GAARA

> Welcome to my writeup on **gaara** from **offsec proving grounds**. This challenge has 2 flags and I am gonna walk you through the steps required to pwn the machine and capture them both. Let's get started!

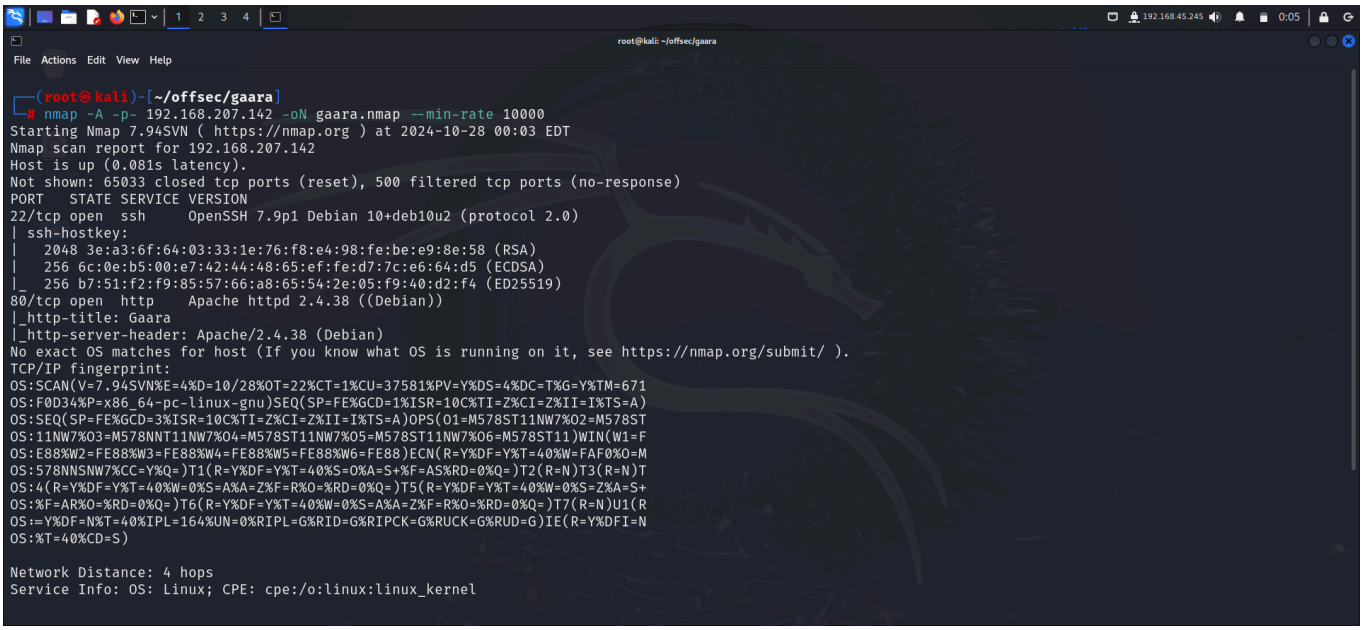
GETTING STARTED

To access the lab, visit proving grounds and download the vpn configuration file. Connect to the vpn using `openvpn <file.ovpn>` and start the machine to get an IP.

> **Note**
>
> This writeup documents the steps that successfully led to pwnage of the machine. It does not include the dead-end steps encountered during the process (which were numerous). This is just my take on pwning the machine and you are welcome to choose a different path.

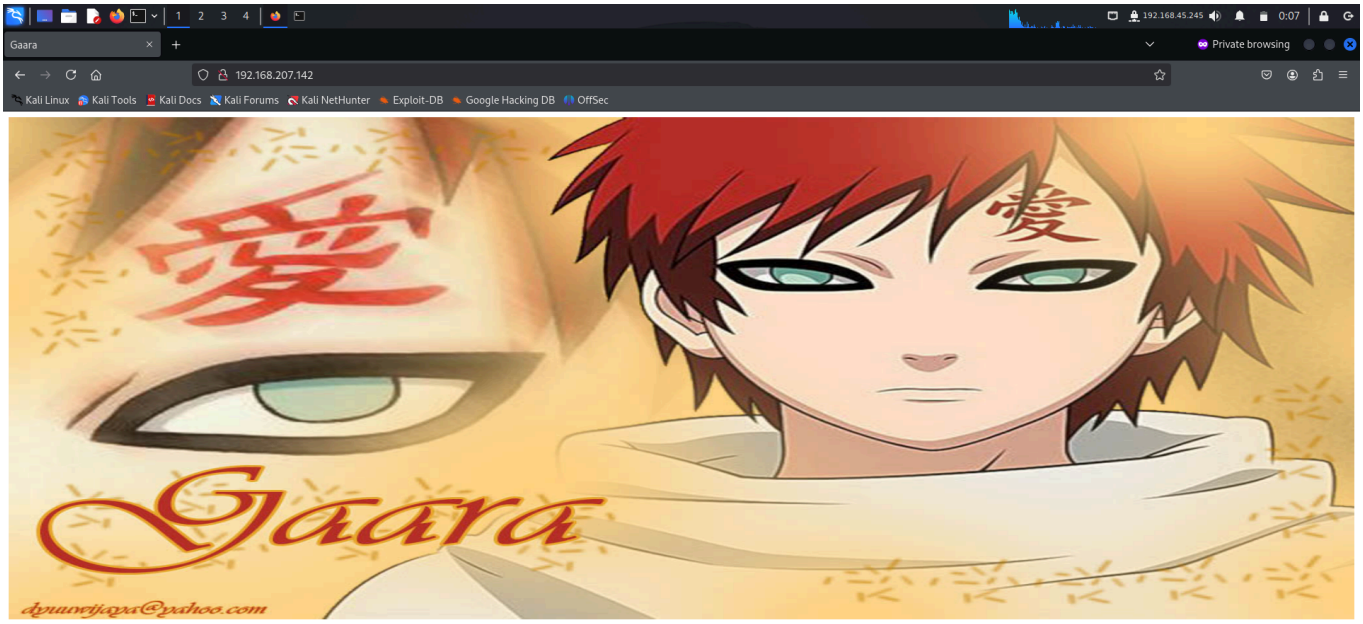
RECONNAISSANCE

I performed an **nmap** aggressive scan on the target to identify open ports and the services running on them along with some additional information.

