REMOTE CONTROL PROJECT
by Hairkal Juhair
Hairkaljuhair93@gmail.com. Done on 12 july 2022

Requirements:

1. Install the relevant applications
2. Check if the connection to the vps is anonymous
3. Connect to a VPS and execute scans
   a. Option 1 - interactive, ask user for inputs
   b. Option 2 - use arguments $1 $2
   c. Option 3 - Inside the script
   d. VPS - digital ocean, aws, Ubuntu, kali
   i. https://m.do.co/c/2099826f2433
   ii. You can create a new droplet (server), Access > Launch Droplet Console
4. Use comments in your code to explain your actions
5. If you are using code from the internet, add credit and links
6. Document your code and steps , proving that the functions work
7. Submit the script (.sh) and the documentations (in pdf)
   a. project: Remote Control <student name><cfc180422>
   b. james@cfcsea.com
8. Deadline: 14th July 2022, 11:59pm SGT

My kali – [redacted].161.128
My vps – [redacted].4.52

I am showing you ssh **BEFORE** running script. Nipe is disabled and connection is established from Kali with original ip address to vps.

Exit and run Script.

My script has a total of 6 Steps. I will give you a breakdown on the step-by-step as we go.

Run script. →

Step 1.
I am calling my script, NRproject.sh located in /home/hairkal/scripting.
So I will be running my script in the scripting folder.
The 1st function is called 'iam'. It shows usersname and original ip before proceeding into going anonymous.

```bash
#!/bin/bash

# Summary.
# 1. establish your IP
# 2. start nipe
# 3. check if you are anonymous.
# 4. echo your anonynmous IP and location
# 5. Run nmap/whois.
# 6. echo the answer and save result

# Step 1. establish your IP

function iam ()
{
    echo "Let's begin. Establishing user.."
    whoami
    hostname -I
    echo "Start Nipe to go anonymous!"
```

```
┌──(hairkal㉿kali)-[~/scripting]
└─$ bash NRproject.sh
Let's begin. Establishing user..
hairkal
[redacted].161.128
Start Nipe to go anonymous!

┌──(hairkal㉿kali)-[~/scripting]
└─$
```

Step 2.
After establishing the user, the script will then proceed to activate nipe. My nipe is located in /home/hairkal/nipe.
Therefore, I will need to specify the location in the script and activated nipe.
I also added on the variable $active to be echoed to user. So that user is being prompt that nipe is activated.

```bash
# Step 2. start nipe

function startnipe ()
{
    echo "Starting nipe.. Hang on.."
    cd /home/hairkal/nipe
    sudo perl nipe.pl start
    active=$(sudo perl nipe.pl status | grep activated | awk '{print $3}')
    echo "Nipe is $active"
}
```

```
┌──(hairkal㉿kali)-[~/scripting]
└─$ bash NRproject.sh
Starting nipe.. Hang on..
Nipe is activated.

┌──(hairkal㉿kali)-[~/scripting]
└─$
```

Step 3 & 4.
We now know that nipe is activated. But just to be sure , I have added in our anonymous checker using if conditions exercise that we previously did into use.
I added the variable $anon which calls out our 'random/anonymous' ip to our user to confirm that we are indeed anonymous and echo new anonymous ip location

```
# Step 3 & 4. User wants to ensure that user is Anonymous. Echo location

function anonchecker ()
{
    echo "Checking if you are Anonymous.."
    cd /home/hairkal/nipe
    anon=$(sudo perl nipe.pl status | grep Ip | awk '{print $3}')
    echo "You are now $anon"
    location=$(geoiplookup $anon | awk -F: '{print $2}')
    echo "located in: $location"
    stat_check=$(sudo perl nipe.pl status | grep -w activated)


if [ ! -z "$stat_check" ]
    then
        echo "You are Anonymous.. Let's go!!"
    else
        echo "You are Expose!! Restart before you proceed.. "
fi
}
```

```
┌──(hairkal㉿kali)-[~/scripting]
└─$ bash NRproject.sh
Checking if you are Anonymous..
[sudo] password for hairkal:
You are now 193.218.118.95
located in:  UA, Ukraine
You are Anonymous.. Let's go!!

┌──(hairkal㉿kali)-[~/scripting]
└─$
```

