Introducing Nmap

Nmap is a tool used for determining the hosts that are running and what services the hosts are running. Nmap can be a valuable diagnostic tool for network administrators while they can be also a potent reconnaissance tool for the Black-hat community (Hackers, Crackers, Script Kiddies, etc). Once the network is charted out using tools like Lan MapShot, the Nmap can be used to determine the type of services and hosts running in the network.

Primary Uses of Nmap

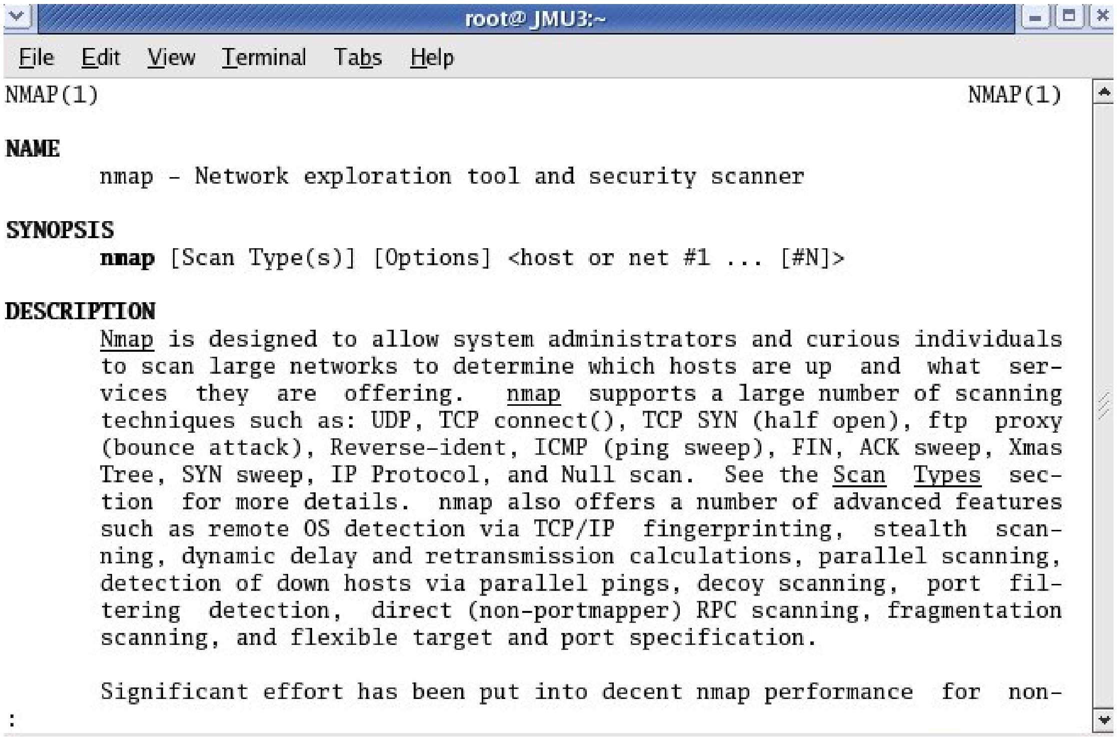
1. Determining open ports and services running in an host:
2. Determine the Operating System running on a host
3. Alter the source IP of the scan (One way is to use –S option)[1]

Nmap using Redhat 9.0

In the Cyberdefense lab scenario, it is recommended that Nmap be run from a Redhat Linux machine. Nmap can be run from a terminal using command lines or it can be run using a front end. Using the front end is more user-friendly. It also automatically shows the command being used. More information on Nmap can be obtained from the manual pages of Redhat using the command ‘man nmap’.