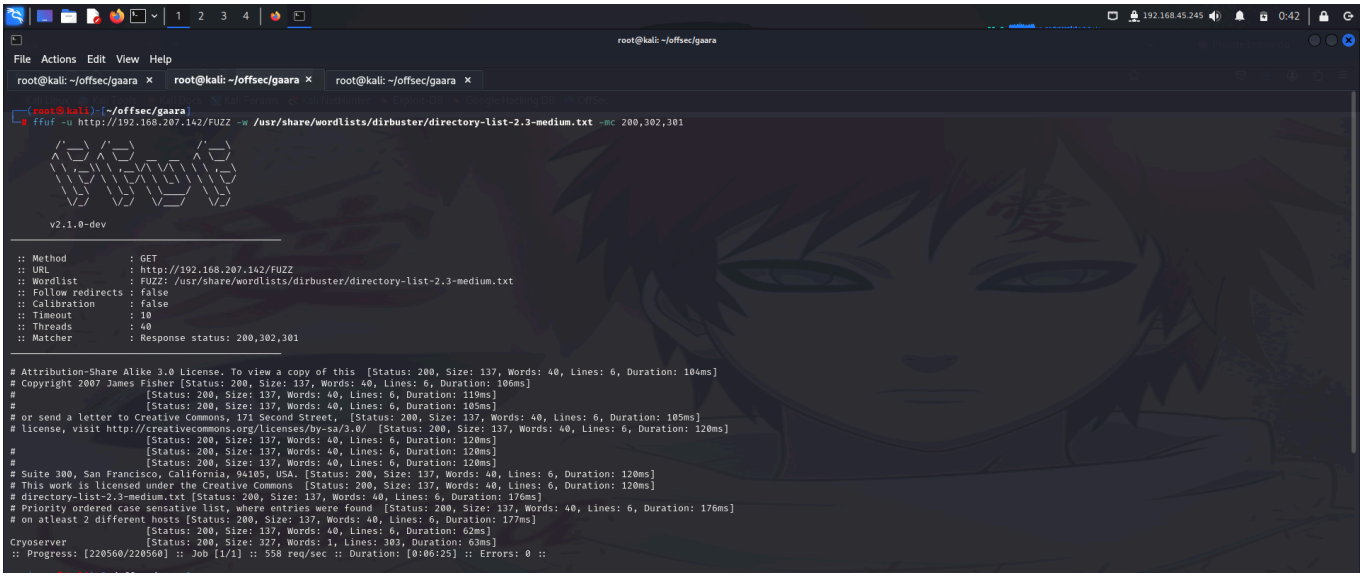


FOOTHOLD

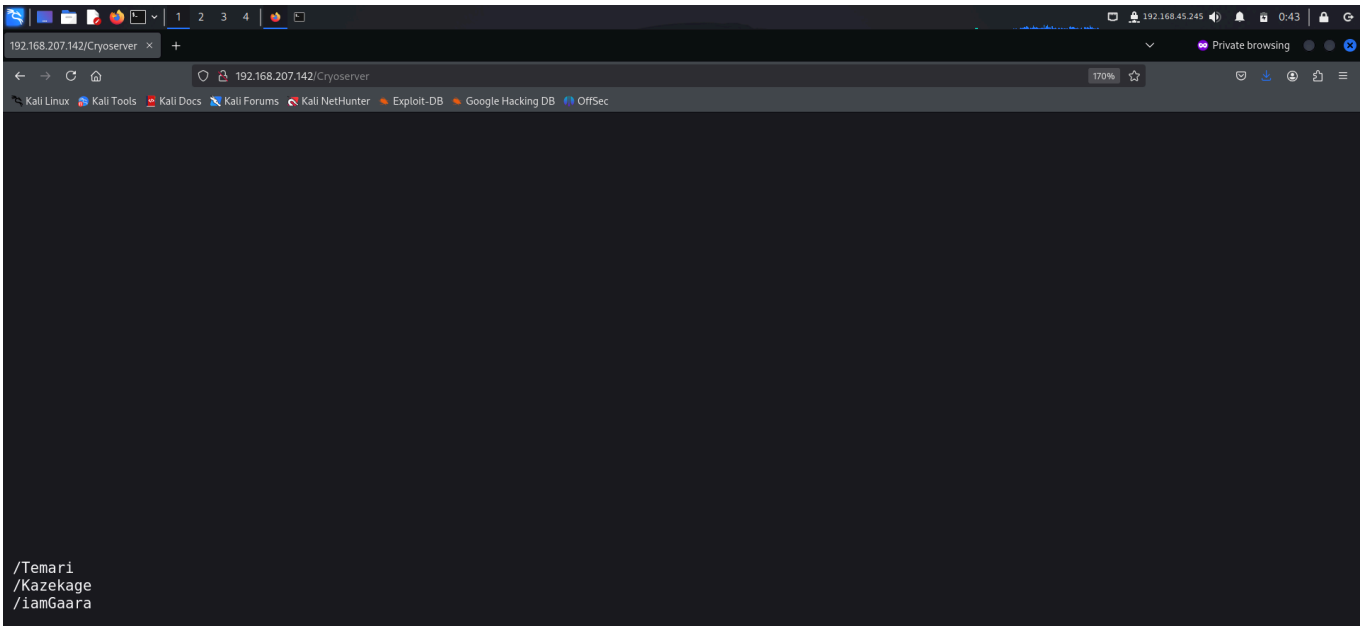
The **nmap** scan discovered an **http** server running on the target. So I accessed the site on browser.



