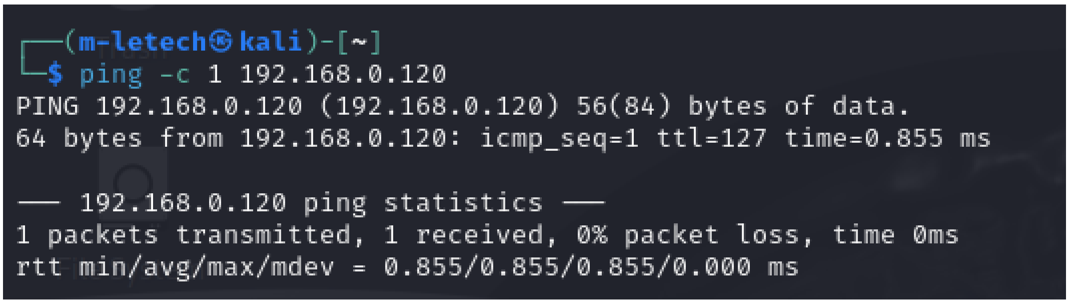
Basic Ping Script

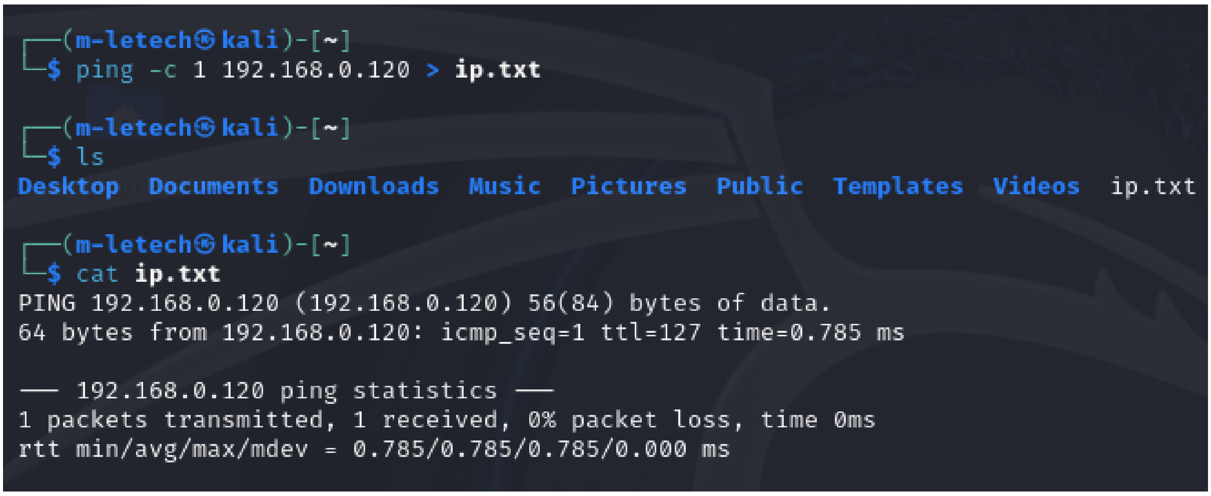
The purpose of this is just to show some basic automation on ping in a bash script and how to automate nmap scanning on multiple IPs at a time.

1.



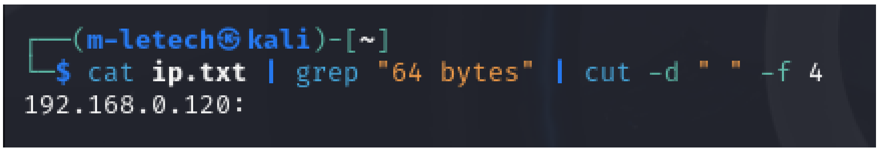
```
┌──(m-letech㉿kali)-[~]
└─$ ping -c 1 192.168.0.120
PING 192.168.0.120 (192.168.0.120) 56(84) bytes of data.
64 bytes from 192.168.0.120: icmp_seq=1 ttl=127 time=0.855 ms

─── 192.168.0.120 ping statistics ───
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.855/0.855/0.855/0.000 ms
```

