

Device Mapper

```
(root@kali)-[/home/gnapac/Desktop/scapyProject]
# python dmap.py -n "192.168.56.104/24"

Device Mapper

192.168.56.0
Begin emission:
Finished sending 256 packets.
***
Received 3 packets, got 3 answers, remaining 253 packets
192.168.56.1      0a:00:27:00:00:14
192.168.56.100   08:00:27:ff:77:5e
192.168.56.111   08:00:27:44:fc:98
```

```
(root@kali)-[/home/gnapac/Desktop/scapyProject]
# python dmap.py --help

Device Mapper

usage: dmap.py [-h] -n NETWORK

options:
  -h, --help            show this help message and exit
  -n NETWORK, --network NETWORK
                        Target Network
```

```
# python dmap.py

Device Mapper

usage: dmap.py [-h] -n NETWORK
dmap.py: error: the following arguments are required: -n/--network
```

Source Code

```
#!/usr/bin/env python
```

from signal import signal, SIGINT

```
exit(0)
```

```
print(' \\V \\V \\V \\V \\V')
```

```
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 -----
```