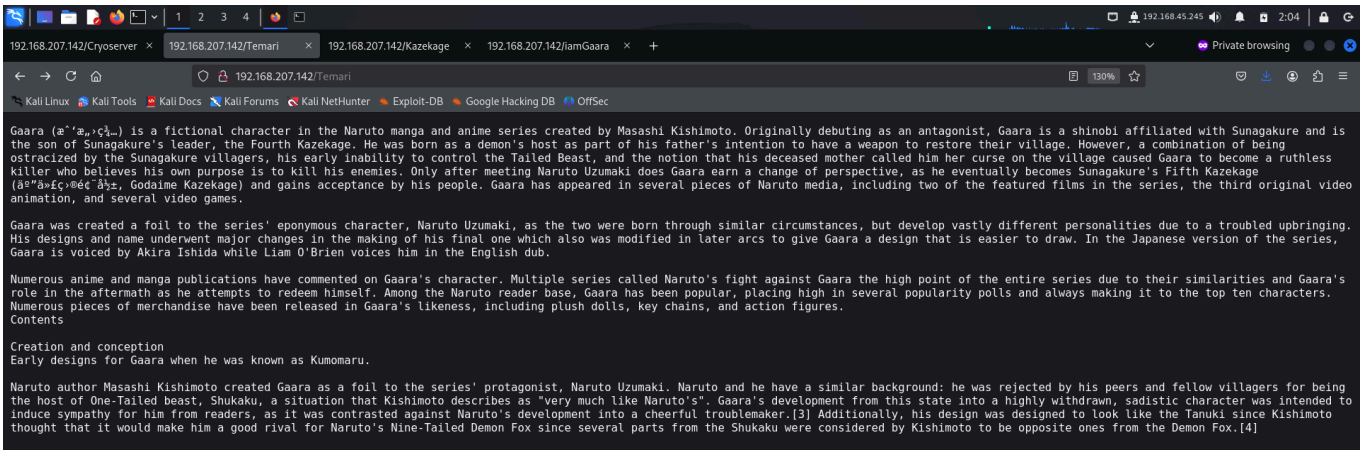
The browser had nothing interesting so I used **ffuf** to perform web directory fuzzing. Through this, I discovered a new directory.

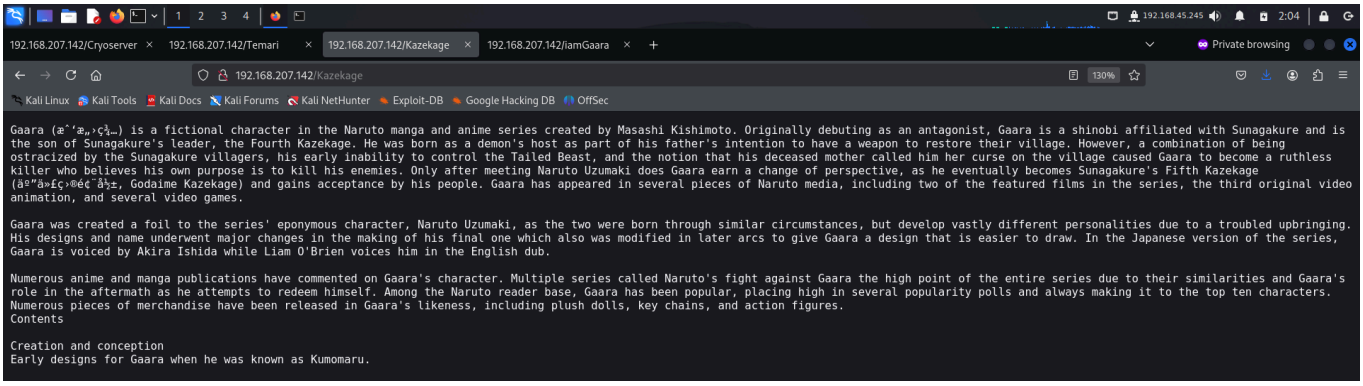


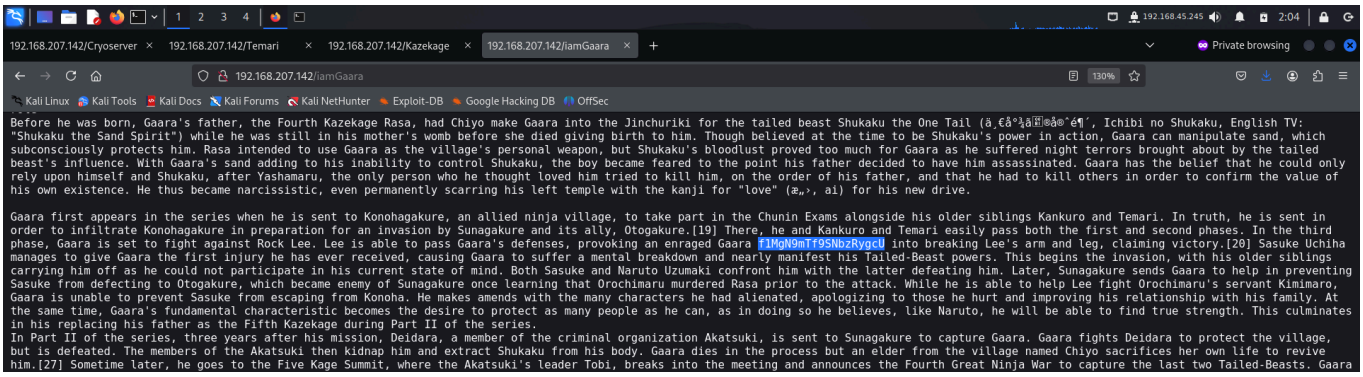
Upon accessing the directory, I found 3 new paths.



