2005

Blackhat fingerprinting of the wired and wireless honeynet

Suen Yek
Edith Cowan University

Blackhat fingerprinting of the wired and wireless honeynet

Suen Yek
School of Computer and Information Science Security
Edith Cowan University
syek@student.ecu.edu.au

Abstract

TCP/IP fingerprinting is a common technique used to detect unique network stack characteristics of an Operating System (OS). Its usage for network compromise is renowned for performing host discovery and in aiding the blackhat to determine a tailored exploit of detected OSs. The honeyd honeynet is able to countermeasure blackhats utilising TCP/IP fingerprinting via host device emulation on a virtual network. Honeyd allows the creation of host personalities that respond to network stack fingerprinting as a real network would. The nature of this technique however, has shown to provide inconsistent and unreliable results when performed over wired and wireless network mediums. This paper presents ongoing research into the TCP/IP fingerprinting capabilities of the popular host discovery tool Network Mapper (NMAP) on the honeyd honeynet. The forensic analysis of raw packet-captures allowed the researcher to identify differences in the modus operandi and outcomes of fingerprinting over the two mediums. The results of this exploratory study show the process of discovery to uncover how TCP/IP fingerprinting with NMAP and honeyd needs to be tested for effective network countermeasure.

Keywords

TCP/IP fingerprinting, NMAP fingerprint, NMAP signatures, honeyd honeypot

INTRODUCTION

TCP/IP fingerprinting is a scanning technique that can be used by network administrators to assess their network in addition to blackhats identifying victims to exploit. The fingerprinting technique itself involves determining any unique differences between Operating Systems (OS) and distinguishing the feature in network packets so that it may be probed (Yarochkin, 1997). An OS fingerprint should have a distinctive identification, that when recognised, reveals the platform name and version such as Microsoft Windows XP SP2.

Network Mapper (NMAP) (Yarochkin, 2004c) is a tool that contains a database of known and previously tested fingerprints by the online security community. Each fingerprint belongs to a specific OS platform that is assigned a signature, which usually is identical to the actual OS name. The fingerprints themselves show a series of TCP/IP probing packet sequences that are sent from the probing OS to the network stack of the probed OS. Probing packets contain flag settings, which are changed according to the type of probe that is sent. There is an extensive list of probes that may be sent by NMAP, mostly malformed TCP/IP packets, which aids a clandestine approach to conducting scans.

NMAP is primarily used to determine the OS running on a selected IP address. This exercise is commonly referred to as host discovery and NMAP performs this through sending probing packets to the specified ports on the OS of the target IP selected. The host name that is discovered should correspond to a signature in the NMAP database if it is known. NMAP can detect the types of services and applications that are running on the ports that it scans. Consequently, a host discovery technique that may be fine-tuned using a variety of probes aids to paint a full picture of the OS that is running on the IP of a target (Wolfgang, 2002). As the tool is freely downloadable from the World Wide Web (WWW) and highly flexible in its use, it has become a popular choice for administrators to use in addition to blackhats of various levels of sophistication (Conry-Murray, 2003; Spitzner, 2003; Yarochkin, 1997, 2002).

The fingerprints in NMAP's database are tested and contributed by the security community, and maintained by the original author Fyodor Yarochkin. When they are submitted to the database, they are tested by several parties on unknown machines and OSs. A limitation that may arise is that the fingerprint may only be effective for host discovery when performed from a specific platform OS or OSs. Consequently, when a newly released OS is used to perform TCP/ IP fingerprinting, the results may not be consistently the same.

One method for testing if NMAP can effectively fingerprint all the OS signatures in it's database is by configuring the honeyd (Provos, 2005) honeynet with all the signatures. A honeypot or honeynet is any digital entity which behaves as a genuine resource when probed or attacked (Spitzner, 2003; The Honeynet Project, 2004). The purpose of a honeypot or a honeynet, which is a network of honeypots, is to employ the characteristics of the resource it is mimicking to deceive the blackhat. Honeyd is a daemon, which creates host devices or a virtual network of devices by emulating lower layer protocols such as TCP, IP, UDP, and ICMP as examples, in addition to upper-layer protocols including FTP, TELNET, and HTTP as part of an OS's personality.

Honeyd was designed to countermeasure NMAP's fingerprinting ability, utilised by blackhats, by employing its own techniques against itself. The virtual hosts and networks are configured through templates that assign NMAP signatures to an assigned IP address. Honeyd can create many thousands of hosts, with a personality that includes services, applications, and protocol instructions for each or any specific port. For example, ports on a host may be configured to accept, drop or tarpit (prolong a connection for an inevitable time) connection attempts initiated by the probing packets sent by the blackhat.

