Study and Implementation of Port Scanning Techniques

Prof. J.G.Borade[1], Prof. K.N.Bapat[2], Prof. V.N.Rane[3]
[1, 2, 3] Dept of Computer Engineering
[1, 2, 3] Gharda Institute of Technology, Lavel, India

Abstract- *Essentially, port scanning involves sending a message to each port, one at a time. The kind of response received indicates whether the port is in use and can therefore be probed for weakness. Port scanning has legitimate uses in managing networks as used by the crackers as well as can be malicious in nature if some hackers are looking for a security breach in the computer system on the network. Some examples of port scanners or port scanning tools are NMap, Found stone Vision and Port scan 2000. Among them, NMap (current version: 4.76) claims the actual standard in the security industry due to its all round capabilities in port scanning. Here, we have implemented a port scanning utility which can perform the different scans and gives standard results about the port states, services running and IP hosts. The implementation of the utility is done in "Java" programming language. We describe various port scanning utilities and standard results demonstrating the success of utility for scanning the ports.*

Keywords- Port scanner; Host Discovery; NMap; Port; Port States.

I. INTRODUCTION

Port Scanning is one of the most popular techniques attackers use to discover services that they can exploit to break into systems. All systems that are connected to a LAN or the Internet via a modem run services that listen to well-known and not so well-known ports. Port scanning has legitimate uses in managing networks as used by the crackers as well as can be malicious in nature if some hackers are looking for a security breach in the computer system on the network By port scanning the attacker can find the following information about the target systems:

- What Services are running?
- What users own those services?
- The port state of target machine ports.

Once the network is charted out using tools like Lan MapShot, the port scanner can be used to determine the type of services and hosts running in the network.

1. Determining open ports and services running on a host:
2. Determine the Operating System running on a host

Our aim is to study a port scanning utility which can scan the ports of the target systems giving the information about the target hosts, the listening ports, the filtered ports, and the services running on the ports. We aim to achieve the scanning of the ports by implementing the different port scanning techniques discussed in the standard tool for port scanning, Nmap.

Port Scan is the act of systematically scanning a computer's ports[3]. Since a port is a place where information goes into and out of a computer, port scanning identifies open doors to a computer. Port scanning has legitimate uses in anaging networks, but port scanning also can be malicious in nature if someone is looking for a weakened access point to break into your computer[1].

Types of port scans:

- Vanilla:the scanner attempts to connect to all 65,535 ports
- Strobe:a more focused scan looking only for known services to exploit
- Fragmented packets:the scanner sends packet fragments that get through imple packet filters in a firewall
- UDP: the scanner looks for open udp ports.
- Sweep: the scanner connects to the same port on more than one machine
- FTP:the scanner goes through an ftpserver in order to disguise the source of the scan
- Stealth scan: the scanner blocks the scanned computer from recording the port scan activities.

In our paper first we studied what is mean by port scanning? Our paper is organized as : In II section we have put up various port states and after that studied Nmap commands for scanning on Ubuntu platform. In section III, we have given specification requirement for implementing port scanning technique in JAVA. In Section IV, we have given snapshot of the small application that is being implemented for port scanning.

II. BACKGROUND

There are hundreds of ports and services registered with the Internet Assigned Number Authority (IANA). In practice, less than one hundred are in common use. The port numbers are divided into three ranges:

Well Known Ports are those from 0 through 1023.
The Registered Ports are those from 1024 through 49151.
The Dynamic or Private Ports are the ports onwards.
Services have assigned ports so that a client can find the service easily on a remote host. For example, telnet servers listen at port 23, ssh on port 22, HTTP on port 80, and SMTP (Simple Mail Transport Protocol) servers listen at port 25. Client applications, like a telnet program or mail reader, use randomly assigned ports typically greater than 1023 [4].

Port scanning is a technique for discovering hosts‘weaknesses by sending port probes. Although sometimes used by system administrators for network exploration, port scanning generally refers to scans carried out by malicious users seeking out network vulnerabilities. The negative effects of port scans are numerous and range from wasting resources, to congesting the network, to enabling future, more serious, attacks [2]
.