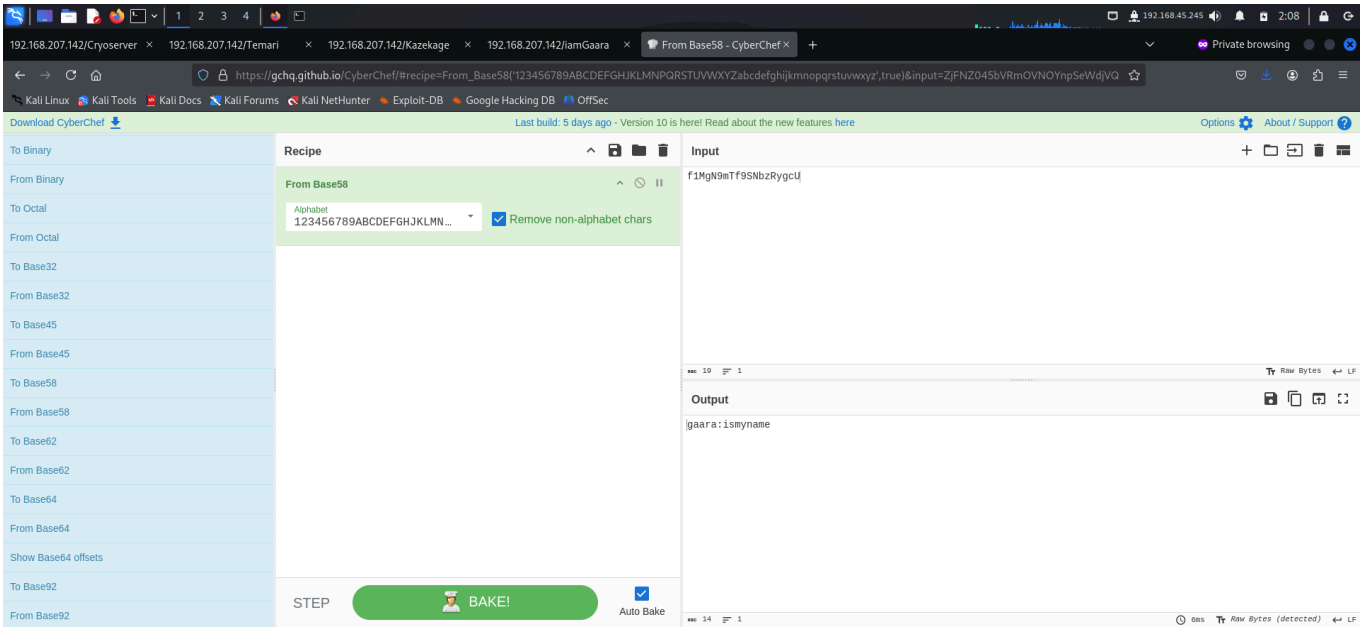
I accessed them one after the other but they all had the same extract. Upon closer inspection, I found that `/iamGaara` had an encoded string.



