



**Centre For  
Cybersecurity**

# **Network Control (Eddie) CFC2407**

# Objective

Creating a script to automate processes to connect to the remote server via ssh protocol anonymously, performing scans such as Nmap, Masscan, and Whois, and saving it onto the local host.

## 1) Update and upgrade the Kali Linux Operating System

Upgrad the Linux operating system to enable to install the relevant tools using the command:

**`sudo apt-get upgrade && sudo apt-get upgrade -y`**

```
└─$ sudo apt-get update && sudo apt-get upgrade -y
Hit:1 http://mirror.aktkn.sg/kali kali-rolling InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer require
d:
  libexporter-tiny-perl libhttp-server-simple-perl liblist-moreutils-perl
  liblist-moreutils-xs-perl libpython3.9-minimal libpython3.9-stdlib
  libwacom-bin python3-dataclasses-json python3-limiter
  python3-marshmallow-enum python3-mypy-extensions python3-responses
  python3-spyse python3-token-bucket python3-typing-inspect python3.9
  python3.9-minimal
Use 'sudo apt autoremove' to remove them.
The following packages have been kept back:
  at-spi2-core cgpt cherrytree cpp faraday g++ gcc gir1.2-atspi-2.0
  gstreamer1.0-plugins-bad ipython3 kismet-capture-linux-bluetooth
  kismet-capture-linux-wifi kismet-capture-nrf-51822
  kismet-capture-nrf-52840 kismet-capture-nrf-mousejack
  kismet-capture-nxp-kw41z kismet-capture-rz-killerbee
  kismet-capture-ti-cc-2531 kismet-capture-ti-cc-2540
  kismet-capture-ubertooth-one kismet-core libatk1.0-0 libatspi2.0-0
  libavcodec59 libavfilter8 libavformat59 libavutil56 libavutil57
  libcephfs2 libgdal31 libgstreamer-plugins-bad1.0-0
  libmagickcore-6.q16-6-extra libmm-glib0 libpoppler-glib8 libpostproc56
  librados2 libruby3.0 libswresample3 libswresample4 libswscale6 libtbb12
  libtbbmalloc2 libtiff5 libwacom-bin linux-image-amd64 modemmanager
  python3-flask-limiter python3-fonttools python3-ipython
  python3-jwschema python3-limits python3-redis python3-scipy ruby-ffi
  ruby-nokogiri ruby-oj ruby-sqlite3 ruby-unf-ext ruby-yajl ruby3.0-dev
  ruby3.0-dev
  ruby3.0-dev
```

## 2) Install and execute Geany onto Kali Linux

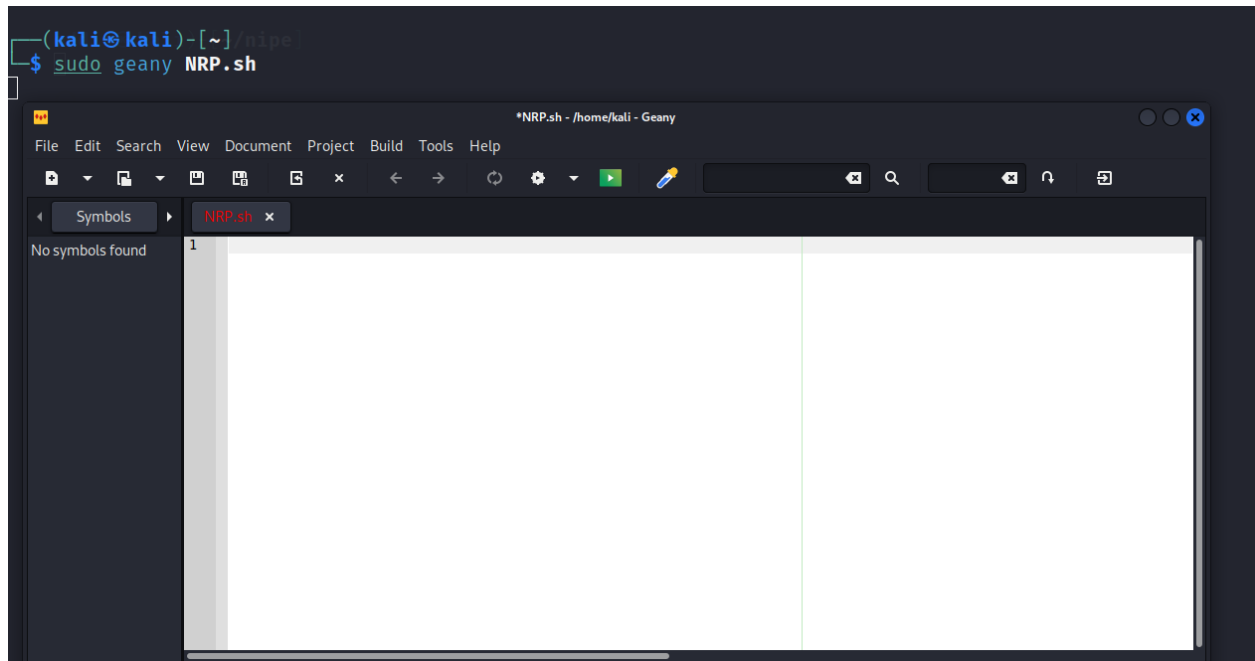
Note: geany is a tool to write, view, and edit files/scripts by opening them from other programs such as file manager.