A. Port States

Following Six Different Port States are recognized:

Open: An application is actively accepting TCP connections, UDP datagrams or SCTP associations on this port. Finding these is often the primary goal of port scanning. Security minded people know that each open port is an avenue for attack. Open ports are also interesting for non security scans because they show services available for use on the network.

Closed:closed port is accessible, but there is no application listening on it. They can be helpful in showing that a host is up on an IP address and as part of OS detection

Filtered: Filtering prevents its probes from reaching the port. The filtering could be from a dedicated firewall device, router rules, or host based firewall software. Filters generally drop the packets without responding. These ports cause the scan to try again and again, thus slows down the scanning.

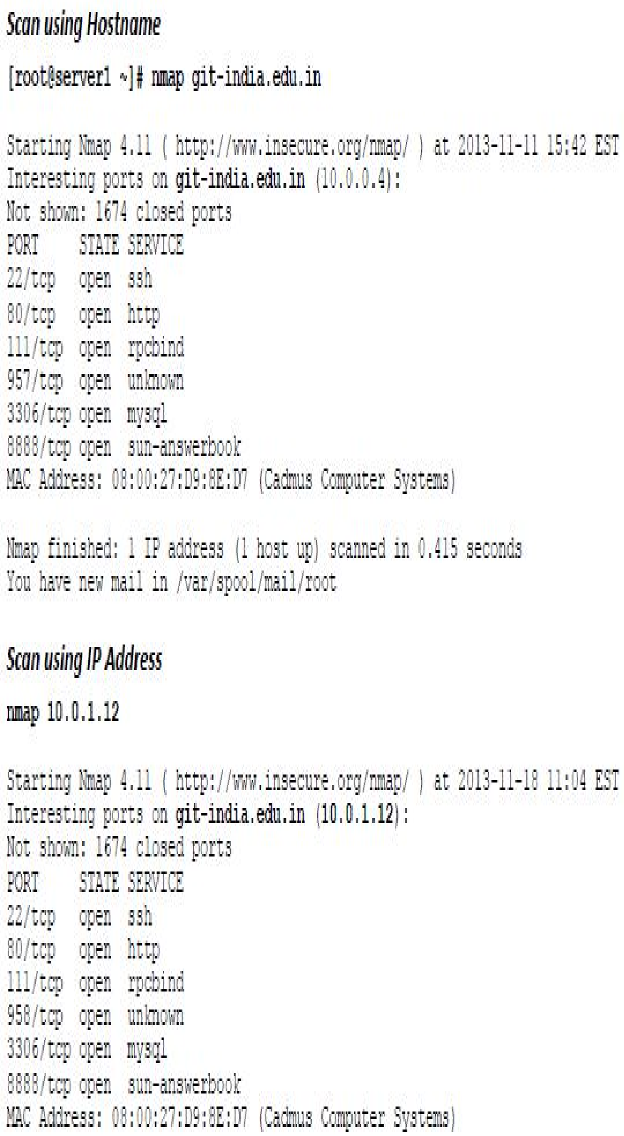
Open Filtered: This state is arrived when the scan is unable to determine whether the port is open or filtered. This occurs for scan types in which open ports give no response.

Closed Filtered: This state is arrived when the scan is unable to determine whether a port is closed or filtered [3].

B. Ports Scanning Using Nmap

Port scanning is accomplished by sending a message to each port, one at a time. The kind of response received indicates whether the port is used and can be probed for further weakness. Port scanners are important to network security technicians because they can reveal possible security vulnerabilities on the targeted system [6]. NMap (Network Mapper) is a security scanner, originally written by Gordon Lyon (also known by his pseudonym Fyodor Vaskovich), used to discover hosts and services on a computer network, thus building a "map" of the network.

NMap uses raw IP packets in novel ways to determine what hosts are available on the network, what services those hosts are offering, what operating systems they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics. Here we installed NMap on Ubuntu platform and various commands are used to scan particular host by using its username and IP address. Also using NMap we scanned various ports as shown in fig.1 and fig.2.