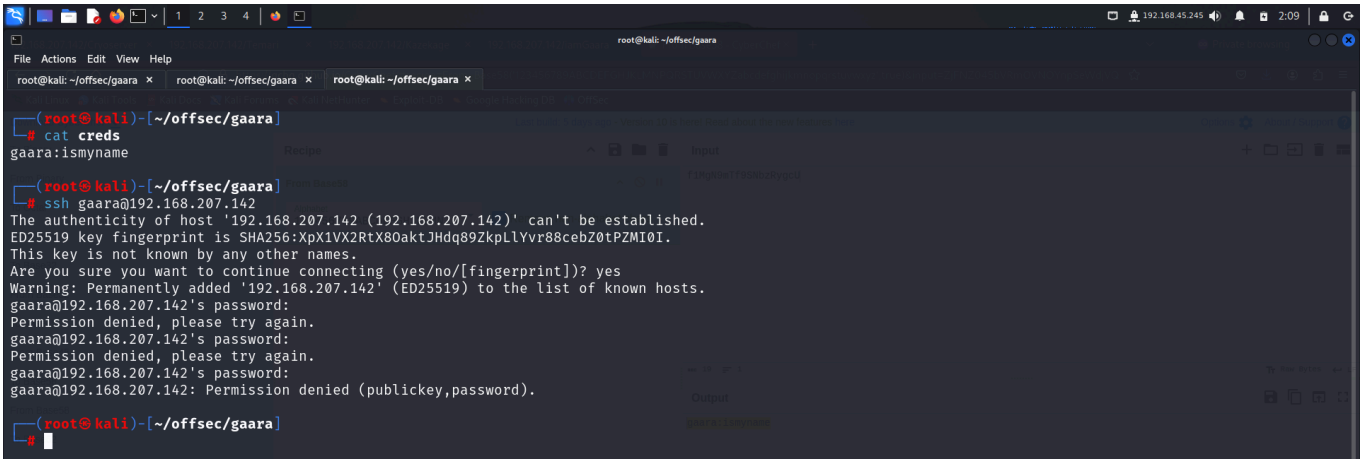




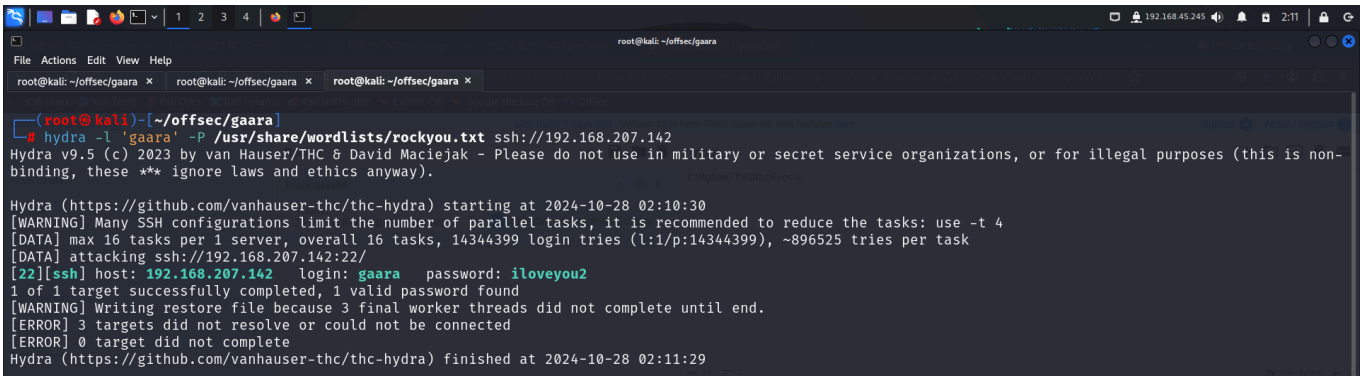
To decode this, I visited **cyberchef** and tried multiple encoding formats, out of which **base58** worked.



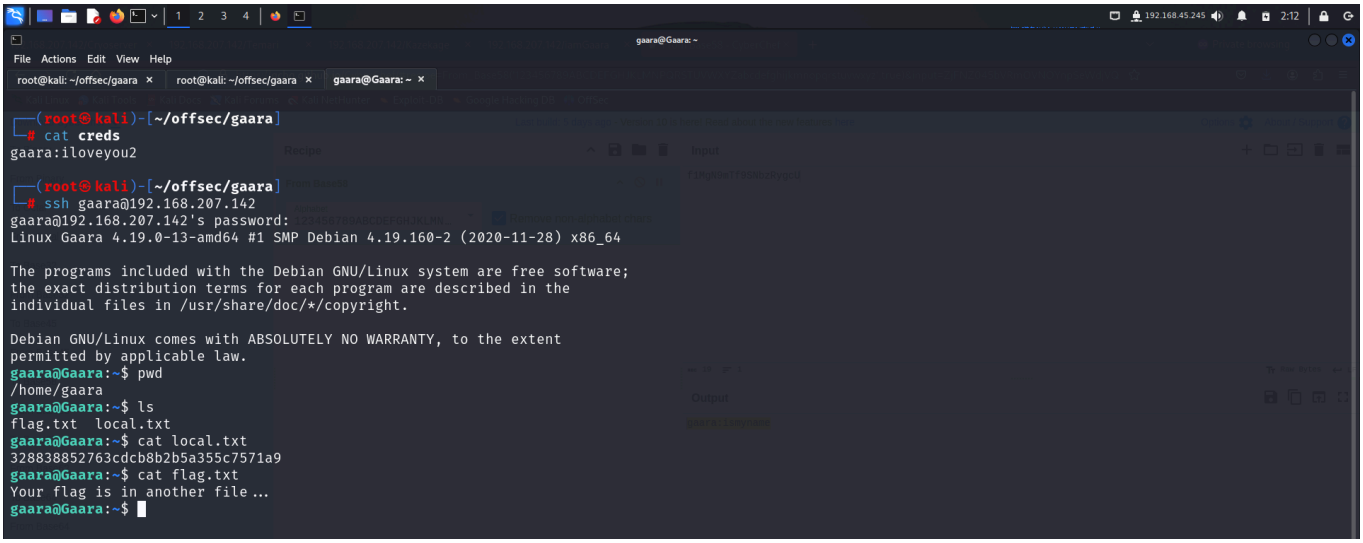
This looked like a pair of credentials so I tried logging in through **ssh** using them. However, this turned out to be a rabbit hole. The password was incorrect.



Since I had no other leads, I tried brute-forcing the password of *gaara* using **hydra** and **rockyou.txt** wordlist.

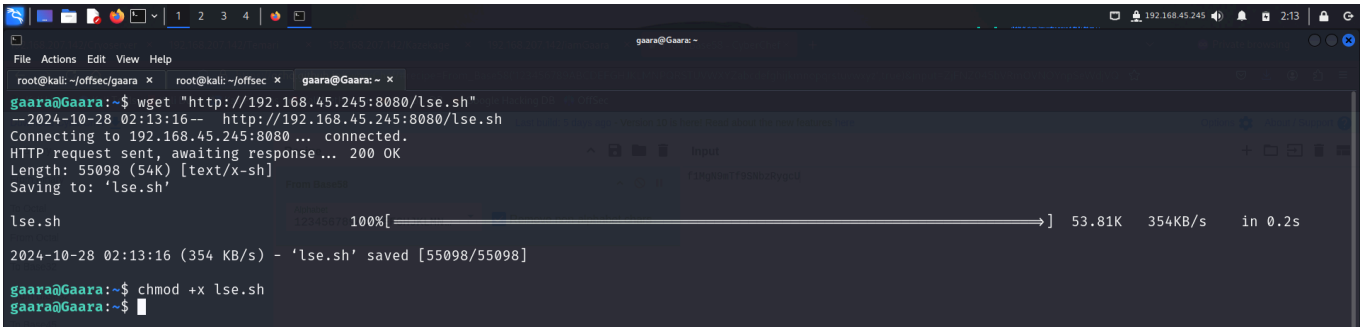


After finding the valid password, I logged in through **ssh** and found the flag in my home directory.

