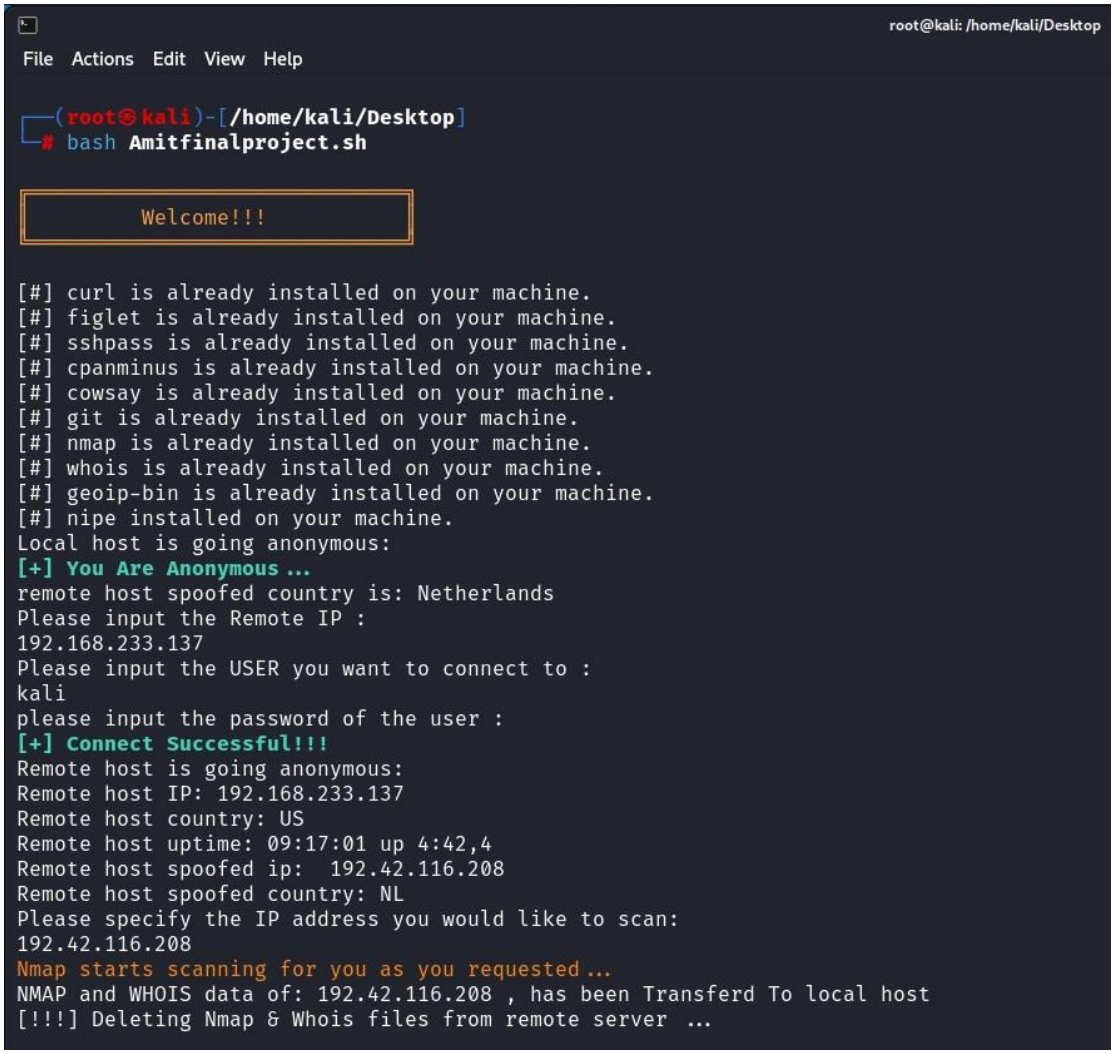
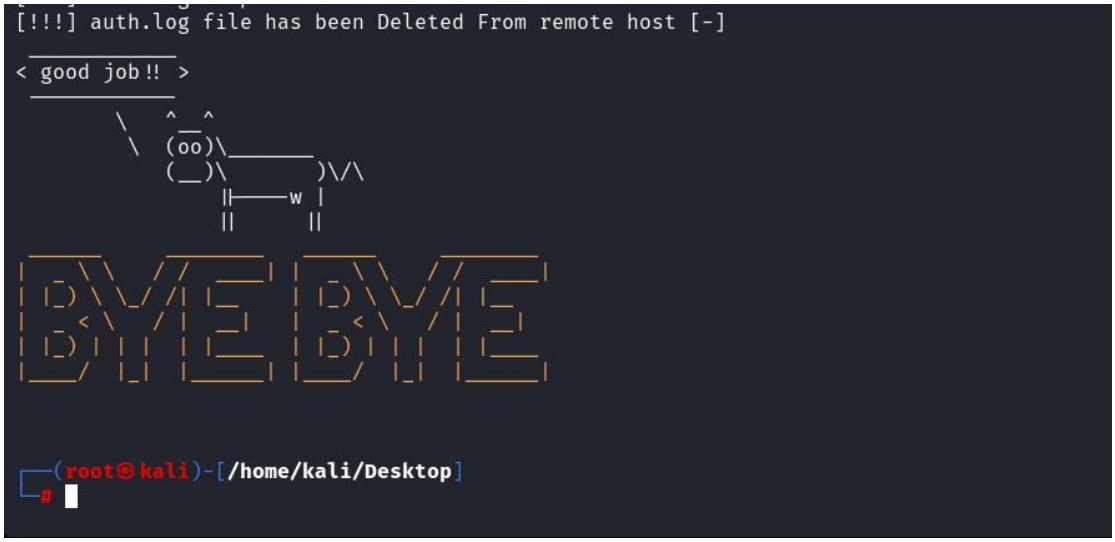
PROJECT REMOTE CONTROL- AMIT PERSKY