When a blackhat sends probing packets from NMAP to honeyd they may believe they are attacking a real network of OSs because the responses that honeyd generates are identical to a real OS. This deceptive capability may act as a network countermeasure for administrators attempting to prevent blackhats from reaching the corporate network. In addition to this, the honeyd honeynet may act as a decoy to distract the blackhat while the administrator monitors their methods and identifies the goal of their endeavour. However, the effectiveness of honeyd to deceive network attackers is limited to its ability to mimic the network stack of its configured hosts.

PREVIOUS STUDIES USING HONEYD TO COUNTERMEASURE NETWORK ATTACK

In addition to the literature on the deceptive capabilities of honeyd and honeynets for decoying blackhats from genuine systems and monitoring their activities when in the honeynet, studies by Gupta (2003) presented results on the effectiveness of using honeyd as a network countermeasure. At the time, the honeynet utilised a Linux Redhat 7.3 installation and honeyd version 0.4a with subsequent upgrade to 0.5a to construct the deceptive network that was tested by voluntary participants of the study with network attacking skills.

Cyclical experiments were conducted where the honeynet was subject to network penetration testing by voluntary blackhat participants, and then upon feedback the honeynet was reconfigured to appear and behave more securely. Three rounds of testing were conducted. The blackhats concluded that the initial honeynet appeared to be unsecured and weakly configured and network logging showed high levels of network (TCP, ICMP, UDP) traffic. By the third round, the blackhats reported that the network appeared to be a well-configured corporate implementation allowing controlled levels of network traffic. These results indicated that the honeyd was effective in deceiving the blackhat while network logging allowed the researcher to monitor their activities while in the honeynet.

Subsequent research by Valli (2003) investigated how honeyd could be improved to deceive the blackhat. It was deduced that the TCP/IP fingerprinting capabilities of NMAP against honeyd was one of the crucial factors contributing to the blackhat's deception. Also using Linux Redhat 7.3 as the base testing OS and the then current honeyd version of 0.7a, Valli tested all possible NMAP signatures to determine which could be fingerprinted across five separate scan-types over a wired medium. The results showed that of the possible 704 signatures, only 152 could effectively fingerprint.

The study utilised five of NMAP's probes, SYN, FIN, UDP, NULL, and XMAS, which are explained as follows. The Synchronise (SYN) flag set in a packet that is usually sent to initiate a TCP connection. The Finish (FIN) flag set in a packet is sent to tear down or terminate a TCP connection. The User Datagram Protocol (UDP) utilises a flag set for a connectionless packet. The Null (NULL) flag has no flags enabled in the packet; and the Christmas Tree (XMAS) flag enables a combination of the FIN, Urgent Pointer (URG) and PUSH flags in a TCP packet. The URG flag indicates the packet requires urgent attention and is usually for TELNET connections. The PUSH flag indicates not to wait before sending data.

Each of the probes form part of a scan-type, which implements the flag settings in a series of packets sent to a target machine. NMAP interprets the response given by it's target. A sophisticated user of NMAP may identify with the types of responses that are produced from the scans. For example, a NULL scan-type that yields a Reset (RST) response is most likely a Microsoft Windows OS (Yarochkin, 2004b).

Of the 152 successful fingerprints, several were chosen to configure a wireless enabled honeynet using honeyd version 0.8b on a Linux Mandrake 9.0 installation (Yek, 2003). The results indicated that the OS signatures that could be previously fingerprinted over the wired medium did not fingerprint consistently or reliability over the wireless medium (Yek, 2004). At the time, the network logging facilities afforded by honeyd proved to be lacking in verbosity to identify the causes of fingerprint failure. The fingerprinting tools NMAP and Nessus (Deraison, 2005) generated reports on the outcomes of scans and vulnerability assessment.

While NMAP could not effectively scan all the OS signatures in honeyd, some could be fingerprinted across the five scan-types previously used by Valli (2003). The network vulnerability assessments performed by Nessus were mostly able to provide at least one significant weakness for a potential blackhat to exploit. The overall results showed that the network would not be effective in deceiving a blackhat and countermeasure network fingerprinting techniques. This study indicated that the TCP/IP fingerprinting of NMAP required further examination into the causes of failure particularly over the wireless medium and to determine a method for effective testing.