[1] This feature needs to be studied further

Figure 1: *man nmap* command

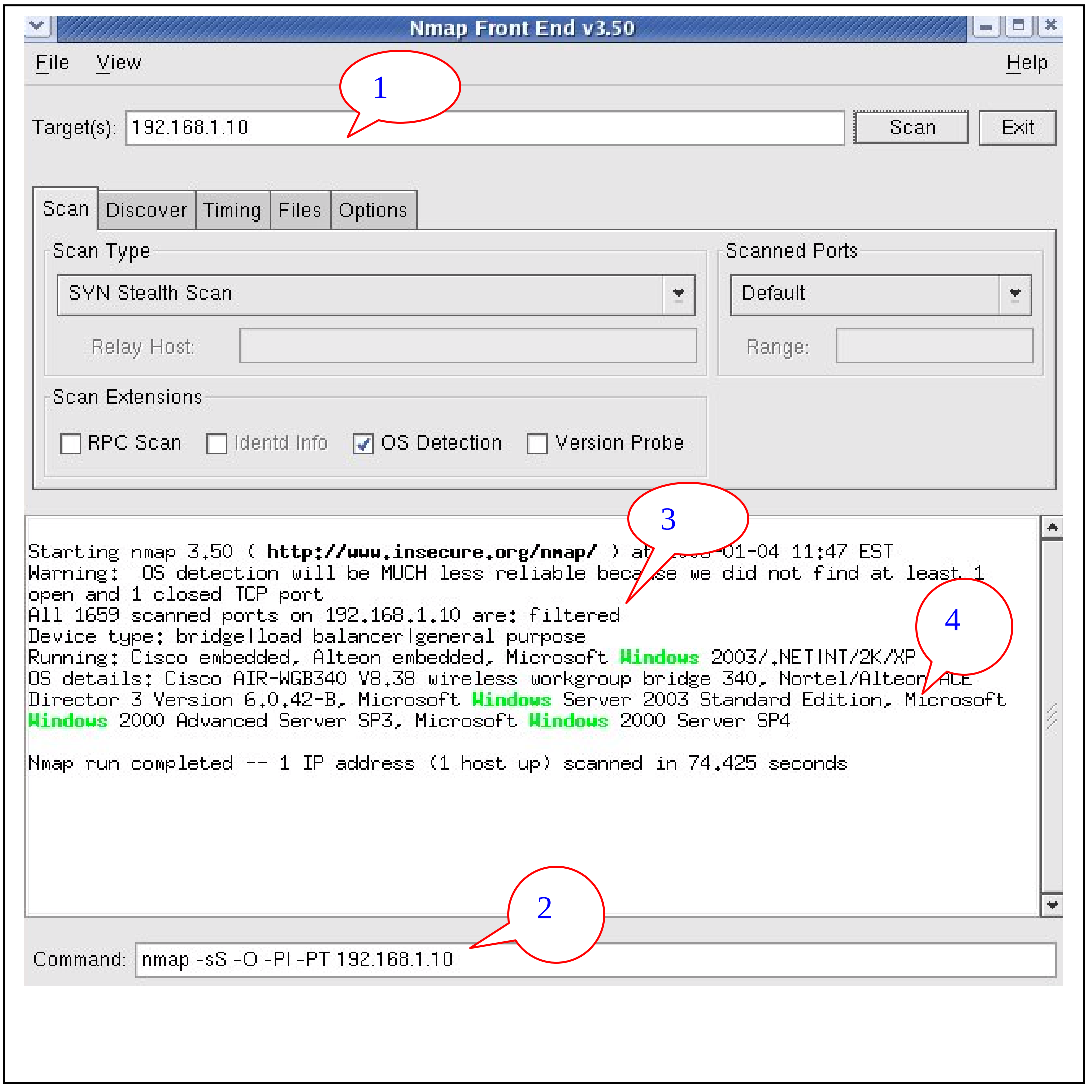


```
NMAP(1)                                                             NMAP(1)

NAME
       nmap - Network exploration tool and security scanner

SYNOPSIS
       nmap [Scan Type(s)] [Options] <host or net #1 ... [#N]>

DESCRIPTION
       Nmap is designed to allow system administrators and curious individuals
       to scan large networks to determine which hosts are up  and  what  ser-
       vices  they  are  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, Xmas
       Tree, SYN sweep, IP Protocol, and Null scan.  See the Scan  Types  sec-
       tion  for more details.  nmap also offers a number of advanced features
       such as remote OS detection via TCP/IP  fingerprinting,  stealth  scan-
       ning, dynamic delay and retransmission calculations, parallel scanning,
       detection of down hosts via parallel pings, decoy scanning,  port  fil-
       tering  detection,  direct (non-portmapper) RPC scanning, fragmentation
       scanning, and flexible target and port specification.

       Significant effort has been put into decent nmap performance  for  non-
:
```