A). Install using the command: **`sudo apt-get install geany`**

```
(kali㉿kali)-[~]
└─$ sudo apt-get install geany
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
geany is already the newest version (1.38-1+b1).
The following packages were automatically installed and are no longer required:
  libev4 libexporter-tiny-perl libfmt8 libhttp-server-simple-perl liblmbase25 liblerc3
  liblist-moreutils-perl liblist-moreutils-xs-perl libopenexr25 libplacebo192 libpoppler118
  libpython3.9-minimal libpython3.9-stdlib libsvtavifenc0 libwebsockets16 python3-dataclasses-json
  python3-limiter python3-marshmallow-enum python3-mypy-extensions python3-responses python3-spyse
  python3-token-bucket python3-typing-inspect python3.9 python3.9-minimal
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
```

B). Execute with the name of the file “NRPeddie.sh” using the command:

**sudo geany NRPeddie.sh**



3) Function ‘inst’ in the script is to install and update all OS (Operating System) and tools.

Note: In the script “function”, it allows you to store a set of commands into a block of codes that can be repeatedly called anytime.

```
1  #!/bin/bash
2
3  #Functions
4  #Basic function format
5
6  function <Variable name>()
7  {
8
9
10     <actions>
11
12 }
13
14
```

A). Upgrade the OS using the command: `sudo apt-get dist-upgrade`, to enable the relevant tools to be compatible with the OS.

```
(eddie@kali)-[~]
$ sudo apt-get dist-upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  libev4 libexporter-tiny-perl libfmt8 libhttp-server-simple-perl liblmbase25 liblerc3 l
  libplacebo192 libpoppler118 libpython3.9-minimal libpython3.9-stdlib libsvtav1enc0 libw
  python3-marshmallow-enum python3-mypy-extensions python3-responses python3-spyse python
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  aspnetcore-runtime-6.0 aspnetcore-targeting-pack-6.0 cpp-12 dotnet-apphost-pack-6.0 dot
  dotnet-sdk-6.0 dotnet-targeting-pack-6.0 g++-12 gcc-12 inetutils-telnet libboost-dev li
  libopenblas-pthread-dev libopenblas0 libopenblas0-pthread libopenexr-3-1-30 libplacebo2
  libsvtav1enc1 libtbbbind-2-5 libwebsockets17 libxsimd-dev linux-image-5.18.0-kali7-amd6
  python3-deprecated python3-gast python3-json-pointer python3-libevdev python3-pythran p
  python3-webcolors python3-wrapt ruby-sdbm tftp-hpa
The following packages have been kept back:
  libavutil56 libswresample3
The following packages will be upgraded:
```

B). Download and install Nipe on Kali Linux. Follow the steps below.