METHOD

In this research, the honeynet was an Athlon 1.5 GHz desktop machine with a Wireless Network Interface Card (WNIC) extending an antenna for 802.11b wireless transmission and reception. Additionally, a NIC allowed for a Category 5 (CAT-5) wired cable to be attached. The exploratory testing was conducted in a university laboratory.

A minimal installation of Linux Mandrake 10.1 was set up as the base OS for running honeyd version 1.0 (Provos, 2005) – the latest release. Mandrake 10.1 was the latest, robust version released at the time of configuration. The honeyd had a minimal installation for a reduced number of unneeded utilities, programs, and software to be running in the background of the honeynet, mitigating insecure programs and allowing greater processing power for honeyd to run. The attack machine had the same base OS, except a full installation including NMAP v.3.55 to allow the machine to run uninhibited. The attack machine was a Pentium III, 800 MHz IBM Thinkpad Laptop which had inbuilt 802.11b wireless capability, which contained a Hermes chipset to support promiscuous packet capture. The inbuilt card was later replaced (due to failure) with an Orinoco 802.11b Direct Sequence (DS) Peripheral Connect (PC) card, which operated in the same promiscuous manner.

The process for testing the TCP/IP fingerprinting was as follows. The NMAP database of OS signatures were extracted as a text file and configured as hosts in the honeyd templates. The current version of NMAP contained 988 OS signatures. The hosts imported into the honeyd templates were named `host0001` to `host0988`. These names held no significance other than representing a sequential numerical order. To spread out the hosts over a realistic network of addresses, three B class network addresses were used, which were `172.16.1.1 - 254, 172.16.2.1 - 254,` and `172.16.23.1 - 226`.

The honeynet machine was given an IP address of `192.168.1.1` on the `eth1` interface and the attack machine was given an IP address of 192.168.1.2 on it's `eth1` interface. The first tests were conducted over a wired medium using a cross over cable connecting the two machines directly to eliminate any interference from other networked devices.

On the initial start-up of honeyd with all the hosts loaded, six errors were reported indicating that the six signatures had inaccuracies, which honeyd could not recognise. These signatures were deleted from honeyd and honeyd was restarted without error. When honeyd was restarted successfully, the attack machine initiated the first round of NMAP scanning to determine host name resolution via TCP/IP fingerprinting over the wired medium, followed by the wireless medium. When the wireless scans were conducted, the interfaces were changed from `eth0` to `wlan0` on the honeynet and the attack machines. No other changes were made to the machines to mitigate the risk of confounding variables affecting the scanning results.

RESULTS OF THE FIRST ROUND OF SCANNING NMAP

For reliability, each scan-type was conducted five times on each IP address. Results showed that if a scan-type could fingerprint the OS once, it would do so on the remaining four occasions. The version of NMAP in use attributed percentage guesses for each OS that it found and ordered them alphabetically. Therefore, several OSs could be guessed, where one or more could be assigned the highest percentage and numerous others were assigned a lower percentage. The OS guess was counted as correct if the guess was one of the highest percentage guesses listed. Figure 1 shows the generated output of an NMAP scan, which found several OS matches in it’s fingerprint of 172.16.0.7. The highest percentage score of 90% for the `3Com Netbuilder Remote Office 222 router` was taken as a correct guess.

```
Warning:  OS detection will be MUCH less reliable
because we did not find at least 1 open and 1 closed
TCP port
Interesting ports on 172.16.0.7:
PORT      STATE SERVICE
1/tcp     open  tcpmux
2/tcp     open  compressnet
3/tcp     open  compressnet
...
...
...
61440/tcp open  netprowler-manager2
61441/tcp open  netprowler-sensor
65301/tcp open  pcanywhere
Device type: router|WAP|general purpose|switch

Running (JUST GUESSING) : 3Com embedded (90%), Compaq
embedded (86%), Netgear embedded (86%), Data General
AOS/VS (85%)

Aggressive OS guesses: 3Com Netbuilder Remote Office
222 router (90%), 3Com Netbuilder Remote Office 222
(ESPL-310), Version 10.1 (SW/NBRO-AB,10.1) (90%), 3Com
Netbuilder II Router Ver 11.4.0.51 (88%), 3Com
NetBuilder-II, OS version SW/NB2M-BR-5.1.0.27 (87%),
WAP: Compaq iPAQ Connection Point or Netgear MR814
(86%), AOS/VS on a Data General mainframe (85%), 3Com
SuperStack II switch SW/NBSI-CF,11.1.0.00S38 (85%)
No  exact  OS  matches  for  host  (test  conditions  non-
ideal).
```

Figure 1 – example NMAP output on scan

Out of the 982 possible OSs in honeyd, NMAP could only resolve six of the signatures and therefore, only six were effectively fingerprinted, which is shown in Table 1. In Table 1, the hostname and the IP addresses do not hold significance towards the fingerprinting result. This initial round of scanning was primarily concerned with determining which OS signatures used on honeyd could be fingerprinted by NMAP consistently and reliably over the wired and wireless mediums.