2. Here we directly sent the output of the ping to ip.txt file



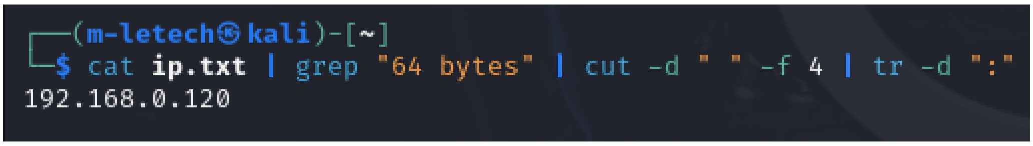
```
┌──(m-letech㉿kali)-[~]
└─$ ping -c 1 192.168.0.120 > ip.txt

┌──(m-letech㉿kali)-[~]
└─$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  ip.txt

┌──(m-letech㉿kali)-[~]
└─$ cat ip.txt
PING 192.168.0.120 (192.168.0.120) 56(84) bytes of data.
64 bytes from 192.168.0.120: icmp_seq=1 ttl=127 time=0.785 ms

─── 192.168.0.120 ping statistics ───
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.785/0.785/0.785/0.000 ms
```

3. This command is designed to find lines containing grep “64 bytes” and then cutthe fourth field, which typically corresponds to the IP address in the ping response



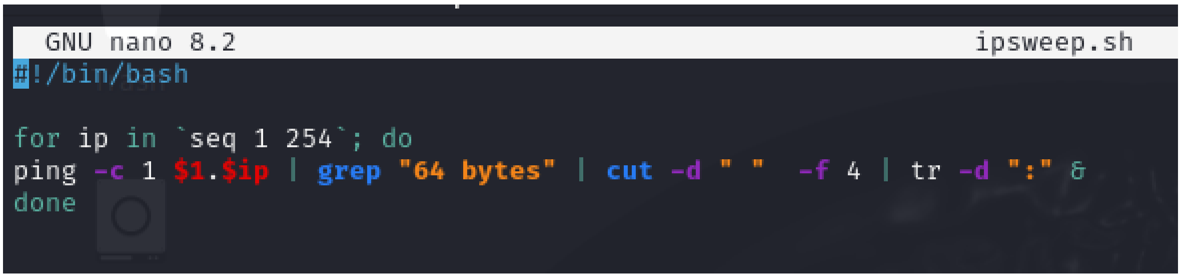
```
┌──(m-letech㉿kali)-[~]
└─$ cat ip.txt | grep "64 bytes" | cut -d " " -f 4
192.168.0.120:
```

4. The command below does similar thing to cut out the semicolon.



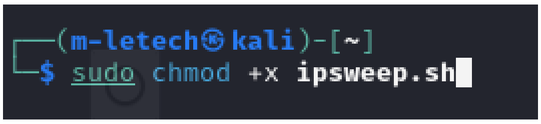
```
┌──(m-letech㉿kali)-[~]
└─$ cat ip.txt | grep "64 bytes" | cut -d " " -f 4 | tr -d ":"
192.168.0.120
```

5. Here is a bash script to automate the process



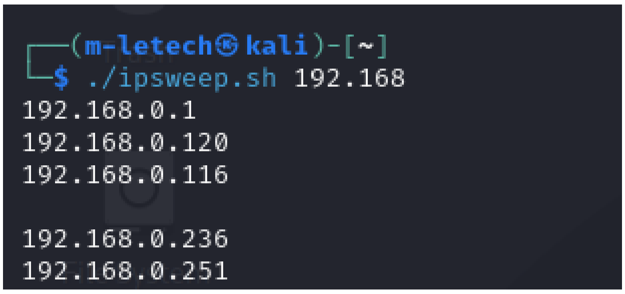
```
  GNU nano 8.2                                          ipsweep.sh
#!/bin/bash

for ip in `seq 1 254`; do
ping -c 1 $1.$ip | grep "64 bytes" | cut -d " "  -f 4 | tr -d ":" &
done
```