Note: Nipe is an engine that aims to make the Tor network your default network gateway. Tor is an open-source software that enables anonymous communication.

i) Download Nipe packets and store them in a Nipe folder using the command:

`git clone https://github.com/htrgouvea/nipe && cd nipe`

```
(eddie@kali)-[~]
$ git clone https://github.com/htrgouvea/nipe && cd nipe
Cloning into 'nipe'...
remote: Enumerating objects: 1660, done.
remote: Counting objects: 100% (131/131), done.
remote: Compressing objects: 100% (87/87), done.
remote: Total 1660 (delta 50), reused 90 (delta 29), pack-reused 1529
Receiving objects: 100% (1660/1660), 253.69 KiB | 12.08 MiB/s, done.
Resolving deltas: 100% (863/863), done.

(eddie@kali)-[~/nipe]
$
```

ii) Install libs and dependencies packets to install Nipe using the command: `sudo cpan install`

`Try::Tiny Config::Simple JSON`

```
(eddie@kali)-[~/nipe]
$ sudo cpan install Try::Tiny Config::Simple JSON
Loading internal logger. Log::Log4perl recommended for better logging
Reading '/root/.cpan/Metadata'
  Database was generated on Thu, 29 Sep 2022 16:55:45 GMT
Try::Tiny is up to date (0.31).
Running install for module 'Config::Simple'
Fetching with LWP:
http://www.cpan.org/authors/id/S/SH/SHERZODR/Config-Simple-4.58.tar.gz
Fetching with LWP:
http://www.cpan.org/authors/id/S/SH/SHERZODR/CHECKSUMS
Checksum for /root/.cpan/sources/authors/id/S/SH/SHERZODR/Config-Simple-4.58.tar.gz ok
'YAML' not installed, will not store persistent state
Configuring S/SH/SHERZODR/Config-Simple-4.58.tar.gz with Makefile.PL
Checking if your kit is complete...
Looks good
Generating a Unix-style Makefile
Writing Makefile for Config::Simple
Writing MYMETA.yml and MYMETA.json
SHERZODR/Config-Simple-4.58.tar.gz
/usr/bin/perl Makefile.PL INSTALLDIRS=site -- OK
Running make for S/SH/SHERZODR/Config-Simple-4.58.tar.gz
cp Simple.pm blib/lib/Config/Simple.pm
AutoSplitting blib/lib/Config/Simple.pm (blib/lib/auto/Config/Simple)
Manifying 1 pod document
```

iii) Install Nipe using the command: `sudo perl nipe.pl install`

```
(eddie@kali)~[/nipe]
$ sudo perl nipe.pl install
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
iptables is already the newest version (1.8.8-1).
iptables set to manually installed.
The following packages were automatically installed and are no longer required:
  libev4 libexporter-tiny-perl libfmt8 libhttp-server-simple-perl liblmbase25 liblerc3 liblist-moreutils-perl liblist-moreutils-perl
  libplacebo192 libpoppler118 libpython3.9-minimal libpython3.9-stdlib libsvtav1enc0 libwebsockets16 python3-dataclasses-json
  python3-marshmallow-enum python3-mypy-extensions python3-responses python3-spyse python3-token-bucket python3-typing-inspect
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  tor-geoipdb torsocks
Suggested packages:
  mixmaster torbrowser-launcher apparmor-utils nix obfs4proxy
The following NEW packages will be installed:
  tor tor-geoipdb torsocks
0 upgraded, 3 newly installed, 0 to remove and 2 not upgraded.
Need to get 3,576 kB of archives.
After this operation, 16.6 MB of additional disk space will be used.
Get:1 http://kali.cs.nctu.edu.tw/kali kali-rolling/main amd64 tor amd64 0.4.7.10-1 [2,013 kB]
Get:2 http://kali.cs.nctu.edu.tw/kali kali-rolling/main amd64 tor-geoipdb all 0.4.7.10-1 [1,486 kB]
Get:3 http://kali.cs.nctu.edu.tw/kali kali-rolling/main amd64 torsocks amd64 2.3.0-3 [76.6 kB]
Fetched 3,576 kB in 4s (867 kB/s)
Selecting previously unselected package tor.
(Reading database ... 364730 files and directories currently installed.)
Preparing to unpack .../tor_0.4.7.10-1_amd64.deb ...
Unpacking tor (0.4.7.10-1) ...
Selecting previously unselected package tor-geoipdb.
Preparing to unpack .../tor-geoipdb_0.4.7.10-1_all.deb ...
```