Step 5 & 6

I will now be going into ssh after confirming that I am indeed anonymous. This function allows me to access the vps and proves that you are able to login as root for the remote server by using the whoami command.. And once I am in, it will run the case options continuously without interruption (ssh -T)

*I also made use of my private and public key to authenticate my login without prompting password.

ssh – provides a default pseudo terminal / needs user interaction
ssh -t – Force pseudo-terminal allocation. This can be used to execute arbitrary screen-based programs on a remote machine, which can be very seful, e.g. when implementing menu services. Multiple -t options force tty allocation, even if ssh has no local tty.
ssh -T – Disable pseudo-terminal allocation.

So, I tried multiple variations into how this could work.

First solution:

1. Creating a script in the vps and run it on the vps itself. It works ☺
2. The script is now stored in the vps. So I tried calling the script and running it from my host terminal. It works ☺

Second solution:

1. Knowing I could call the script to run on the vps from my host. I decided to create the same script from my host. Scp the sh.file to the vps and execute it from my main script. It works ☺

Third solution: (it works in one script, so I went with this ☺)

1. Call out the case option before entering vps.
2. Once user has input the selection, store that variable.
3. Enter vps with said variable and run the script. Resulting, all being run on one whole script and echo and save result.

So here's how I did it.

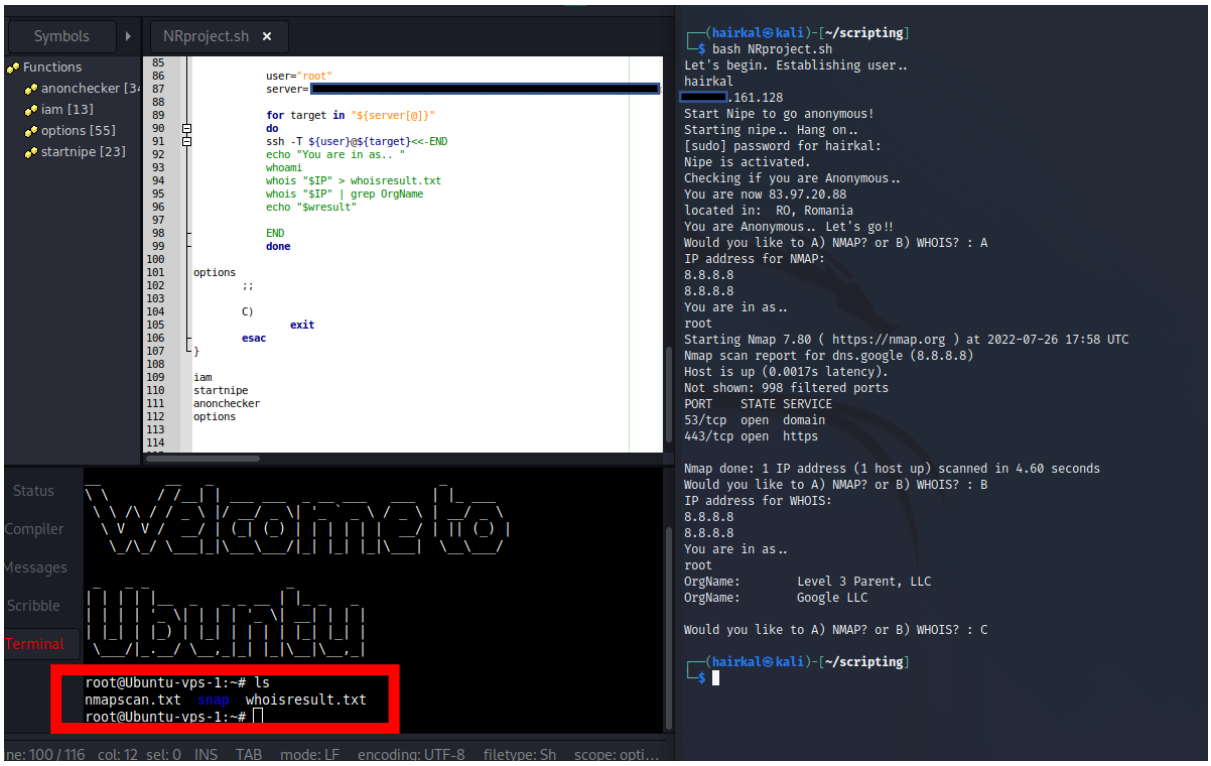
```bash
# Step 5 & 6. Run nmap/whois. echo the answer

function options ()
{
    read -p "Would you like to A) NMAP? or B) WHOIS? : " choose
case $choose in
        A)
            echo "IP address for NMAP: "
            read IP
            echo "$IP"


            user="root"
            server=[redacted]

            for target in "${server[@]}"
            do
            ssh -T ${user}@${target}<<-END
            echo "You are in as.. "
            whoami
            nmap "$IP" -oN nmapscan.txt

            END
            done

options
        ;;

        B)
            echo "IP address for WHOIS: "
            read IP