This project entails setting up a system that initiates with the installation of required applications, ensuring avoidance of repeated installations. It performs an anonymity check of the network connection, alerting if non-anonymous, and revealing the spoofed country name if anonymous. It also accepts user-specified scan targets. Furthermore, the system can establish a remote SSH connection to retrieve server details and execute commands such as Whois and open port scans. Finally, it saves the gathered data into local files and maintains a log for auditing data collection activities.

Output of my script:





Explanation of output according to the project structure:

1. Installations and Anonymity Check

1.1 Install the needed applications

1.2 If the applications are already installed, don’t install them again:

As we can see in the output some are installed, and some are not.

```
[#] curl is already installed on your machine.
[#] figlet is already installed on your machine.
[#] sshpass is already installed on your machine.
[#] cpanminus is already installed on your machine.
[#] cowsay is already installed on your machine.
[#] git is already installed on your machine.
[#] nmap is already installed on your machine.
[#] whois is already installed on your machine.
[#] geoip-bin is already installed on your machine.
[#] nipe installed on your machine.
```

1.3 Check if the network connection is anonymous; if not, alert the user and exit:

1.4 If the network connection is anonymous, display the spoofed country name.

As we can see in the output we get a "green light" for being anonymous, and the name of the country we are supposedly currently in.

```
[+] You Are Anonymous ...
remote host spoofed country is: Netherlands
```

1.5 Allow the user to specify the address to scan via remote server; save into a variable.

As we can see in the output, after we enter the variables that are automatically saved to the script, the connection (in green) is created from which we will proceed to the next part.

```
Please input the Remote IP :
192.168.233.137
Please input the USER you want to connect to :
kali
please input the password of the user :
[+] Connect Successful!!!
```

2. Automatically Connect and Execute Commands on the Remote Server via SSH

2.1 Display the details of the remote server (country, IP, and Uptime)

2.2 Get the remote server to check the Whois of the given address

As we can see in the output the connection created results in us getting details about the server we connected to, and we give it an address to check on.

```
Remote host is going anonymous:
Remote host IP: 192.168.233.137
Remote host country: US
Remote host uptime: 09:17:01 up 4:42,4
Remote host spoofed ip:  192.42.116.208
Remote host spoofed country: NL
```

2.3 Get the remote server to scan for open ports on the given address:

As we can see in the output an address is requested for scanning, an address is given and after that NMAP goes to scans.

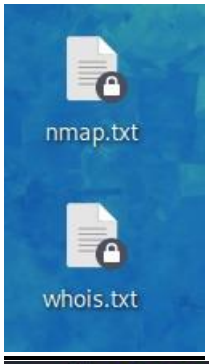
```
Please specify the IP address you would like to scan:
192.42.116.208
Nmap starts scanning for you as you requested ...
```

3. Results

3.1 Save the Whois and Nmap data into files on the local computer:

As we can see in the output the results are saved on our machine, on the desktop, and those files are deleted from the server.

```
NMAP and WHOIS data of: 192.42.116.208 , has been Transferd To local host
[!!!] Deleting Nmap & Whois files from remote server ...
```




```
┌──(kali㉿kali)-[~/Desktop]
└─$ ls
Amitfinalproject.sh  Copy  nipe  nmap.txt  Steghide  whois.txt
```

3.2 Create a log and audit your data collecting:

As we can see in the output we are eliminating traces..

```
[!!!] auth.log file has been Deleted From remote host [-]
```