To use Nmap in a Redhat machine follow these steps:

1. Open a terminal and type *nmapfe* to access the front end of namp. If the nmapfe command is not recognized, we need to install the program.
2. Go to Start (Red hat) → System Settings → Add/Remove Application
3. From the set of applications displayed, under the *System head*, select *System Tools* and hit *Details*, usually highlighted in blue to the right hand side.
4. From the new window that is shown, select *nmap* and *nmap-frontend*. The system will prompt for the required CDs. (Make sure you have all the three Redhat 9 CDs ready).

Figure 2: *nmapfe command* output



Target(s): 192.168.1.10

Scan Type: SYN Stealth Scan

Relay Host:

Scanned Ports: Default

Range:

Scan Extensions: □ RPC Scan □ Identd Info ☑ OS Detection □ Version Probe

```
Starting nmap 3.50 ( http://www.insecure.org/nmap/ ) at 2005-01-04 11:47 EST
Warning:  OS detection will be MUCH less reliable because we did not find at least 1
open and 1 closed TCP port
All 1659 scanned ports on 192.168.1.10 are: filtered
Device type: bridge|load balancer|general purpose
Running: Cisco embedded, Alteon embedded, Microsoft Windows 2003/.NET|NT/2K/XP
OS details: Cisco AIR-WGB340 V8.38 wireless workgroup bridge 340, Nortel/Alteon ACE
Director 3 Version 6.0.42-B, Microsoft Windows Server 2003 Standard Edition, Microsoft
Windows 2000 Advanced Server SP3, Microsoft Windows 2000 Server SP4

Nmap run completed -- 1 IP address (1 host up) scanned in 74.425 seconds
```

Command: nmap -sS -O -PI -PT 192.168.1.10

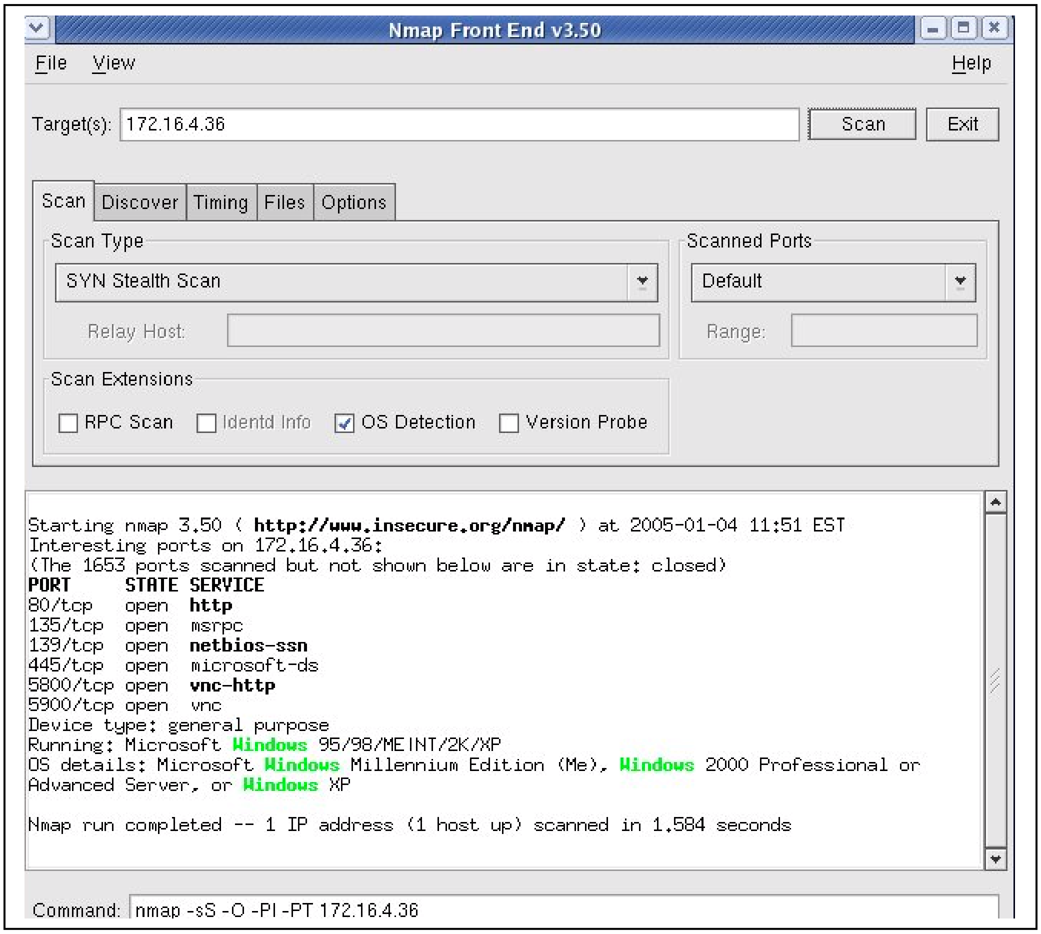
The figure shows the Nmap front end in the Fedora Core 2 which is very similar to the one in Redhat 9.0.