            echo "$IP"

            user="root"
            server=[redacted]

            for target in "${server[@]}"
            do
            ssh -T ${user}@${target}<<-END
            echo "You are in as.. "
            whoami
            whois "$IP" > whoisresult.txt
            whois "$IP" | grep OrgName
            echo "$wresult"

            END
            done

options
        ;;

        C)
                exit
        esac
}
```

```
┌──(hairkal㉿kali)-[~/scripting]
└─$ bash NRproject.sh
Would you like to A) NMAP? or B) WHOIS? : A
IP address for NMAP:
8.8.8.8
8.8.8.8
You are in as..
root
Starting Nmap 7.80 ( https://nmap.org ) at 2022-07-26 17:52 UTC
Nmap scan report for dns.google (8.8.8.8)
Host is up (0.0016s latency).
Not shown: 998 filtered ports
PORT    STATE SERVICE
53/tcp  open  domain
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 4.51 seconds
Would you like to A) NMAP? or B) WHOIS? : B
IP address for WHOIS:
8.8.8.8
8.8.8.8
You are in as..
root
OrgName:        Level 3 Parent, LLC
OrgName:        Google LLC

Would you like to A) NMAP? or B) WHOIS? : C

┌──(hairkal㉿kali)-[~/scripting]
└─$
```

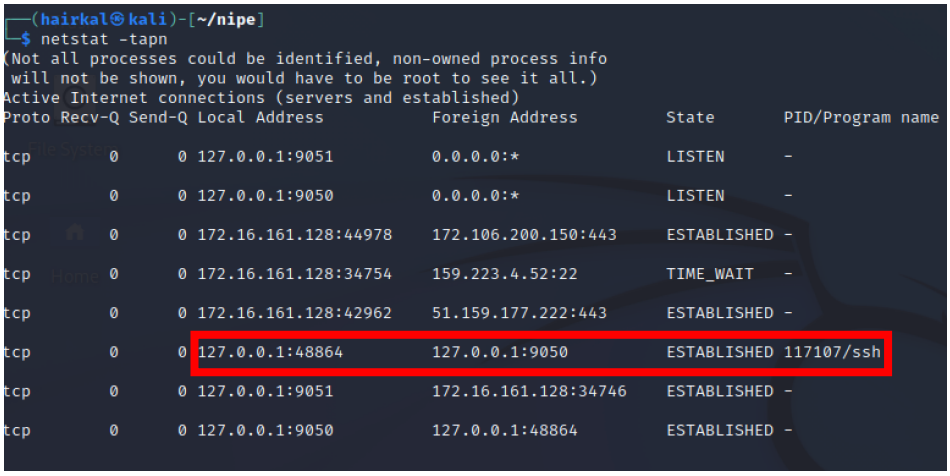
```
# Step 5 & 6. Run nmap/whois. echo the answer

function options ()
{
    read -p "Would you like to A) NMAP? or B) WHOIS? : " choose
case $choose in
        A)
            echo "IP address for NMAP: "
            read IP
            echo "$IP"


            user="root"
            server=[redacted]

            for target in "${server[@]}"
            do
            ssh -T ${user}@${target}<<-END
            echo "You are in as.. "
            whoami
            nmap "$IP" -oN nmapscan.txt

            END
            done

options
        ;;

        B)
            echo "IP address for WHOIS: "
            read IP
            echo "$IP"

            user="root"
            server=[redacted]

            for target in "${server[@]}"
            do
            ssh -T ${user}@${target}<<-END
            echo "You are in as.. "
            whoami
            whois "$IP" > whoisresult.txt
            whois "$IP" | grep OrgName
            echo "$wresult"

            END
            done

options
        ;;

        C)
                exit
        esac
}
```