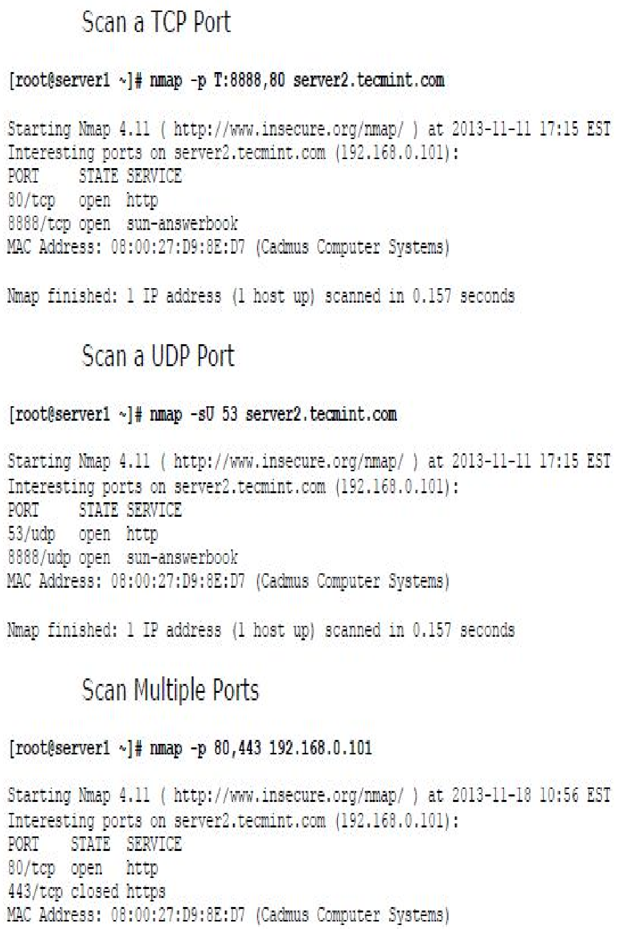
Scan using Hostname

```
[root@server1 ~]# nmap git-india.edu.in

Starting Nmap 4.11 ( http://www.insecure.org/nmap/ ) at 2013-11-11 15:42 EST
Interesting ports on git-india.edu.in (10.0.0.4):
Not shown: 1674 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
111/tcp  open  rpcbind
957/tcp  open  unknown
3306/tcp open  mysql
8888/tcp open  sun-answerbook
MAC Address: 08:00:27:D9:8E:D7 (Cadmus Computer Systems)

Nmap finished: 1 IP address (1 host up) scanned in 0.415 seconds
You have new mail in /var/spool/mail/root
```

Scan using IP Address

```
nmap 10.0.1.12

Starting Nmap 4.11 ( http://www.insecure.org/nmap/ ) at 2013-11-18 11:04 EST
Interesting ports on git-india.edu.in (10.0.1.12):
Not shown: 1674 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
111/tcp  open  rpcbind
958/tcp  open  unknown
3306/tcp open  mysql
8888/tcp open  sun-answerbook
MAC Address: 08:00:27:D9:8E:D7 (Cadmus Computer Systems)
```

figure1: Port scanning using hostname and IP address.



Scan a TCP Port

```
[root@server1 ~]# nmap -p T:8888,80 server2.tecmint.com

Starting Nmap 4.11 ( http://www.insecure.org/nmap/ ) at 2013-11-11 17:15 EST
Interesting ports on server2.tecmint.com (192.168.0.101):
PORT     STATE SERVICE
80/tcp   open  http
8888/tcp open  sun-answerbook
MAC Address: 08:00:27:D9:8E:D7 (Cadmus Computer Systems)

Nmap finished: 1 IP address (1 host up) scanned in 0.157 seconds
```