iv) Install sshpass onto Kali Linux using the command: `sudo apt-get install sshpass`

Note: sshpass is a tool/command to provide the password automation for ssh base login.

```
(eddie@kali)~[~]
$ sudo apt-get install sshpass
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libev4 libexporter-tiny-perl libfmt8 libhttp-server-simple-perl liblmbase25 liblerc3 liblist-moreutils-perl liblist-moreutils-perl
  libplacebo192 libpoppler118 libpython3.9-minimal libpython3.9-stdlib libsvtav1enc0 libwebsockets16 python3-dataclasses-json
  python3-marshmallow-enum python3-mypy-extensions python3-responses python3-spyse python3-token-bucket python3-typing-inspect
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  sshpass
0 upgraded, 1 newly installed, 0 to remove and 2 not upgraded.
Need to get 13.0 kB of archives.
After this operation, 38.9 kB of additional disk space will be used.
Get:1 http://http.kali.org/kali kali-rolling/main amd64 sshpass amd64 1.09-1+b1 [13.0 kB]
Fetched 13.0 kB in 2s (7,543 B/s)
Selecting previously unselected package sshpass.
(Reading database ... 364806 files and directories currently installed.)
Preparing to unpack .../sshpass_1.09-1+b1_amd64.deb ...
Unpacking sshpass (1.09-1+b1) ...
Setting up sshpass (1.09-1+b1) ...
Processing triggers for man-db (2.10.2-3) ...
Processing triggers for kali-menu (2022.4.1) ...
```

4) Function 'anon' in the script is to run nipe to ensure that we are anonymous.

A). We would need to change the directory into the nipe folder where the nipe program is located in order to run it using the command: `cd "/home/kali/nipe"`,

Note: Use the command: `ls` to ensure that the nipe program "nipe.pl" is in the folder.

```
(eddie@kali)-[~]
$ cd "/home/eddie/nipe"

(eddie@kali)-[~/nipe]
$ ls
lib  LICENSE.md  nipe.pl  README.md  SECURITY.md

(eddie@kali)-[~/nipe]
$
```

B). Start/restart the nipe program to route the network anonymously using the command: `sudo perl nipe.pl restart`.

C). Follow by checking in Nipe is running using the command: `sudo perl nipe.pl status`.

When Nipe successfully executed the output will print out the status 'activated' with its anonymous IP address.

```
(eddie@kali)-[~/nipe]
$ sudo perl nipe.pl restart
[sudo] password for eddie:

(eddie@kali)-[~/nipe]
$ sudo perl nipe.pl status

[+] Status: activated.
[+] Ip: 185.220.101.2
```

D). To ensure the IP address is anonymous, by checking the connection is from your country of origin using the command: `curl ifconfig.io/country_code`

The output is "T1" indicating that the connection is anonymous.

Note: This command will output the country code of the current connection.

```
(eddie@kali)-[~/nipe]
$ sudo perl nipe.pl restart
[sudo] password for eddie:

(eddie@kali)-[~/nipe]
$ sudo perl nipe.pl status

[+] Status: activated.
[+] Ip: 185.220.101.168
```

5) Function “scan” is to automate access to the remote server via ssh using sshpass and executing Nmap scans, Masscan and Whois queries and saving onto the local host.