5. Once the front end opens (Refer to Figure 2), the Target host IP could be entered for scanning (Callout 1 in Figure 2). The command used for scanning s also shown in the front end (Callout 2 in Figure 2).

   The Nmap was run on the gateway server that is protected by IPTables firewall. Nmap identifies the ports scanned as filtered (Callout 3 in Figure 2). The major disadvantage of Nmap is also identified in Figure 2. Although the server is Redhat 9, since the server is protected by a firewall, Nmap could not identify the host operating system correctly. It identified the OS as Windows erroneously (Callout 4 in Figure 2). **This is currently a disadvantage of Nmap.**

6. Figure 3 shows an Nmap run on an unprotected host running Windows 2000.

Figure 3: Nmap on a Windows 2000 machine



Target(s): 172.16.4.36

Scan Type: SYN Stealth Scan

Relay Host:

Scanned Ports: Default

Range:

Scan Extensions: □ RPC Scan □ Identd Info ☑ OS Detection □ Version Probe

```
Starting nmap 3.50 ( http://www.insecure.org/nmap/ ) at 2005-01-04 11:51 EST
Interesting ports on 172.16.4.36:
(The 1653 ports scanned but not shown below are in state: closed)
PORT     STATE SERVICE
80/tcp   open  http
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
5800/tcp open  vnc-http
5900/tcp open  vnc
Device type: general purpose
Running: Microsoft Windows 95/98/ME|NT/2K/XP
OS details: Microsoft Windows Millennium Edition (Me), Windows 2000 Professional or
Advanced Server, or Windows XP

Nmap run completed -- 1 IP address (1 host up) scanned in 1.584 seconds
```

Command: nmap -sS -O -PI -PT 172.16.4.36

The Nmap identifies the open ports, the services, and the flavor of operating system running on the host.

Running Nmap on Windows

Nmap can be installed on windows. The installation files have to be downloaded[2] from the Internet. The two important files to be installed are as follows:

a) nmap-<version>-win32.zip[3]

b) WinPcap 3.0 stable version. (WinPcap is the packet capture library for Nmap). The download page for Nmap offers a link for downloading WinPcap.

