
OS Fingerprinting



-Chirag Rajyaguru
rajyagur@usc.edu

Outline

- Introduction
 - Part I
 - OS Fingerprinting concepts
 - OS Fingerprinting techniques
 - Examples using nmap
 - Part II
 - Countermeasures
 - General design
 - Techniques (IP Scrubbing, TCP scrubbing)
-

Introduction

- **OS Fingerprinting** :: Determining the type of operating system used by studying the types of packets flowing from a system. Passive OS fingerprinting only analyzes the packets. Active OS fingerprinting sends challenges to the OS and examines the type of responses
 - **TCP/IP stack fingerprinting** :: is a technique whereby you examine the response (or lack thereof) you get from a computer when requesting a TCP connection.
 - **ICMP OS Fingerprinting** :: deals with examining how a computer responds to various ICMP messages.
 - **Fingerprinting Scrubber** :: It is a new tool to restrict a remote user's ability to determine the operating system of another host on the network
-

Introduction (cont..)

- What is OS Fingerprinting.
- Why is it needed
- What are its uses
- Legitimate (i.e. harmful or not)

“Know your enemy and know yourself, and in a hundred battles you will never be defeated“

-Anonymous

Outline

- Introduction
 - Part I
 - **OS Fingerprinting concepts**
 - OS Fingerprinting techniques
 - Examples using nmap
 - Part II
 - Countermeasures
 - General design
 - Techniques (IP Scrubbing, TCP scrubbing)
-

OS Fingerprinting Concepts

- A point to remember is that any system connected to Internet is vulnerable to fingerprinting
 - Two basic types methods for fingerprinting
 - Classic/traditional method
 - Study of the TCP stack, ICMP messages, Timestamp, Syn numbers etc. (will be discussed further on)...
-

Outline

- Introduction
 - Part I
 - OS Fingerprinting concepts
 - **OS Fingerprinting techniques**
 - Examples using nmap
 - Part II
 - Countermeasures
 - General design
 - Techniques (IP Scrubbing, TCP scrubbing)
-

Classic Fingerprinting methods

Also known as banner capture

- Telnet
 - FTP
 - HTTP
 - SMTP etc.....
-

OS Fingerprinting tools

- Nmap
 - checkos
 - SIRC
 - SS
 - queso
-

Fingerprinting methods

- FIN probe
 - BOGUS flag probe
 - TCP ISN sampling
 - TCP timestamp probe
 - TCP options probe
 - TOS probe
 - ICMP error message scanning
 - ICMP message quoting
-

Outline

- Introduction
 - Part I
 - OS Fingerprinting concepts
 - OS Fingerprinting techniques
 - **Examples using nmap**
 - Part II
 - Countermeasures
 - General design
 - Techniques (IP Scrubbing, TCP scrubbing)
-

Nmap

- Three step process
 - Probe for open ports
 - Send specially formed packets as discussed above
 - Process the results against the database
 - 9 tests in total
 - 8 against TCP
 - 1 against UDP
-

Nmap Examples

```
amy~#nmap -O -sS vectra/24

Starting nmap V. 2.2-BETA4 by Fyodor (fyodor@dhp.com, www.insecure.org/nmap/)
Host (192.168.0.0) seems to be a subnet broadcast address (returned 1 extra pi
ngs). Skipping host.
Interesting ports on playground.guma.net (192.168.0.1):
Port      State    Protocol  Service
22        open    tcp       ssh
111       open    tcp       sunrpc
635       open    tcp       unknown
1024      open    tcp       unknown
2049      open    tcp       nfs

TCP Sequence Prediction: Class=random positive increments
                        Difficulty=3916950 (Good luck!)
Remote operating system guess: Linux 2.1.122 - 2.1.132; 2.2.0-pre1 - 2.2.2

Interesting ports on vectra.guma.net (192.168.0.5):
Port      State    Protocol  Service
13        open    tcp       daytime
21        open    tcp       ftp
22        open    tcp       ssh
23        open    tcp       telnet
37        open    tcp       time
79        open    tcp       finger
111       open    tcp       sunrpc
113       open    tcp       auth
513       open    tcp       login
514       open    tcp       shell

TCP Sequence Prediction: Class=random positive increments
                        Difficulty=17719 (Worthy challenge)
Remote operating system guess: OpenBSD 2.2 - 2.3

Nmap run completed -- 256 IP addresses (2 hosts up) scanned in 6 seconds
amy~#
```

Nmap Examples..

- C:\Documents and Settings\Chirag Rajyaguru\Desktop\nmap-3.81>nmap -F -sU -T4 www.usccreditunion.org
 - Starting nmap 3.81 (<http://www.insecure.org/nmap>) at 2005-03-10 12:24 Pacific Standard Time
 - Interesting ports on www-usccreditunion.usc.edu (128.125.253.144):
(The 988 ports scanned but not shown below are in state: closed)
 - PORT STATE SERVICE
 - 111/udp open|filtered rpcbind
 - 135/udp open|filtered msrpc
 - 136/udp open|filtered profile
 - 137/udp open|filtered netbios-ns
 - 138/udp open|filtered netbios-dgm
 - 139/udp open|filtered netbios-ssn
 - 161/udp open|filtered snmp
 - Nmap finished: 1 IP address (1 host up) scanned in 110.228 seconds
-

Nmap Examples..