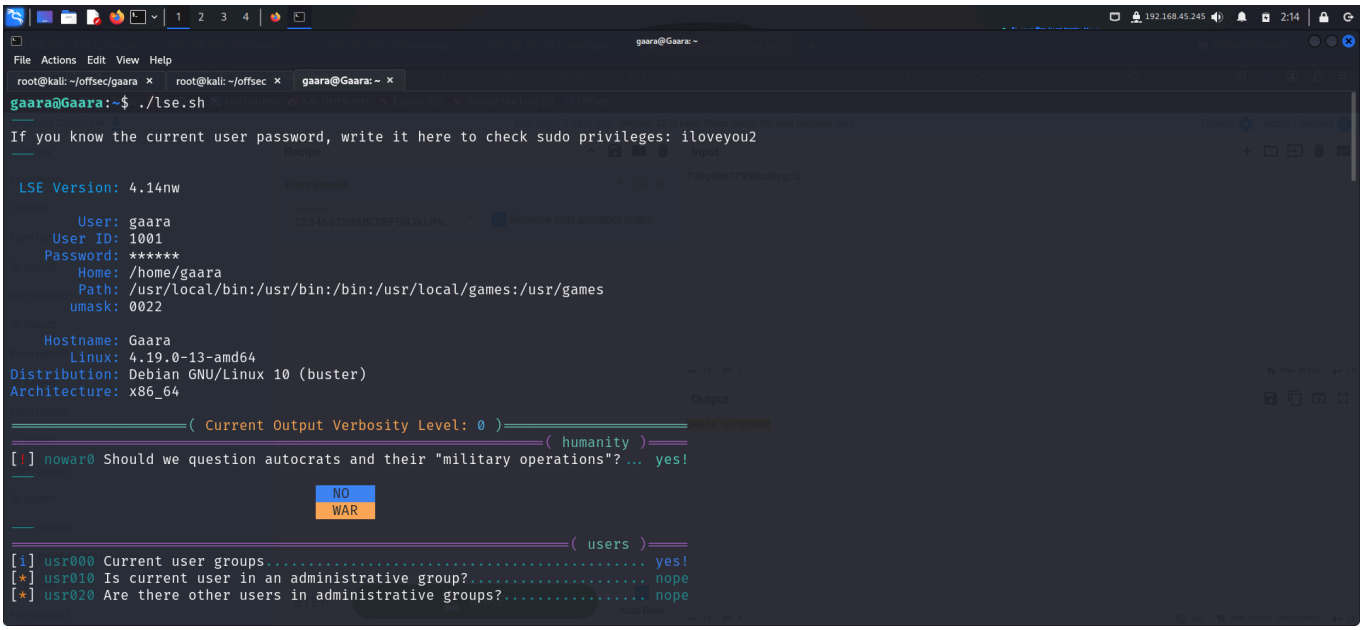


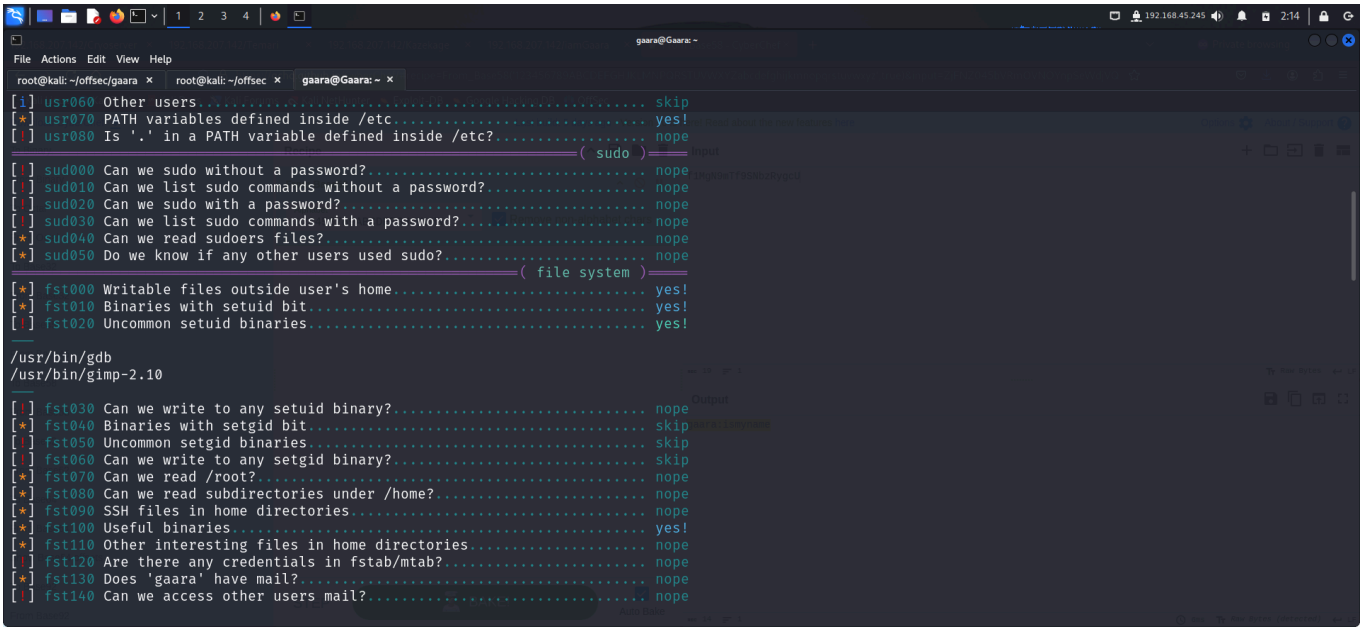
PRIVILEGE ESCALATION

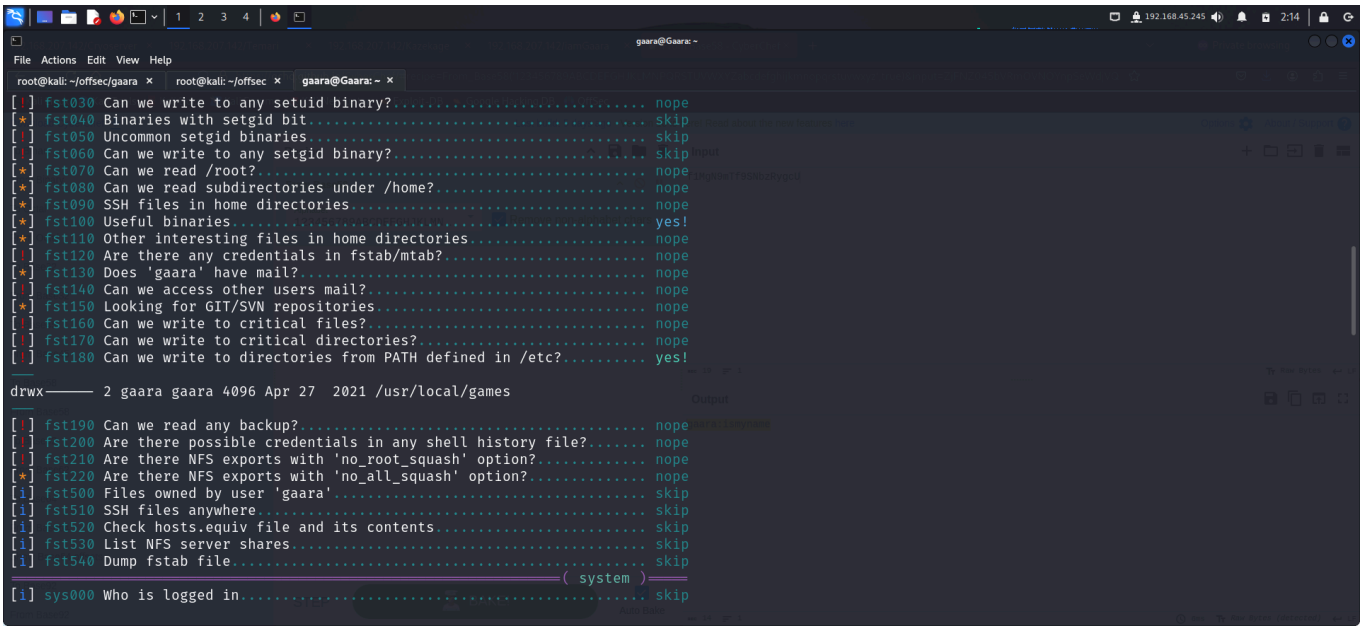
After getting initial access, I downloaded **linux smart enumeration** script on the target to look for