6. Here i made the script executable



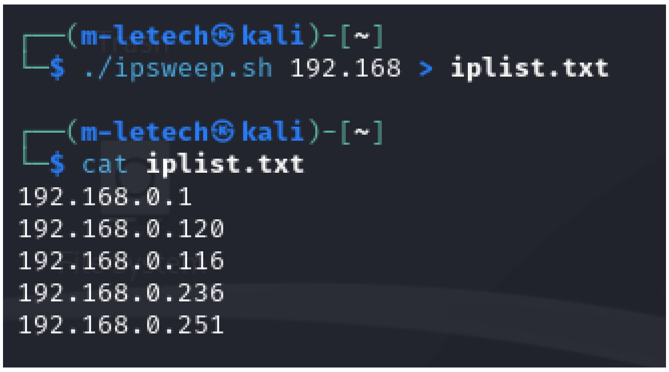
```
┌──(m-letech㉿kali)-[~]
└─$ sudo chmod +x ipsweep.sh
```

7. Here is the result when the bash script is run with the network section of the IP



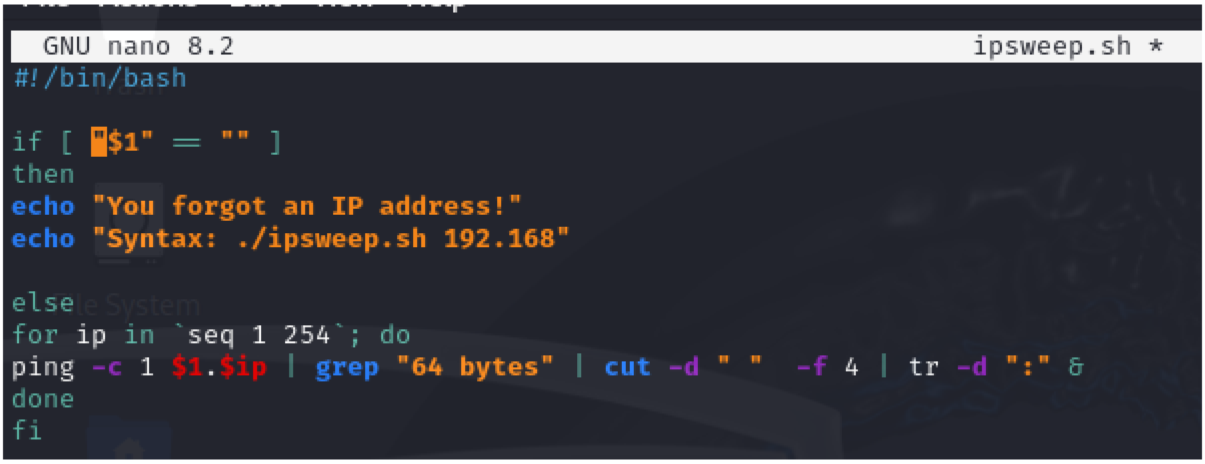
```
┌──(m-letech㉿kali)-[~]
└─$ ./ipsweep.sh 192.168
192.168.0.1
192.168.0.120
192.168.0.116

192.168.0.236
192.168.0.251
```

8. Redirection of the script to a text file



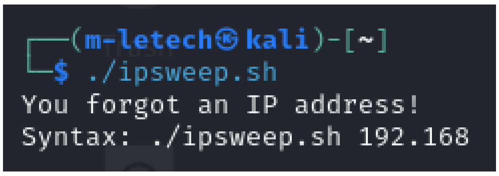
```
┌──(m-letech㉿kali)-[~]
└─$ ./ipsweep.sh 192.168 > iplist.txt

┌──(m-letech㉿kali)-[~]
└─$ cat iplist.txt
192.168.0.1
192.168.0.120
192.168.0.116
192.168.0.236
192.168.0.251
```