Scan a UDP Port

```
[root@server1 ~]# nmap -sU 53 server2.tecmint.com

Starting Nmap 4.11 ( http://www.insecure.org/nmap/ ) at 2013-11-11 17:15 EST
Interesting ports on server2.tecmint.com (192.168.0.101):
PORT     STATE SERVICE
53/udp   open  http
8888/udp open  sun-answerbook
MAC Address: 08:00:27:D9:8E:D7 (Cadmus Computer Systems)

Nmap finished: 1 IP address (1 host up) scanned in 0.157 seconds
```

Scan Multiple Ports

```
[root@server1 ~]# nmap -p 80,443 192.168.0.101

Starting Nmap 4.11 ( http://www.insecure.org/nmap/ ) at 2013-11-18 10:56 EST
Interesting ports on server2.tecmint.com (192.168.0.101):
PORT    STATE  SERVICE
80/tcp  open   http
443/tcp closed https
MAC Address: 08:00:27:D9:8E:D7 (Cadmus Computer Systems)
```

Figure 2a: Various port scanning using NMap tool.

III. REQUIREMENT

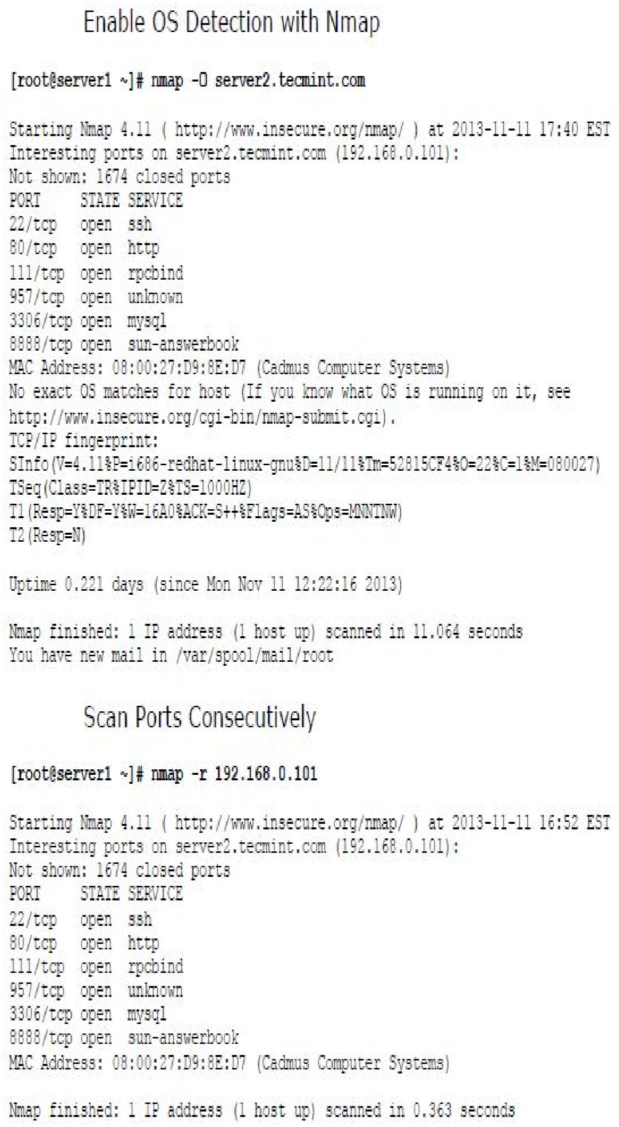
Following hardware and software are required

A. Hardware Requirements

- 1GHz or faster 32-bit(x86) or 64-bit (x64) processor
- 1GB RAM
- Internet access