misconfigurations that could lead to privilege escalation.



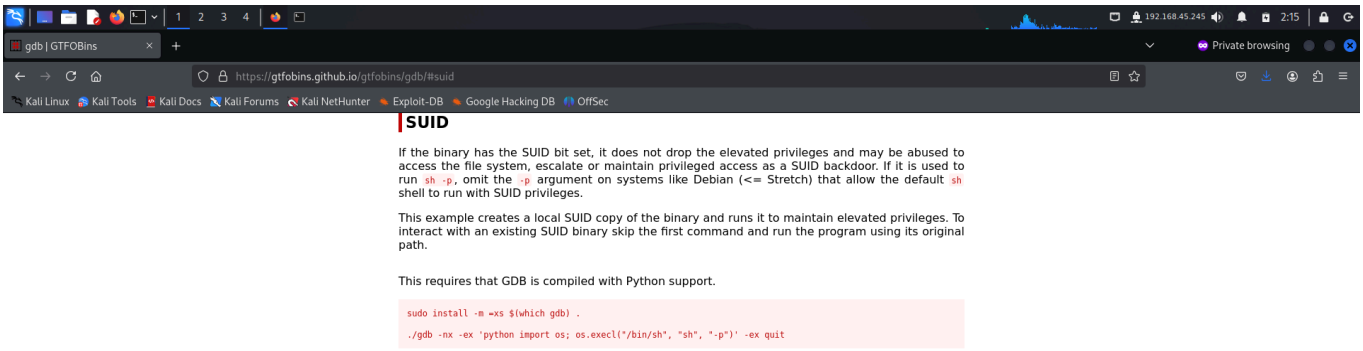
I ran the script and found an interesting set of binaries with **setuid** bit. I also found that I could write into the `/usr/local/games` directory.