Table 1 - NMAP signatures that fingerprinted across all scan-types

HOSTNAME	IP	NMAP SIGNATURE
host0002	172.16.0.2	3Com Access Builder 4000 Switch
host0057	172.16.0.57	Apple Color LaserWriter 600 Printer
host0070	172.16.0.70	Apple Mac OS 8.5.1 (Appleshare IP 6.0)
host0186	172.16.0.186	Cisco 7206 running IOS 11.1(24)
host0341	172.16.1.87	DSL Router: Flowpoint 144/22XX v3.0.8 or SpeedStream 5851 v4.0.5.1

host0855	172.16.3.93	SCO Open Desktop 2.0

Upon assessing the remaining fingerprint results, 18 more signatures were fingerprinted at least four or more times over the five scan-types conducted, which is shown in Table 2. The wired and wireless scores out of five indicate the number of scan-types were successful. `host0001`, with signature `3Com / USR TotalSwitch Firmware: 02.02.00R` was successfully fingerprinted over three scan-types over the wire and two scan-types over the wireless medium. For the purpose of this reporting, the type of scan-type did not matter, as the goal was to fingerprint across all scan-types. However, it was found the SYN scans were the most successful.

TABLE 2 - NMAP signatures that fingerprinted four or more times across all scan-types

HOST NAME	**IP**	**NMAP SIGNARURE**	**WIRED /5**	**WIRELESS /5**	**TOTAL /10**
host0001	172.16.0.1	3Com / USR TotalSwitch Firmware: 02.02.00R	3	2	5
host0009	172.16.0.9	3Com NetBuilder-II, OS version SW/NB2M-BR-5.1.0.27	5	4	9
host0055	172.16.0.55	Apple Color LaserWriter 12/660 PS (Model No. M3036)	5	4	9
host0056	172.16.0.56	Apple Color LaserWriter 600 Printer	5	4	9
host0091	172.16.0.91	Asante FriendlyNet FR3004 Series Internet Hub	5	4	9
host0185	172.16.0.185	Cisco 7206 router (IOS 11.1(17)	5	4	9
host0258	172.16.1.4	Compatible Systems (RISC Router, IntraPort)	5	4	9
host0325	172.16.1.71	D-Link 704P Broadband Gateway or DI-713P WAP	3	2	5
host0495	172.16.1.241	IBM MVS	5	2	7
host0497	172.16.1.243	IBM MVS TCP/IP TCPMVS 3.2	5	3	8
host0505	172.16.1.251	IBM OS/390 V5R0M0	5	4	9
host0541	172.16.2.33	Lantronix EPS2 print server Version V3.5/2(970721)	5	4	9
host0558	172.16.2.50	Linksys BEFW11S4 WAP or BEFSR41 router	4	1	5
host0681	172.16.2.173	Microsoft Windows Server 2003	1	3	4
host0717	172.16.2.209	MultiTech CommPlete Controller (terminal server)	5	4	9
host0805	172.16.3.43	OpenBSD 3.0-STABLE (X86)	5	1	6
host0910	172.16.3.148	Speedstream 5871 DSL router	5	1	6
host0948	172.16.3.186	Toshiba TR650 ISDN Router	5	4	9

At this point of testing, the hard disk on the honeyd honeynet failed and a reinstall was required. The new honeyd that was configured remained as a Linux Mandrake 10.1 installation to maintain consistency; however, the NMAP signatures had been updated to include over a 1000 in total, which was previously at 988. The attack machine was also modified as the Auditor Security Collection distribution version 200605-02-no-ipw2100 which was released and tested by Valli (personal communication, September 5, 2005) to verify the reliability of the wireless security tools that were part of the distribution OS.