B. Software Requirements

- Any Operating System
- JVM,JAVA 1.8
- Netbean 8.2



Enable OS Detection with Nmap

```
[root@server1 ~]# nmap -O server2.tecmint.com

Starting Nmap 4.11 ( http://www.insecure.org/nmap/ ) at 2013-11-11 17:40 EST
Interesting ports on server2.tecmint.com (192.168.0.101):
Not shown: 1674 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
111/tcp  open  rpcbind
957/tcp  open  unknown
3306/tcp open  mysql
8888/tcp open  sun-answerbook
MAC Address: 08:00:27:D9:8E:D7 (Cadmus Computer Systems)
No exact OS matches for host (If you know what OS is running on it, see
http://www.insecure.org/cgi-bin/nmap-submit.cgi).
TCP/IP fingerprint:
SInfo(V=4.11%P=i686-redhat-linux-gnu%D=11/11%Tm=52815CF4%O=22%C=1%M=080027)
TSeq(Class=TR%IPID=Z%TS=1000HZ)
T1(Resp=Y%DF=Y%W=16A0%ACK=S++%Flags=AS%Ops=MNNTNW)
T2(Resp=N)

Uptime 0.221 days (since Mon Nov 11 12:22:16 2013)

Nmap finished: 1 IP address (1 host up) scanned in 11.064 seconds
You have new mail in /var/spool/mail/root
```

Scan Ports Consecutively

```
[root@server1 ~]# nmap -r 192.168.0.101

Starting Nmap 4.11 ( http://www.insecure.org/nmap/ ) at 2013-11-11 16:52 EST
Interesting ports on server2.tecmint.com (192.168.0.101):
Not shown: 1674 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
111/tcp  open  rpcbind
957/tcp  open  unknown
3306/tcp open  mysql
8888/tcp open  sun-answerbook
MAC Address: 08:00:27:D9:8E:D7 (Cadmus Computer Systems)

Nmap finished: 1 IP address (1 host up) scanned in 0.363 seconds
```

Figure 2b: Various port scanning using NMap tool.

IV. IMPLEMENTATION AND SCREENSHOTS

The language used in the code is Java. A short description of the features implemented and programming interface is provided.

A. Features

- Connect_Scan technique is implemented to scan the ports by establishing a *TCP connection* through a Socket.
- Ping Scan is implemented to find the a live ports in the network traffic.
- SYN_Scan technique is implemented to determine the state of the ports using a TCP packet.
- FIN_Scan technique is implemented to determine the state of the ports using a TCP packet with the flags set as FIN.
- Multiple packets are sending to scan multiple networking IPs and ports over the network.
- For reliability, few packets are sending with the false source to prevent detection of the filters running on the target.
- For avoiding packet loss due to congestion, an option for inter packet delay is given between every packet.
- A timeout is defined for the packet to arrive from the target.
- UDP_Scan technique is implemented to determine the state of the ports using a UDP packet.

.

B.Implementatin

For implementation portscanner following packages are used :

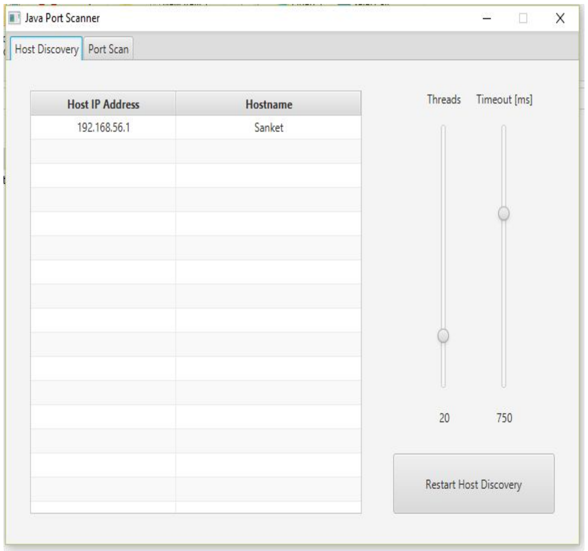
- *java.net package is used as , it* contains classes and interfaces that provide a powerful infrastructure for networking in Java
- *java.util* Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context. Contains all of the classes for creating user interfaces and for painting graphics and images.
- JavaFX is a software platform for creating and delivering desktop applications, as well as rich internet applications (RIAs) that can run across a wide variety of devices. JavaFX is intended to replace Swing as the standard GUI library for Java SE, but both will be included for the foreseeable future.