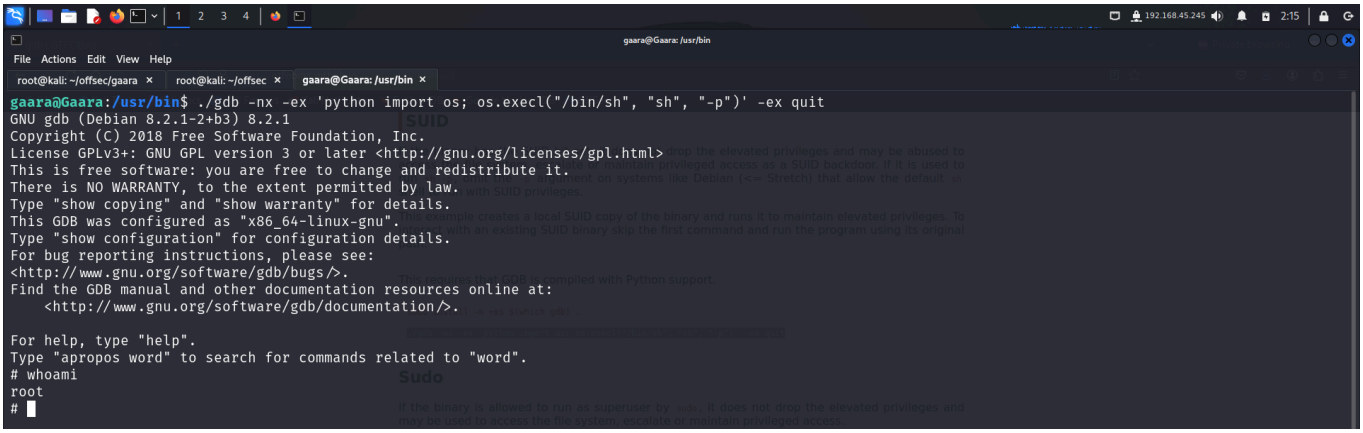

gaara@Gaara: ~
File Actions Edit View Help
root@kali: ~/offsec/gaara
root@kali: ~/offsec
gaara@Gaara: ~
gaara@Gaara:~$ ./lse.sh
If you know the current user password, write it here to check sudo privileges: iloveyou2
LSE Version: 4.14nw
User: gaara
User ID: 1001
Password: ******
Home: /home/gaara
Path: /usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
umask: 0022
Hostname: Gaara
Linux: 4.19.0-13-amd64
Distribution: Debian GNU/Linux 10 (buster)
Architecture: x86_64
(Current Output Verbosity Level: 0)
(humanity)
[!] nowar0 Should we question autocrats and their "military operations"? ... yes!
NO
WAR
(users)
[i] usr000 Current user groups ... yes!
[*] usr010 Is current user in an administrative group? ... nope
[*] usr020 Are there other users in administrative groups? ... nope


gaara@Gaara: ~
File Actions Edit View Help
root@kali: ~/offsec/gaara
root@kali: ~/offsec
gaara@Gaara: ~
[i] usr060 Other users ... skip
[*] usr070 PATH variables defined inside /etc ... yes!
[!] usr080 Is '.' in a PATH variable defined inside /etc? ... nope
(sudo)
[!] sud000 Can we sudo without a password? ... nope
[!] sud010 Can we list sudo commands without a password? ... nope
[!] sud020 Can we sudo with a password? ... nope
[!] sud030 Can we list sudo commands with a password? ... nope
[*] sud040 Can we read sudoers files? ... nope
[*] sud050 Do we know if any other users used sudo? ... nope
(file system)
[*] fst000 Writable files outside user's home ... yes!
[*] fst010 Binaries with setuid bit ... yes!
[!] fst020 Uncommon setuid binaries ... yes!
/usr/bin/gdb
/usr/bin/gimp-2.10
[!] fst030 Can we write to any setuid binary? ... nope
[*] fst040 Binaries with setgid bit ... skip
[!] fst050 Uncommon setgid binaries ... skip
[!] fst060 Can we write to any setgid binary? ... skip
[*] fst070 Can we read /root? ... nope
[*] fst080 Can we read subdirectories under /home? ... nope
[*] fst090 SSH files in home directories ... nope
[*] fst100 Useful binaries ... yes!
[*] fst110 Other interesting files in home directories ... nope
[!] fst120 Are there any credentials in fstab/mtab? ... nope
[*] fst130 Does 'gaara' have mail? ... nope
[!] fst140 Can we access other users mail? ... nope

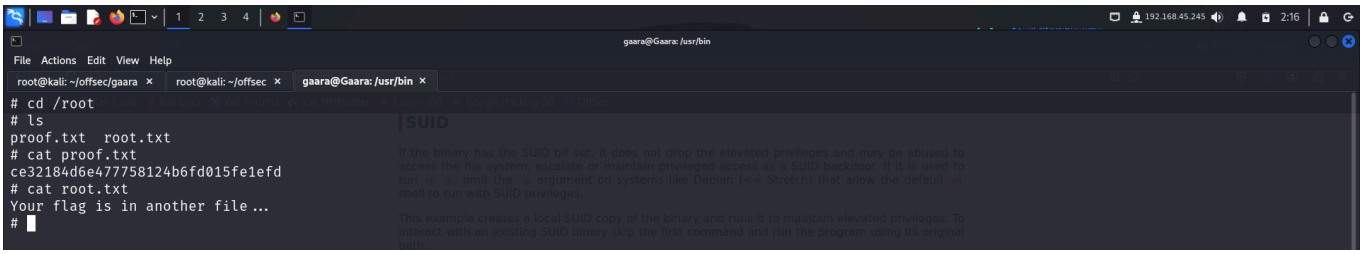


I started off with the **setuid** bit binaries and searched **gtfobins** for ways to exploit them. I found a way to escalate privilege with **gdb** so I repeated the steps mentioned on the site.





After getting **root** access, I captured the final flag from the `/root` directory.



CONCLUSION

Here's a summary of how I pwned the machine:

- I performed a directory brute-force attack to discover hidden directories.
- This directory revealed more paths that I could analyze.
- One of the page contained a base58 encoded string which contained a username and password.
- After failing to log in with the given password, I brute-forced the correct password using **hydra**.
- **hydra** identified the valid set of credentials and I used them to log in as **gaara**.
- I then captured the first flag in my home directory.
- I discovered uncommon setuid bits on 2 binaries and visited **gtfobins** to look for ways to exploit them,
- On **gtfobins**, I found a way to escalate my privilege using **gdb**
- I used the provided method to become a **root** user and captured the final flag from the `/root` directory.

That's it from my side!
Until next time :)

ALEX BECKFORD
@StayBrutalAlex

This is going to bug me all night but who would win in a fight.

Sandman V.S. Gaara