A). In the script, **echo “Input remote server IP address:”** for the user to input the IP address in order to save in a variable **“ip”** using the **“read”** command.

```
55 | echo "Input remote server IP address:"
56 | read ip
```

Followed by, **echo “Input remote server username:”** for the user to input the IP address in order to save in a variable **“user”** using the **“read”** command.

```
57 | echo "Input remote server username:"
58 | read user
```

Finally, **echo “Password:”** for the user to input the IP address in order to save in a variable **“passwd”** using the **“read”** command.

```
59 | echo "Password:"
60 | read passwd
```

B). Change the directory into the “kali” folder to save the scans files in it using the command: **cd /home/kali**

```
63 | #Changing to directory folder to save the scans
64 | cd /home/kali
```

C). Access the remote server using via ssh using sshpass and executing Whois query and storing it in the local host using the command:

**sshpass -p \$passwd ssh -o StrictHostKeyChecking=No \$user@\$ip "whois 8.8.8.8" > whois.scan**

Notes: Whois command is a query information about the registered Domain Names, an IP address block, Name Servers and a much wider range of information services

```
(kali@kali)-[~]
$ sshpass -p tc ssh -o StrictHostKeyChecking=No tc@192.168.149.131 "whois 8.8.8.8" > whois.scan

(kali@kali)-[~]
$ cat whois.scan
#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2022, American Registry for Internet Numbers, Ltd.
#
# start
54 | function scan()
55 | {
56 |     echo "Input remote server IP address:"
57 |     read ip
58 |     echo "Input remote server username:"
59 |     read user
60 |     echo "Password:"
61 |     read passwd
62 |     cd /home/kali
63 |     #To access the remote server and execute 'whois' and saving in 'whois.scan' file
64 |     sshpass -p $passwd ssh -o StrictHostKeyChecking=No $user@$ip "whois 8.8.8.8" > whois.scan
65 |     #To access the remote server and execute 'nmap' and saving in 'nmap version.scan'
66 |     sshpass -p $passwd ssh -o StrictHostKeyChecking=No $user@$ip "nmap 8.8.8.8 -v" > nmap.version.scan
67 | }
```



Syntax: `sshpass -p <remote server password> ssh -o <option from the config file> username@remoteserverIP <"command on executed on the remote server"> > <file name on the local host>`

Meaning of flags:

-p = To set the prompt password

-o = To give options in the format used in the configuration file.

Notes: Example, in this case we use 'StrictHostKeyChecking=No'. Which disable host key checking from prompting during connection.

D). Access the remote server using via ssh using sshpass and executing nmap scan and storing it in the local host using the command:

`sshpass -p $passwd ssh -o StrictHostKeyChecking=No $user@$ip "nmap 8.8.8.8 -Pn" > nmap_version.scan`

```
(kali㉿kali)-[~]
└─$ sshpass -p tc ssh -o StrictHostKeyChecking=No tc@192.168.149.131 "nmap 8.8.8.8 -Pn" > nmap_Pn.scan

(kali㉿kali)-[~]
└─$ cat nmap_Pn.scan
Starting Nmap 7.80 ( https://nmap.org ) at 2022-10-08 07:46 UTC
Nmap scan report for dns.google (8.8.8.8)
Host is up (0.0072s 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 57.28 seconds

(kali㉿kali)-[~]
└─$
```

Notes: Nmap is tool to explore network ports, the flag -Pn means to treat all host as if its online but avoid being discovered.

E).Access the remote server using via ssh using sshpass and executing masscan and storing it in the local host using the command:

`sshpass -p $passwd ssh -o StrictHostKeyChecking=No $user@$ip "sudo -S masscan 8.8.8.8 -p80 -p443" > masscan_p80_p443.scan`

```
(kali㉿kali)-[~]
└─$ sshpass -p tc ssh -o StrictHostKeyChecking=No tc@192.168.149.131 "sudo -S masscan 8.8.8.8 -p80 -p443" > masscan_p80_p443
.scan
[sudo] password for tc: tc
Starting masscan 1.3.2 (http://bit.ly/14GZzcT) at 2022-10-08 08:11:39 GMT
Initiating SYN Stealth Scan
Scanning 1 hosts [2 ports/host]

(kali㉿kali)-[~]
└─$ cat masscan_p80_p443.scan
Discovered open port 443/tcp on 8.8.8.8

(kali㉿kali)-[~]
└─$
```

Notes: Masscan is the fast internet scanner that scans for open port at rate of 100packets /s, the flag -S enables the prompt of the sudo password when executing the command on a remote server.

