an intruder would seem contrary to good honeypot design criteria.

However, keeping this in perspective it means that there are 22% or 152 signatures that will under high levels of probing by an intruder reliably present bogus OS signatures to the NMAP fingerprinting tool.

These 152 signatures were further classified into generic device categories as displayed in Table 2.

Generic Device/Type	**Count**
Computer OS	17
Hubs/Switches	10
Printers	13
Routers/Terminal Servers	51
Servers	56
Telephones	3

Table 2: Compostion of reliably fingerprinted devices

The composition of devices in Table 2 allows for a sufficently rich selection of devices from which to design a deceptive honeypot. This should allow for the production of a high quality honeynet that will provide deceptive OS signatures to a would be attacker. This allows the honeypot administrator to produce a credible hardened layer of deception or digital armor with the honeyd honeypot.

This digital armor allows for the seamless passing of the intruder through their forward intelligence phase of pinging hosts and attempted fingerprints into the next level of activity which is to the probe or attack the service/server layers in the honeypot. The intruder for all intents and purposes would believe that they are

conversing at a TCP/IP stack level with the bogus fingerprinted device. This then leads them to launch attacks at the system based on the bogus fingerprint. This should have several effects, one it will make the intruder leave a larger and more visible forensic trail due to increased activity. Secondly, it should promote intruder confidence in their attack and make them attempt higher levels of engagement with the bogus system. That is if you seek a server, find one, and all your forward intelligence tools confirm this you will be inclined to launch higher levels of attack in an attempt to compromise the server. Typically common exploits for the system would then be targeted at the server.

CONCLUSION

The research has confirmed that honeyd can produce high quality deception at levels that will largely invalidate hacker tools and techniques used for preliminary intelligence gathering. The fact that 78% of signatures are invalidated under heavy probes is not as significant as it would first appear. This means that 22% or 152 hosts produce highly deceptive signatures. This high level or failure though points to the need for extensive testing of deceptive measures such as these to ensure reliability and stability of the intended deception.

Honeyd provides competent digital armor for the outer layers of a honeypot network structure. To build upon this outer ring the next ring which is topical emulation of services has significant scope for research and investigation. The provisioning of credible armoured deception such as that provided by honeyd deceptive fingerprinting will allow for forensic recovery, investigation and research of intruders intent, motivation and *modus operandi*.

REFERENCES

Fyodor. (1998). Remote OS detection via TCP/IP stack fingerprinting. Retrieved 10 May, 2002, from the World Wide Web: http://www.insecure.org/nmap/nmap-fingerprinting-article.txt

Gupta, N. (2002). Improving the Effectiveness of Deceptive Honeynets through an Empirical Learning Approach. Paper presented at the 2002 Australian Information Warfare and Security Conference, Perth, Western Australia.

Spitzner, L. (2002). Honeypots: Definitions and Value of Honeypots. Retrieved 14th April, 2002, from the World Wide Web: http://www.enteract.com/~lspitz/honeypot.html

Valli, C. (2002). With Speed the Hacker Cometh. Paper presented at the 2002 Australian Information Warfare and Security Conference, Perth.

COPYRIGHT