That basically summarize all of the step. Put it all together and run. →

Running Result: script is running on kali. Result stored in txt in vps.



Anonymous ssh result ☺



Credits :

Torify ssh: https://youtu.be/B9kogZO0omI

Ssh private / public key : https://www.cyberciti.biz/faq/how-to-set-up-ssh-keys-on-linux-unix/

Executing scripts while in ssh: https://youtu.be/o9H303Z9ukc

NRproject.sh (main script)

```
#!/bin/bash

# Summary.
# 1. establish your IP
# 2. start nipe
# 3. check if you are anonymous.
# 4. echo your anonynmous IP and location
# 5. Run nmap/whois.
# 6. echo the answer and save result

# Step 1. establish your IP

function iam ()
{
		echo "Let's begin. Establishing user.."
		whoami
		hostname -I
		echo "Start Nipe to go anonymous!"
}

# Step 2. start nipe

function startnipe ()
{
		echo "Starting nipe.. Hang on.."
		cd /home/hairkal/nipe
		sudo perl nipe.pl start
		active=$(sudo perl nipe.pl status | grep activated | awk '{print $3}')
		echo "Nipe is $active"
}

# Step 3 & 4. User wants to ensure that user is Anonymous. Echo location

function anonchecker ()
{
		echo "Checking if you are Anonymous.."
		cd /home/hairkal/nipe
		anon=$(sudo perl nipe.pl status | grep Ip | awk '{print $3}')
		echo "You are now $anon"
		location=$(geoiplookup $anon | awk -F: '{print $2}')
		echo "located in: $location"
		stat_check=$(sudo perl nipe.pl status | grep -w activated)


if [ ! -z "$stat_check" ]
		then
				echo "You are Anonymous.. Let's go!!"
		else
				echo "You are Expose!! Restart before you proceed.. "
fi
}

# Step 5 & 6. Run nmap/whois. echo the answer

function options ()
{
		read -p "Would you like to A) NMAP? or B) WHOIS? : " choose
case $choose in
	A)
						echo "IP address for NMAP: "
						read IP
						echo "$IP"


						user="root"
						server="[redacted]"

						for target in "${server[@]}"
						do
						ssh -T ${user}@${target}<<-END
						echo "You are in as.. "
			whoami
						nmap "$IP" -oN nmapscan.txt

						END
						done

options
	;;

	B)
						echo "IP address for WHOIS: "
		read IP
						echo "$IP"

		user="root"
						server="[redacted]"
```

```
                                        for target in "${server[@]}"
                                        do
                                        ssh -T ${user}@${target}<<-END
                                        echo "You are in as.. "
                    whoami
                                        whois "$IP" > whoisresult.txt
        whois "$IP" | grep OrgName
        echo "$wresult"

                                        END
                                        done

options
    ;;

    C)
        exit
    esac
}

iam
startnipe
anonchecker
options
```