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)

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
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

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 is 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

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 from the Internet. The two important files to be installed are as follows:

a) nmap-<version>-win32.zip

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_performance.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
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).

**The Options in Nmap**

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. **TCP Connect Scanning**: 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. **TCP SYN Scanning**: The monitoring host attempts a three way hand shake but does not complete 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.
3. **TCP FIN Scanning**: 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. **Fragmentation Scanning**: The TCP header of the probe packet is split to smaller packets making it difficult for detection. But beware that this kind of scanning can cause many programs to be unstable.

5. **ICMP Port Unreachable Scanning**: The scan uses the property of the closed port sending ICMP_port_unreachable error message for closed port for detection.