Further to the changes in the honeynet and attack machine, it was also decided that the testing environment was unsuitable in a laboratory that was located within an 802.11b wireless saturated location of the university. Using AiroPeek v.2.0, packet captures of the wireless traffic traversing the attack machine and honeynet showed a high number of corrupted packets. TCPdump (Network Research Group, 2004) was used to gather raw network packet captures of the wired and wireless traffic and was viewed through the network packet analyser Ethereal (Combs, 2004). The comparison between a wired Ethernet header packet and wireless header packet is shown in Figure 2 and indicates no significant errors at the TCP/IP level of network packets. The packets shown in Figure 2 are a reflection of the results of TCPdump and it was deduced that the errors occurring in the TCP/IP fingerprinting might have been at the lower network levels of the physical and data link.

The new proposed solution was a faraday cage environment, in which a decommissioned stainless steel cool-room was experimented with; however, the temperature became too hot with the equipment running and there was no HVAC to circulate the air. The subsequent decision was to build a faraday cage with inbuilt fans to regulate the heat and temperatures by circulating air out of the cage. A metal cabinet was utilised, and holes where drilled to allow for minimal cabling and a fan at the top and bottom of the back wall. The honeynet and packet capture machines were moved into the faraday cage and the attacking laptop sat on top. When the cage was closed, it was able to keep out external 802.11b wireless traffic while the antenna reception and transmission was unaffected inside the cage.

Source Port **61493**		Destination Port **http (80)**	
Sequence Number **0**			
Acknowledgement Number **0**			
Length **20**	Unused	Flags **0x0010** **ACK**	Window Size **3072**
Checksum **0x6b71 (correct)**		Urgent Pointer **Not set**	
Type	Length	Data	

Wireless IP header packet from the attack machine – frame# 13

Figure 2 Comparison of a wired and wireless network packet

Wired TCP header packet from the attack machine – frame# 13

Version **4**	Header Length **20**	Type of Service **Differentiated service field**	Total Length **40**
Identification **0x911a (37146)**		Flags **0x00**	Fragment Offset **0**
Time to Live **46**	Protocol **TCP** **(0x06)**	Header Checksum **0x8dfa (correct)**	
Source IP Address **192.168.1.2**			
Destination IP Address **172.16.0.1**			
Options (if any)			

Several benefits were found when the NMAP scans were retested inside the faraday cage. Firstly, none of the scans took more than one to two minutes to complete, whereas prior to testing in the cage, some wireless scans took up to 60 hours. It was deduced that the attack machine might have been performing a Denial of Service (DoS) onto the honeynet. Network packets may have been lost in transmission or bits in the packets may have changed due to the wide area exposed to the wireless network packets. High numbers of corrupted packets were detected by AiroPeek to support this theory.

Additionally, the shorter scan times could be attributed to the faraday cage negating the outside wireless transmissions of the university's wireless Local Area Network (LAN). Another significant difference was the upgrade of NMAP as part of using the Auditor distribution OS as the attack machine. NMAP version 3.75 was pre-installed on Auditor and the subsequent changes included no percentage scores attributed to OS guesses. NMAP now reported all the OSs that it guessed as most likely and did not report on possible other guesses. These OSs were also organised in alphabetical order; therefore, if a guess was placed third in the list, it did not indicate it was a third guess. All the guesses had equal weighting.

RESULTS OF THE SECOND ROUND OF SCANNING NMAP

An odd result that occurred in this round of testing was that the original six OS signatures that fingerprinted successfully across all scan-types, did not succeed when tested again in the faraday cage. The 18 tentative OS signatures that tested either four times or more in the laboratory tested more successfully in the cage. Table 3 shows the results of the testing in the faraday cage. For this testing, all the redundant OS signatures were removed from honeyd, and only the signatures that were to be tested this time were included in the honeyd configuration file. Consequently, only 24 signatures were incorporated as `host01` to `host24`. The first six OS signatures were those that tested effectively during the first round of scanning.