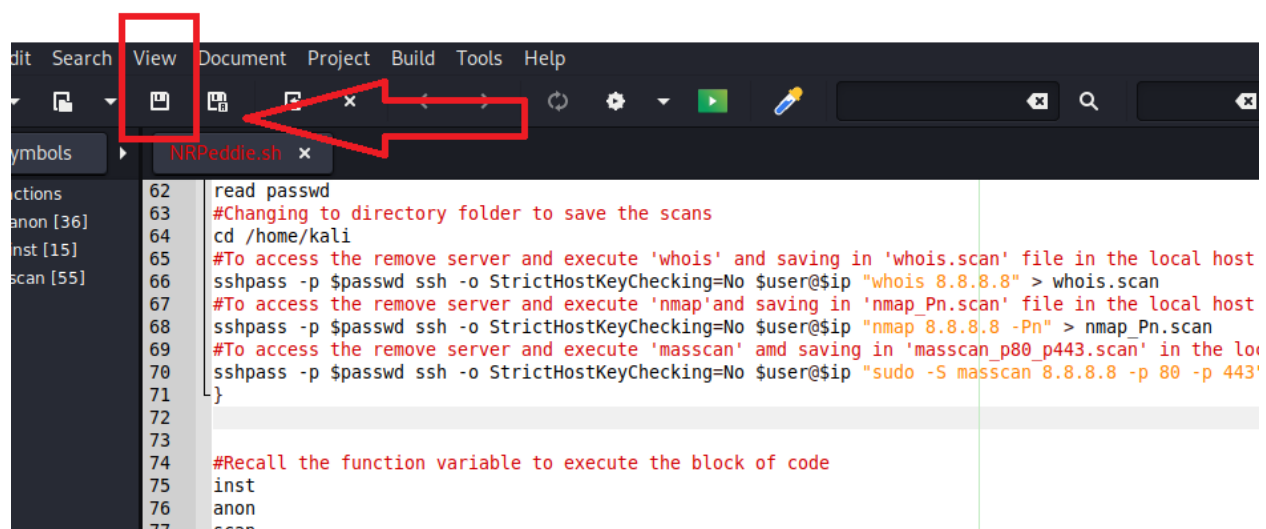


## 6. Execute the function variable by recalling them in this order

1. inst
2. anon
3. scan

```
sshpass -p $passwd ssh -o StrictHostKeyChecking=No $user@$ip "nmap 8.8.8.8.  
#To access the remove server and execute 'masscan' amd saving in 'masscan  
sshpass -p $passwd ssh -o StrictHostKeyChecking=No $user@$ip "sudo -S mas  
}  
  
inst  
anon  
scan
```

## 7). Saving the script file by clicking the floppy disk icon.



```
62 read passwd  
63 #Changing to directory folder to save the scans  
64 cd /home/kali  
65 #To access the remove server and execute 'whois' and saving in 'whois.scan' file in the local host  
66 sshpass -p $passwd ssh -o StrictHostKeyChecking=No $user@$ip "whois 8.8.8.8" > whois.scan  
67 #To access the remove server and execute 'nmap'and saving in 'nmap_Pn.scan' file in the local host  
68 sshpass -p $passwd ssh -o StrictHostKeyChecking=No $user@$ip "nmap 8.8.8.8 -Pn" > nmap_Pn.scan  
69 #To access the remove server and execute 'masscan' amd saving in 'masscan_p80_p443.scan' in the lo  
70 sshpass -p $passwd ssh -o StrictHostKeyChecking=No $user@$ip "sudo -S masscan 8.8.8.8 -p 80 -p 443'  
71 }  
72  
73  
74 #Recall the function variable to execute the block of code  
75 inst  
76 anon  
77 scan
```