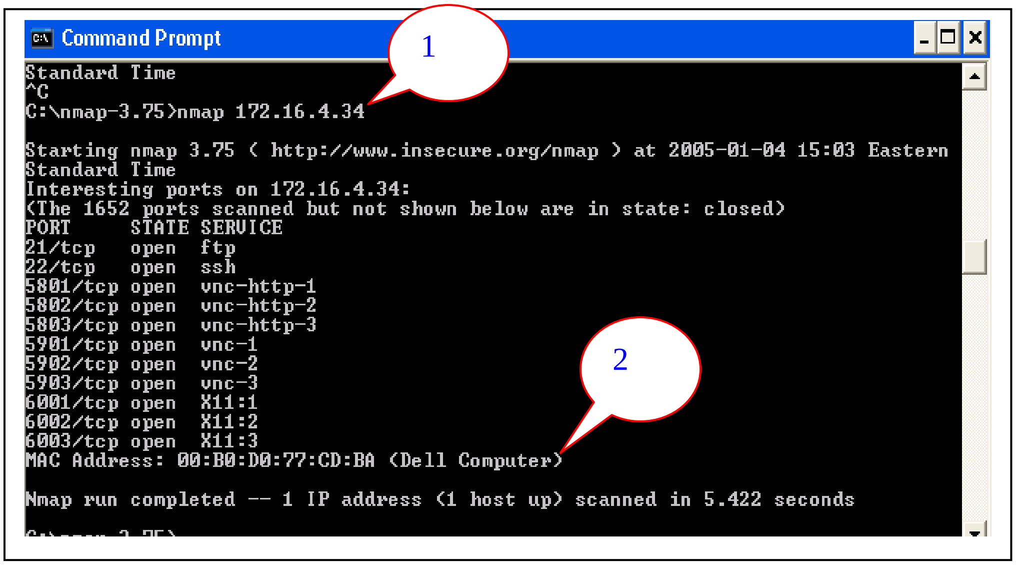
To install and run Nmap from windows follow these steps:

1. Download the installation files to a folder. Unzip the Nmap installation files to the C: drive. A new folder nmap-<version> is created in the C: drive.
2. To improve performance, it is advised that nmap_performace.reg be applied to the system registry. To do this, double click on the nmap_performance.reg in the C:\nmap-<version> folder.
3. Double click on the WinPcap install icon to install the WinPcap.
4. From the command prompt, navigate to the folder nmap-<Version>.
5. In the folder, we can run Nmap with command ‘nmap <IP address>’. Other complex Nmap commands can be run from this location.

[2] www.insecure.org

[3] Currently the downloadable version is nmap-3.75

Figure 4: nmap in Windows command prompt



```
Standard Time
^C
C:\nmap-3.75>nmap 172.16.4.34

Starting nmap 3.75 ( http://www.insecure.org/nmap ) at 2005-01-04 15:03 Eastern
Standard Time
Interesting ports on 172.16.4.34:
(The 1652 ports scanned but not shown below are in state: closed)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
5801/tcp open  vnc-http-1
5802/tcp open  vnc-http-2
5803/tcp open  vnc-http-3
5901/tcp open  vnc-1
5902/tcp open  vnc-2
5903/tcp open  vnc-3
6001/tcp open  X11:1
6002/tcp open  X11:2
6003/tcp open  X11:3
MAC Address: 00:B0:D0:77:CD:BA (Dell Computer)

Nmap run completed -- 1 IP address (1 host up) scanned in 5.422 seconds

C:\nmap-3.75>
```