9. Updated script with condition



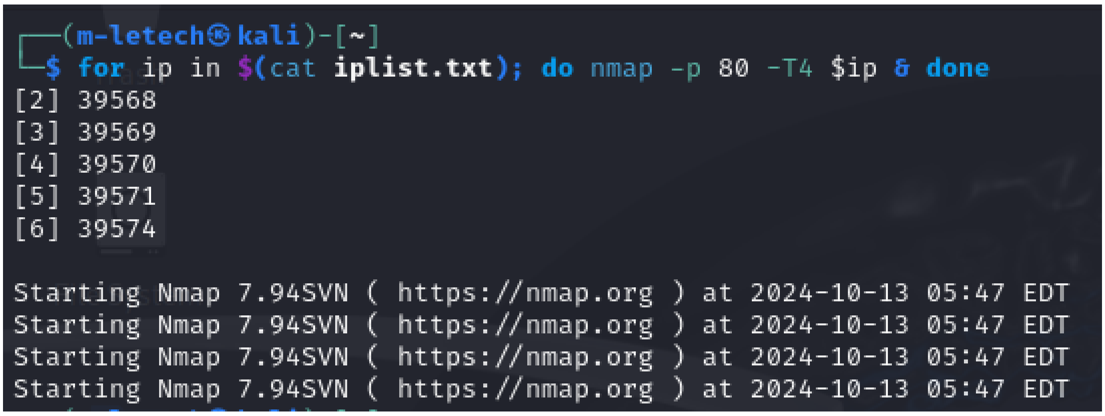
```
  GNU nano 8.2                                                    ipsweep.sh *
#!/bin/bash

if [ "$1" == "" ]
then
echo "You forgot an IP address!"
echo "Syntax: ./ipsweep.sh 192.168"

else
for ip in `seq 1 254`; do
ping -c 1 $1.$ip | grep "64 bytes" | cut -d " "  -f 4 | tr -d ":" &
done
fi
```