Table 3 – NMAP signatures tested in faraday cage

HOST NAME	IP	NMAP OS SIGNATURE	NUMBER CORRECT FINGER-PRINTS /5
host01	172.16.0.1	3Com Access Builder 4000 Switch	5
host02	172.16.0.2	Apple Color LaserWriter 600 Printer	5
host03	172.16.0.3	Apple Mac OS 8.5.1 (Appleshare IP 6.0)	3
host04	172.16.0.4	Cisco 7206 running IOS 11.1(24)	2
host05	172.16.0.5	DSL Router: Flowpoint 144/22XX v3.0.8 or SpeedStream 5851 v4.0.5.1	5
host06	172.16.0.6	SCO Open Desktop 2.0	3
host07	172.16.0.7	3Com / USR TotalSwitch Firmware: 02.02.00R	0
host08	172.16.0.8	3Com NetBuilder-II, OS version SW/NB2M-BR-5.1.0.27	5
host09	172.16.0.9	Apple Color LaserWriter 12/660 PS (Model No. M3036)	5
host10	172.16.0.10	Apple Color LaserWriter 600 Printer	5
host11	172.16.0.11	Asante FriendlyNet FR3004 Series Internet Hub	2
host12	172.16.0.12	Cisco 7206 router (IOS 11.1(17)	5
host13	172.16.0.13	Compatible Systems (RISC Router, IntraPort)	5
host14	172.16.0.14	D-Link 704P Broadband Gateway or DI-713P WAP	2
host15	172.16.0.15	IBM MVS	3
host16	172.16.0.16	IBM MVS TCP/IP TCPMVS 3.2	2
host17	172.16.0.17	IBM OS/390 V5R0M0	2

host18	172.16.0.18	Lantronix EPS2 print server Version V3.5/2(970721)	5
host19	172.16.0.19	Linksys BEFW11S4 WAP or BEFSR41 router	0
host20	172.16.0.20	Microsoft Windows Server 2003	2
host21	172.16.0.21	MultiTech CommPlete Controller (terminal server)	0
host22	172.16.0.22	OpenBSD 3.0-STABLE (X86)	5
host23	172.16.0.23	Speedstream 5871 DSL router	5
host24	172.16.0.24	Toshiba TR650 ISDN Router	2

The results show that only three of the original six OS signatures fingerprinted effectively across all five scan-types conducted. Further testing of the additional 18 showed that eight of the signatures could now fingerprint across all the scan-types. These results firstly indicated that there was a variation with the NMAP fingerprinting because the first six should have fingerprinted effectively in the faraday if they could be fingerprinted in an open laboratory with high interference.

The possible reasons for the discrepancy in results were that NMAP v.3.55 was used for the first round of scans and during the eight-month period between testing, NMAP v.3.75 was released and pre-installed on Auditor. With each upgrade of NMAP, changes and improvements are introduced that may modify the scanning results (Yarochkin, 2004a). This is due to changes in the scanning engine and sometimes, modifications to the fingerprints. This is the most likely reason that the results differed in the six original fingerprints. An additional possibility is that the Auditor OS was used as the base OS instead of Linux Mandrake 10.1, as was previously used. This difference is however, less likely to affect the outcomes as to the change in NMAP itself.

The results that showed more effective results from the faraday testing, that is the remaining 18 OS signatures, indicated that the testing was also affected by the faraday because some previous signatures that could not fingerprint were fingerprinting successfully. The NMAP upgrade is also likely to be the reason for their testing success as the database of OS signatures and fingerprints were expanded and improvements to some scan were made (*ibid*, 2004a)

From the now eleven available signatures that could fingerprint effectively, it was proposed to design a virtual network using these signatures to build up OS personalities on eleven network hosts. However, upon closer examination of the OS signatures, it was found that none was an Access Point (AP) and only the `OpenBSD 3.0-STABLE (X86)` signature could act as a server, and none were able to act as clients. Most of the signatures that could be fingerprinted were routers. Therefore, some additional testing was required so that a more believable network could be designed that included servers, clients, and an AP.

RESULTS OF SUPPLEMENTARY NMAP SCANNING

The researcher then perused the list of fingerprints for potential OS signatures that could be used to fill out a virtual network topology. Different OS signatures were chosen based on their believability for existing as part of a network. For the proposed virtual DMZ, various Solaris, IBM, AIX, FreeBSD and NetBSD fingerprints were tested across the five scan-types and from these, only FreeBSD and NetBSD could be successfully fingerprinted on all accounts. Therefore, these OSs were chosen for the DMZ where the personalities could later be incorporated with mail, ftp and http services.

For client machines, various Microsoft Windows and Apple Macintosh desktop OSs were tested and only Macintosh signatures appeared to work. According to previous research results, Windows machines do not fingerprint when using XMAS, UDP and NULL scans over the wireless medium. This has been reported due to Windows responding in a RST for all these scan-types when no response should be the standard reaction. However, a Windows XP laptop with a centrino chip was borrowed and tested against NMAP to determine if it could be fingerprinted. The result was that NMAP was able to fingerprint the Windows laptop across all the scan-types and was able to guess the Windows XP OS correctly, but among some other OSs as well.