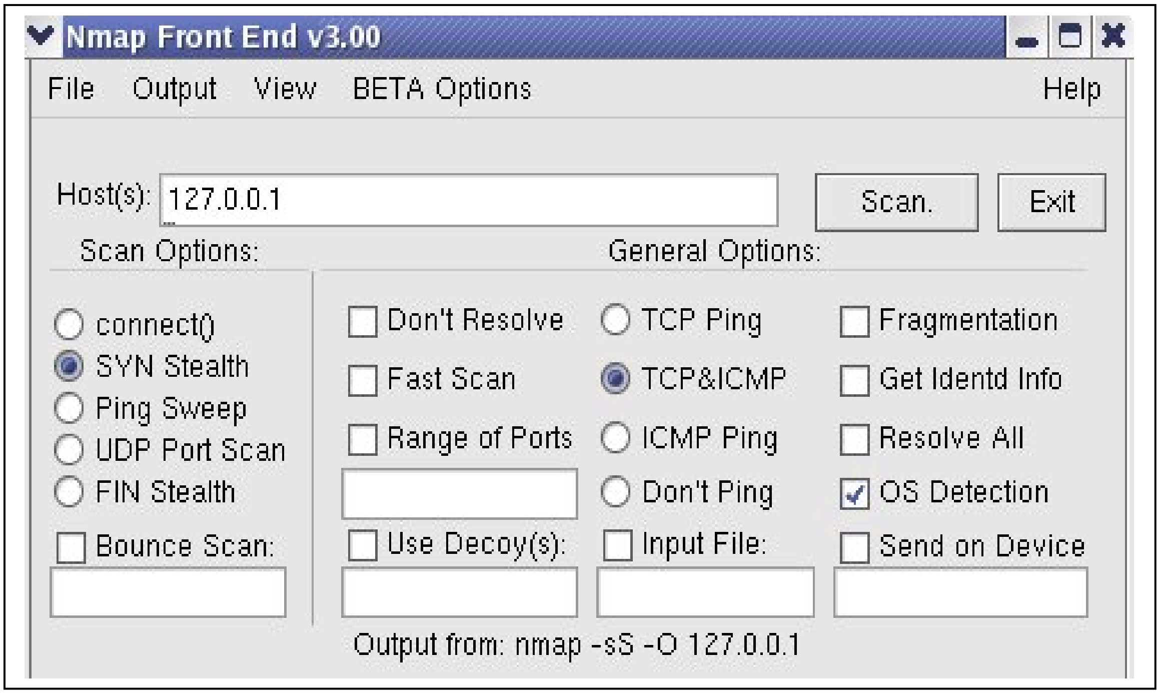
The figure shows Nmap run on a host with IP 172.16.4.34 from a Windows machine (Callout 1 in Figure 4). The make of the computer and the MAC or the NIC ‘s physical address is also detected (Callout 2 in Figure 4).

<u>The Options in Nmap</u>

Refer to Figure 5 to see the options. The screen shot is taken from a Redhat 9 machine. Some of the Nmap options are explained below:

1. <u>TCP Connect Scanning</u>: Any host can issue a *connect ()* system call to try and open an interesting port on a machine. If the port is open the call succeeds. Though it is the fastest scan it is easily detectable and blockable.
2. <u>TCP SYN Scanning</u>: The monitoring host attempts a three way hand shake but does not comple the third step, while negotiating a TCP connection. Once an acknowledgement is received from the target host, the connection is reset. SYN scanning is picked up by most firewalls and packet filters.

Figure 5: nmap options



Host(s): 127.0.0.1

Scan Options:

○ connect()
● SYN Stealth
○ Ping Sweep
○ UDP Port Scan
○ FIN Stealth
□ Bounce Scan:

General Options:

□ Don't Resolve
□ Fast Scan
□ Range of Ports
□ Use Decoy(s):

○ TCP Ping
● TCP&ICMP
○ ICMP Ping
○ Don't Ping
□ Input File:

□ Fragmentation
□ Get Identd Info
□ Resolve All
☑ OS Detection
□ Send on Device

Output from: nmap -sS -O 127.0.0.1

3. <u>TCP FIN Scanning</u>: FIN packets tend to be undetected by firewalls and packet filters. TCP property forces closed port to respond with a RST packet to a FIN packet. This property is used for scanning to determine the open and closed ports.
4. <u>Fragmentation Scanning:</u> The TCP header of the probe packet is spilt to smaller packets making it difficult for detection. But beware that this kind of scanning can cause many programs to be unstable.
5. <u>ICMP Port Unreachable Scanning:</u> The scan uses the property of the closed port sending ICMP_port_unreachable error message for closed port for detection.

.