Following classes are imported from the above packages.

import java.net.InetSocketAddress;
import java.net.Socket;
import java.util.concurrent.ExecutorService;
import java.util.concurrent.Future;
import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.scene.control.TabPane;
import javafx.stage.Stage;
import java.util.logging.Level;
import java.util.logging.Logger;

C. Screen Shot

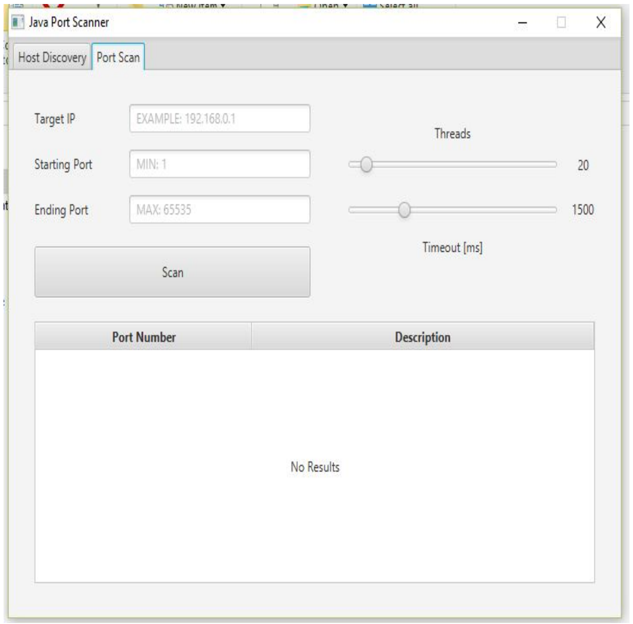
As shown in the given fig.3 and fig4, this scanner discovers active devices on the local network segment by sending a series of ARP broadcasts and incrementing the value for the target IP address field in each broadcast packet. This type of scan will have every IP device on the network respond with its own IP address in response. This scan will effectively map out an entire network.



Host IP Address	Hostname
192.168.56.1	Sanket

Threads 20
Timeout [ms] 750

figure 3. Snapshot of port scanner showing Host Discovery.



Target IP EXAMPLE: 192.168.0.1
Starting Port MIN: 1
Ending Port MAX: 65535
Threads 20
Timeout [ms] 1500

Port Number	Description
No Results	

figure 4. Snapshot of port scanner showing port scanning for particular Host for various ports.

V. CONCLUSION AND FUTURE SCOPE

Every system is vulnerable to port scanning. The best offense is a good defense. Port scanning is an important part of foot printing. Foot printing is a information gathering this software can be is most useful part to find system vulnerabilities. This tool is determining the hosts that are running and what services the hosts are running. Port scanning 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). This scanner is particularly useful in conditions where there is a need to quickly perform a port scanning. In future with some modification port scanner can be used as IP spoofing or Scan Detection.

REFERENCES

[1] P. Dokas, L. Ertoz, V. Kumar, A. Lazarevic, J. Srivastava,and P. Tan. Datmining for network intrusion detection. In Proc. NSF Workshop on Next Generation Data Mining, 2002.

[2] David J. Marchette, V. Nair, M. Jordan, S. L. Lauritzen, and J. Lawless.Computer Intrusion Detection and Network Monitoring: A Statistical Viewpoint. *Statistics for Engineering and Information Science. Springer-Verlag*, New York, 2001.

[3] Tariq Ahamad , Ahanger, Port Scan - A Security Concern, *Internationa Journal of Engineering and Innovative Technology IJEIT)*Volume 3, Issue 10, April 2014.

[4] Bryan Parno and Tony Bartoletti. Internet ballistics: Retrieving forensic data from network scans. Poster Presentation , the 13th USENIX Security Symposium,August 2004.

[5] Paxson, V.: Bro: a system for detecting network intruders in real-time. *Computer Networks (Amsterdam, Netherlands)* 31(23–24), 2435–2463 (1999)

[6] J. McPherson, K.L. Ma, P. Krystosk, T. Bartoletti, and M. Christensen. Portvis:A tool for port -based detection of security events. *In ACM VizSEC 2004 Workshop*,pages 73–81, 2004.