Lastly, serval AP signatures were tested and it was found that a `Cisco WGB350 802.11b WorkGroup Bridge` OS signature could fingerprint across all the scan-types effectively. Table 4 shows the additional signatures that were added to the honeyd virtual network.

Table 4 – Additional NMAP OS signatures

HOST NAME	IP	NMAP OS SIGNATURES
host25	172.16.0.25	Cisco WGB350 802.11b WorkGroup Bridge
host26	172.16.0.26	FreeBSD 5.0-RELEASE
host27	172.16.0.27	Apple Mac OS 7.1
host28	172.16.0.28	NetBSD 1.6
Host29	192.168.1.13	Windows XP SP2

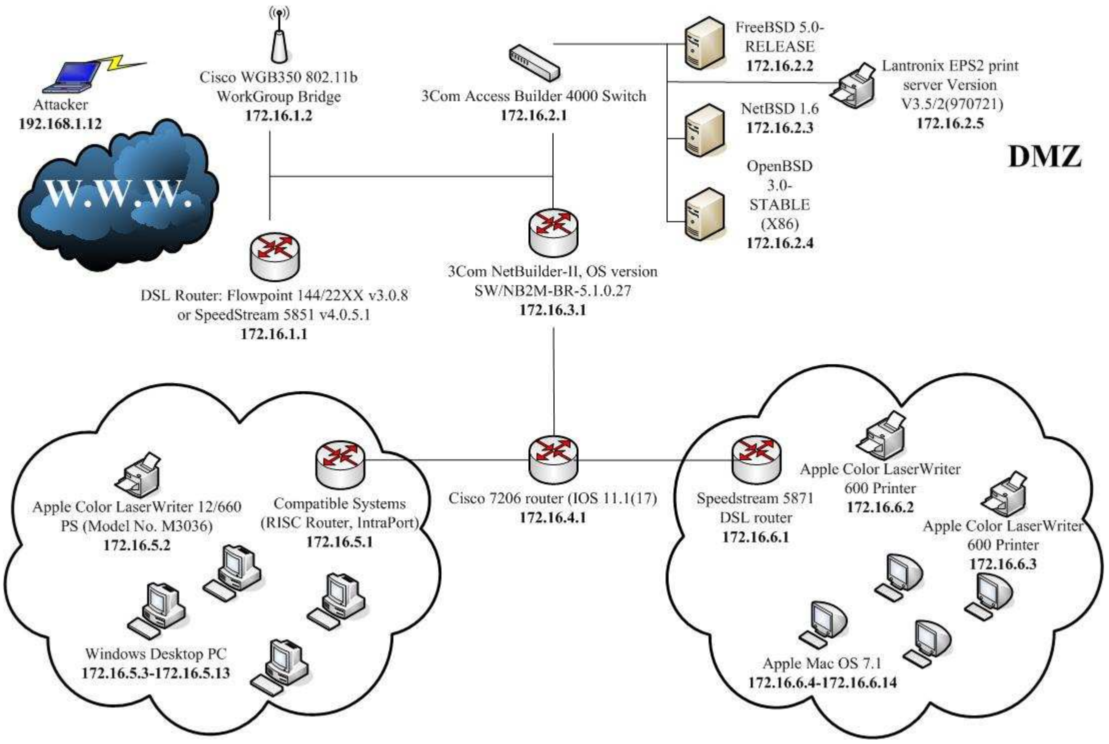

Figure 3 – Proposed honeyd network topology

The proposed virtual network of host devices including Macintosh clients, OpenBSD, FreeBSD, NetBSD servers and a mix of networking mechanisms including an AP is illustrated in Figure 3. The attack machine is also shown in the diagram as a laptop using the IP 192.168.1.12 connecting via it's wireless antenna. This network will be tested with other network fingerprinting tools such as Xprobe2 (Sys-Security Group, 2003), which functions similarly to NMAP in an active manner, and p0f (Zalewski, 2004), which is a passive fingerprinting tool that assesses captured network traffic instead of actively probing the target. It is intended that upon successful TCP/IP fingerprinting, the personalities of each host will be configured to incorporate services, applications, and routing latency.

