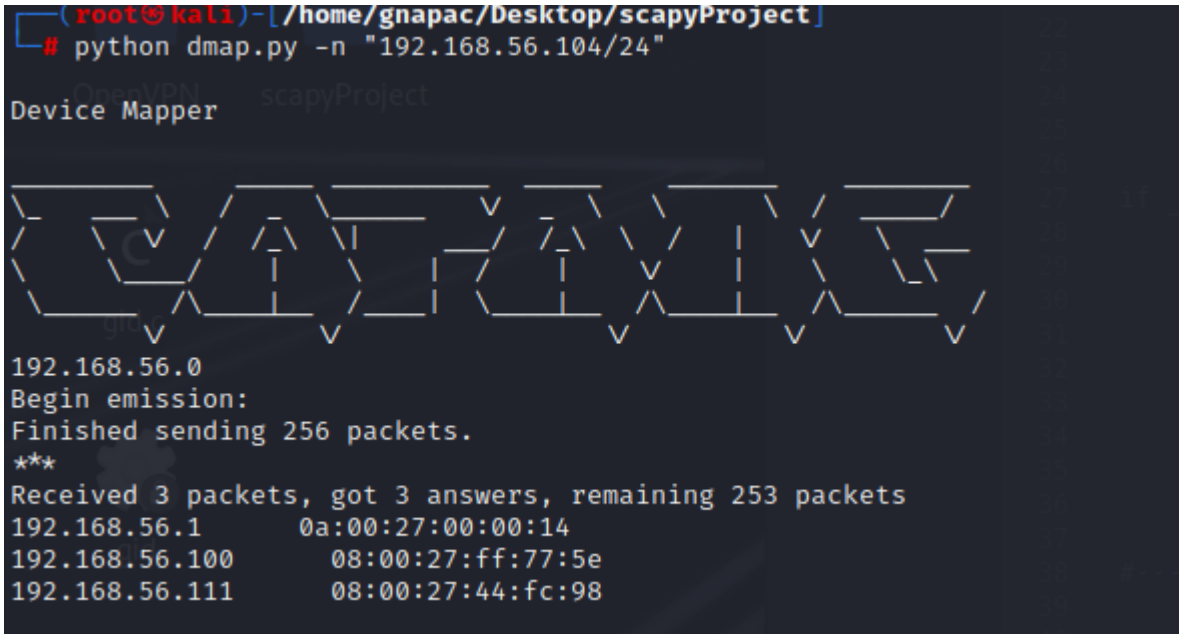
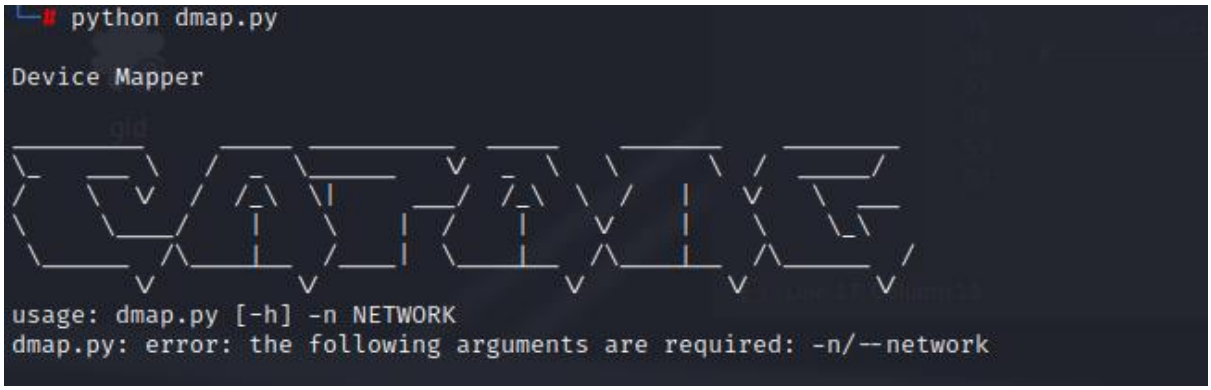
Device Mapper





Source Code

#!/usr/bin/env python

```
import scapy.all as scapy
import argparse
from signal import signal, SIGINT


#DEVICE MAPPER

def handler(signal_received, frame):
    # Handle any cleanup here
    print('\n.............Exiting gracefully')
    front()
    exit(0)

def front():
    # Molek sikit
    print('_________    _____ __________ _____   _______   ________ ')
    print('\\_  ___ \\  / _ \\_____  \\/ _ \\  \\     \\ /  _____/')
    print('/    \\ \\/ / /_\\ \\|     ___/ /_\\ \\ \\ /   |   \\/  \\ ___ ')
    print('\\    \\____/   |    \\    | /    |    \\/    |    \\  \\_\\ ')
    print(' \\_______ /\\____|__ /____|  \\____|__ /\\____|__ /\\_______ /')
    print('        \\/        \\/             \\/        \\/       \\/ ')




if __name__ == '__main__':
    print('\nDevice Mapper\n')
    front()
    signal(SIGINT, handler)
    parser = argparse.ArgumentParser()
```

```
    # Device Mapping Part
    parser.add_argument("-n", "--network", dest="network", help="Target Network", required=True)

    options = parser.parse_args()

#------------------ DEVICE MAPPER ---------------------
    request = scapy.ARP()
    request.pdst = options.network
    print(request.pdst)
    broadcast = scapy.Ether()

    broadcast.dst = 'ff:ff:ff:ff:ff:ff'

    request_broadcast = broadcast / request
    clients = scapy.srp(request_broadcast, timeout = 1,iface="eth1")[0]
    for element in clients: #loop through received packets
        print(element[1].psrc + "      " + element[1].hwsrc)
#------------------ DEVICE MAPPER ---------------------
```