Test output of the updated script

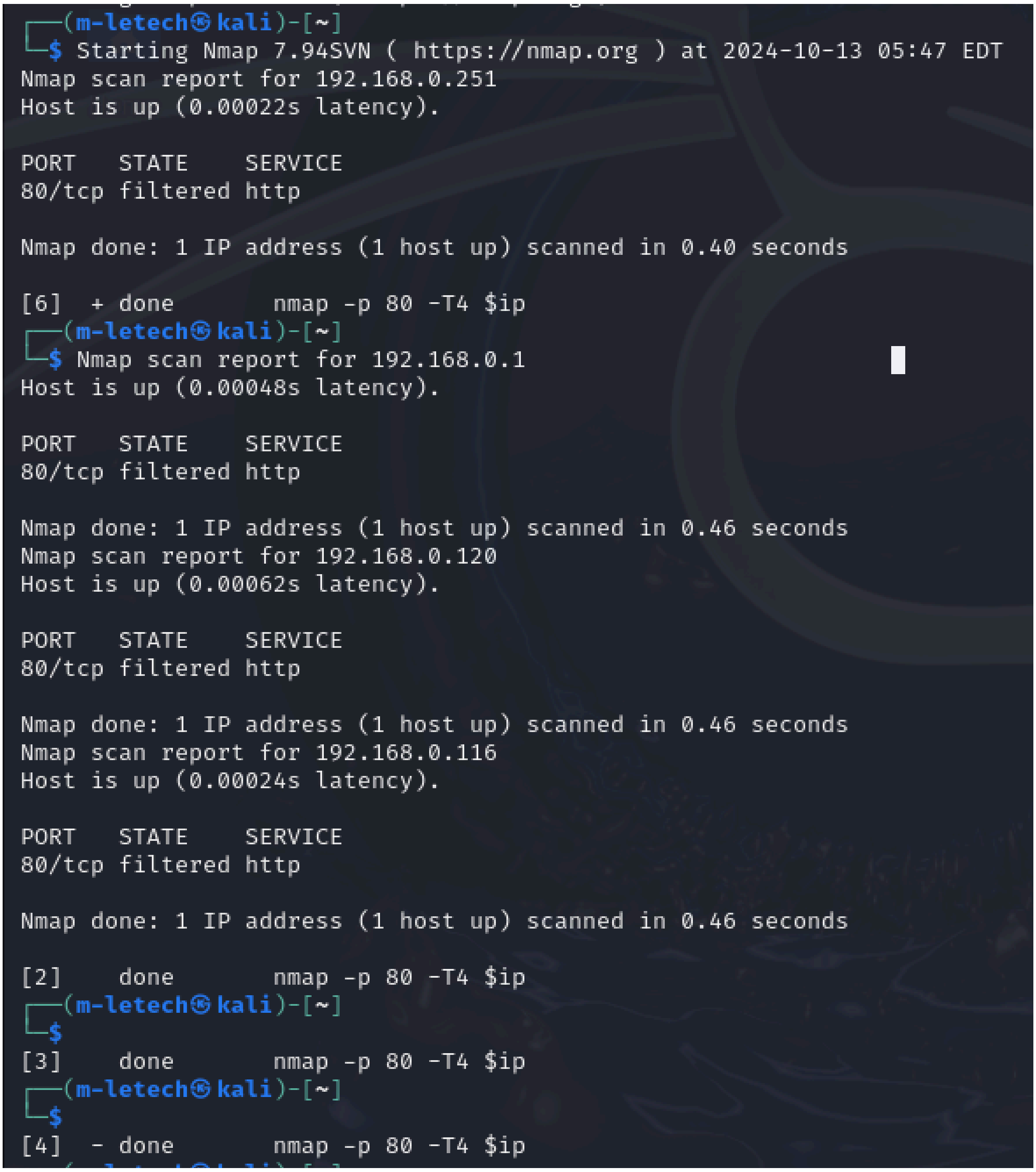


```
┌──(m-letech㉿kali)-[~]
└─$ ./ipsweep.sh
You forgot an IP address!
Syntax: ./ipsweep.sh 192.168
```

10. Using nmap to run the ping script and the output



```
┌──(m-letech㉿kali)-[~]
└─$ for ip in $(cat iplist.txt); do nmap -p 80 -T4 $ip & done
[2] 39568
[3] 39569
[4] 39570
[5] 39571
[6] 39574

Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-13 05:47 EDT
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-13 05:47 EDT
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-13 05:47 EDT
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-13 05:47 EDT
```



```
┌──(m-letech㉿kali)-[~]
└─$ Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-13 05:47 EDT
Nmap scan report for 192.168.0.251
Host is up (0.00022s latency).

PORT   STATE    SERVICE
80/tcp filtered http

Nmap done: 1 IP address (1 host up) scanned in 0.40 seconds

[6]  + done       nmap -p 80 -T4 $ip
┌──(m-letech㉿kali)-[~]
└─$ Nmap scan report for 192.168.0.1
Host is up (0.00048s latency).

PORT   STATE    SERVICE
80/tcp filtered http

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
Nmap scan report for 192.168.0.120
Host is up (0.00062s latency).

PORT   STATE    SERVICE
80/tcp filtered http

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
Nmap scan report for 192.168.0.116
Host is up (0.00024s latency).

PORT   STATE    SERVICE
80/tcp filtered http

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds

[2]    done       nmap -p 80 -T4 $ip
┌──(m-letech㉿kali)-[~]
└─$
[3]    done       nmap -p 80 -T4 $ip
┌──(m-letech㉿kali)-[~]
└─$
[4]  - done       nmap -p 80 -T4 $ip
```

From the result I could see that multiple IP addresses were scanned simultaneously and the result printed.