CONCLUDING DISCUSSION

Effective TCP/IP fingerprinting is an integral part of deceiving the blackhat when performing host discovery. The honeyd honeynet implements NMAP's OS signatures and fingerprints for counteracting network-fingerprinting tools like NMAP. Ongoing research has shown that the effectiveness of NMAP fingerprinting has varied both over the wired and wireless mediums, and over varying releases of the each software in use. The best possible way of determining the best OS emulation in honeyd is to test regularly the NMAP signatures and fingerprints.

This research reported on the results acquired from NMAP primarily as a comparison between wired and wireless NMAP scans to determine where the problems have arisen. It was found the interference of other wireless activity and large room space hinders effective physical and data link layer transmission between an attacking machine performing the fingerprinting against the honeynet machine. It was then decided to test the fingerprinting in a faraday cage to ascertain differences in results.

The faraday cage proved more successful results in that 15 OS signatures could be fingerprinted, as opposed to the six originally tested. This change was attributed to significant upgrades on the NMAP program itself, from v3.55 to 3.75 in addition to the confining space and walls of the faraday, which allowed transmissions to occur faster and without interference, resulting in the significantly shorter scanning times.

The outcomes of this research are that NMAP is a continuingly developing program as are honeynet architectures. However, the implementation of NMAP signatures in honeyd is the most significant change that affects the TCP/IP fingerprinting ability of the deceptive network. Therefore, effective TCP/IP fingerprinting is more reliant on the environment in which it is performed, whether there is interference or a far range for signals to travel, in addition to the version of NMAP used.

A final proposed test is to perform the attacks outside the faraday to determine if honeyd is still able to respond effectively where there is the outside interference. This testing may illuminate on the honeynet's ability to deceive in a potentially live and realistic environment. Interference and uncontrolled signals are realistic obstacles for the blackhat and are subsequent problems for the defender when attempting to employ deceptive network countermeasure against TCP/IP fingerprinting. This can be further verified by inviting guest attackers to penetration test the network and determine if these deceptive mechanisms work on a real blackhat!

REFERENCES

Combs, G. (2004). Ethereal (Version 0.9.14) [Network packet analyser].

Conry-Murray, A. (2003). *Vulnerability assessment tools find new uses*. Retrieved 7 October, 2004, from http://www.networkmagazine.com/shared/article/showArticle.jhtml?articleId=14400061&pgno=2

Deraison, R. (2005). Nessus (Version 2.2.5) [Network vulnerability scanner].

Gupta, N. (2003, 25 November). *Is honeyd effective or not?* Paper presented at the 1st Australian Computer, Information and Network Forensics Conference, Scarborough, Western Australia.

Network Research Group. (2004). TCPdump (Version 3.8.3) [Network traffic viewer].

Provos, N. (2005). Honeyd (Version 1.0) [Honeypot].

Spitzner, L. (2003). *Honeypots - tracking hackers*. Boston: Pearson Education Inc.

Sys-Security Group. (2003). Xprobe2 (Version 2.0.2) [Fingerprinting tool].

The Honeynet Project. (2004). *Know your enemy: learning about security threats*. Boston: Addison-Wesley.

Wolfgang, M. (2002). *Host discovery with NMAP*. Retrieved 1 September, 2005, from http://moonpie.org/writings/discovery.pdf

Yarochkin, F. (1997). *The art of port scanning*. Retrieved 1 September, 2005, from http://www.insecure.org/nmap/nmap_doc.html

Yarochkin, F. (2002). *Remote OS detection via TCP/IP stack fingerprinting*. Retrieved 1 September, 2005, from http://www.insecure.org/nmap/nmap-fingerprinting-article.html

Yarochkin, F. (2004a). *Nmap 3.75 released: less crashes and more OS fingerprints*. Retrieved 15 September, 2005, from http://seclists.org/lists/nmap-hackers/2004/Oct-Dec/0001.html

Yarochkin, F. (2004b). *NMAP network security scanner man page*.Unpublished manuscript.

Yarochkin, F. (2004c). NMAP: network mapper (Version 3.70) [Network exploration tool and security scanner].

Yek, S. (2003, 25 November). *Measuring the effectiveness of deception in a wireless honeypot.* Paper presented at the 1st Australian Computer Network & Information Forensics Conference, Scarborough.

Yek, S. (2004, 26 November). *Implementing network defence using deception in a wireless honeypot.* Paper presented at the 2nd Australian Computer Network, Information and Forensics Conference, Fremantle.

Zalewski, M. (2004). The new p0f (Version 2.0.4) [Passive OS fingerprinting tool].

COPYRIGHT