- C:\Documents and Settings\Chirag Rajyaguru\Desktop\nmap-3.81>nmap -F -A -T4 www.usccreditunion.org
 - Starting nmap 3.81 (<http://www.insecure.org/nmap>) at 2005-03-10 12:47 Pacific Standard Time
 - Insufficient responses for TCP sequencing (0), OS detection may be less accurate
 - Insufficient responses for TCP sequencing (0), OS detection may be less accurate
 - Interesting ports on www-usccreditunion.usc.edu (128.125.253.144):
(The 1186 ports scanned but not shown below are in state: closed)
 -

PORT	STATE	SERVICE	VERSION
21/tcp	open	ftp	Sun Solaris ftpd
22/tcp	open	ssh	OpenSSH 3.9p1 (protocol 1.99)
23/tcp	open	telnet	Sun Solaris telnetd
25/tcp	filtered	smtp	
79/tcp	open	finger	Sun Solaris fingerd
80/tcp	open	http	Apache httpd 1.3.31 ((Unix) mod_fastcgi/2.2.

 -
-

Outline

- Introduction
 - Part I
 - OS Fingerprinting concepts
 - OS Fingerprinting techniques
 - Examples using nmap
 - Part II
 - **Countermeasures**
 - General design
 - Techniques (IP Scrubbing, TCP scrubbing)
-

Fingerprint Scrubber

- Used to restrict a remote user to determine the operating system of another host on the network
 - Works at both Network and transport layers
 - Its goal is to block known stack fingerprinting in a general fast, scalable and transparent manner
-

More info...

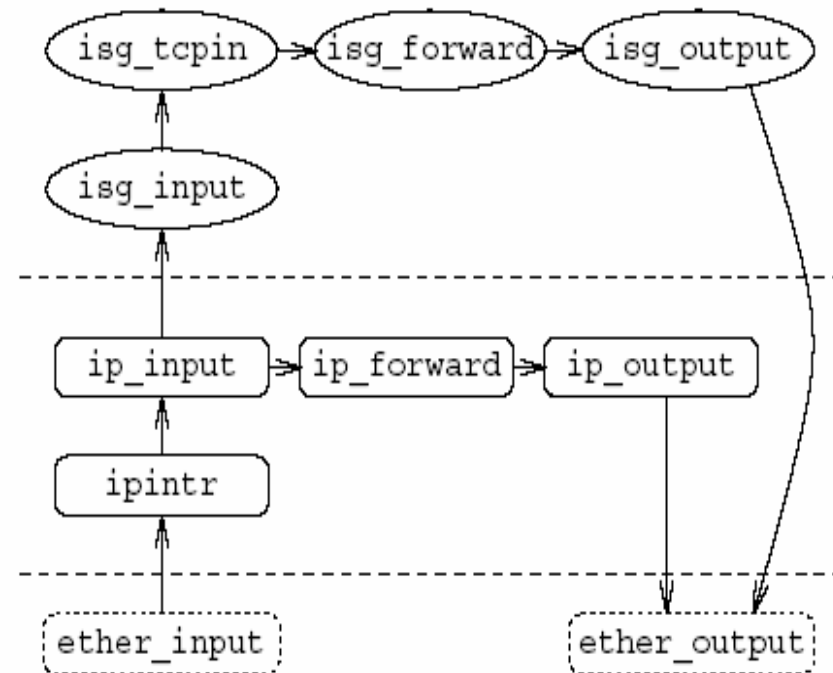
- It is put between the set of hosts/subnets to be protected and extranet
 - Most effectively implemented in a gateway machine or in firewall
 - As it scans for traffic going through itself only administrators will still be able to scan the intranet without turning off the scrubber.
-

Outline

- Introduction
 - Part I
 - OS Fingerprinting concepts
 - OS Fingerprinting techniques
 - Examples using nmap
 - Part II
 - Countermeasures
 - **General design**
 - Techniques (IP Scrubbing, TCP scrubbing)
-

Design issues

- Two interfaces (*trusted* and *untrusted*)
- Maintains some state information for TCP connections (explained further on..)
- *isg* :: *Internet Scrubbing Gateway*
- *isg_tcpin* :: *state maintenance*
- *isg_forward* :: *TCP level processing*



Outline

- Introduction
 - Part I
 - OS Fingerprinting concepts
 - OS Fingerprinting techniques
 - Examples using nmap
 - Part II
 - Countermeasures
 - General design
 - **Techniques (IP Scrubbing, TCP scrubbing)**
-

IP Scrubbing

- It normalizes IP TOS and Fragment bits in the header.
 - Done for all ICMP, IGMP, TCP and UDP packets
 - Uncommon and unused options are removed
 - DF flag is reset if the MTU is greater than the packet size
-

TCP Scrubbing

- Keeps a track of open connections
 - Reorder the TCP options to a standard/canonical format
 - Modifying the normal TCP sequence numbers
-

Timing attacks...

- Very difficult to create a generic method to defeat timing related scans.
 - Add small random delay, send packets out of order; leads to queuing and performance issues.
 - Still ineffective against timing attacks as $RTT > \text{delay introduced}$
 - Only scrubs attacks against ICMP by limiting the rate at which ICMP messages are sent across untrusted interface
-

References

- Fyodor, “*Remote OS detection via TCP/IP Stack FingerPrinting*”,
[www.insecure.org/nmap]
 - Matthew Smart, G. Robert Malan, Farnam Jahanian, “*Defeating TCP/IP Stack Fingerprinting*”, Proceedings of the 9th USENIX Security Symposium [Aug 2000]
-

Questions !!
