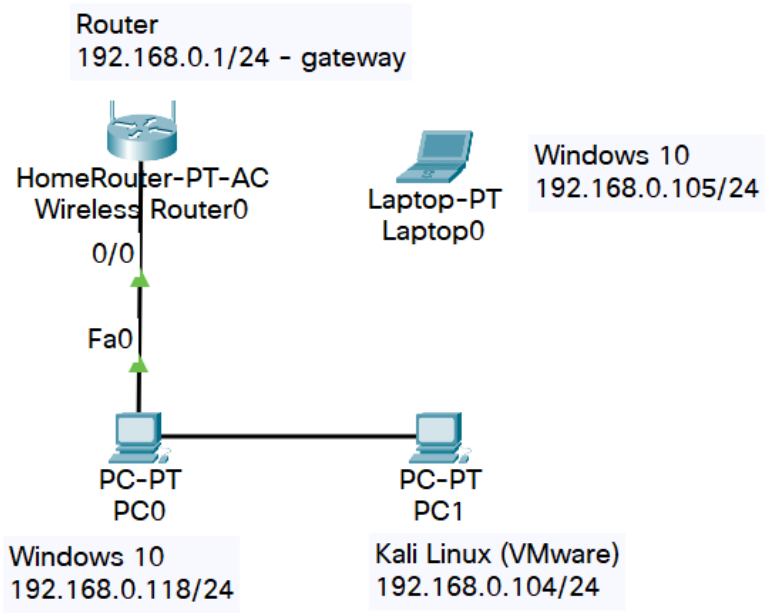
# Informatyka Śledcza - Laboratorium 4

Magdalena Ślusarczyk

## 1 Informacje o sieci

Rysunek przedstawiajacy sieć:




Wyniki poleceń ifconfig i nmap:

```
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.104  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::20c:29ff:fe76:a297  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:76:a2:97  txqueuelen 1000  (Ethernet)
        RX packets 105722  bytes 149141253 (142.2 MiB)
        RX errors 0  dropped 57  overruns 0  frame 0
        TX packets 55279  bytes 4595852 (4.3 MiB)
        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 176  bytes 10513 (10.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 176  bytes 10513 (10.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

TRACEROUTE
HOP RTT     ADDRESS
1   0.10 ms 192.168.0.118

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 31.57 seconds
```

```
Nmap scan report for 192.168.0.105
Host is up.
```

# 2 TCPdump

## 2.1 Obserwowanie pingu z innej maszyny

Ping z 192.168.0.105 do 192.168.0.104.

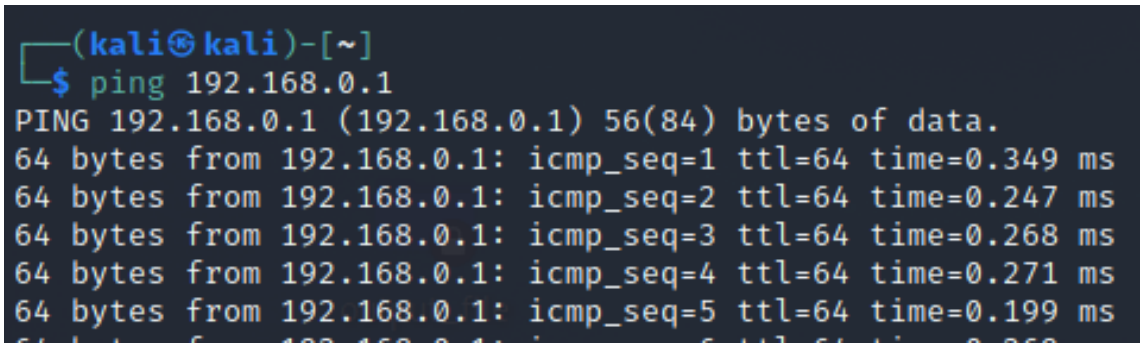
```
PS C:\Users\magda> ping 192.168.0.104

Pinging 192.168.0.104 with 32 bytes of data:
Reply from 192.168.0.104: bytes=32 time=1ms TTL=64
Reply from 192.168.0.104: bytes=32 time=1ms TTL=64
Reply from 192.168.0.104: bytes=32 time=3ms TTL=64
Reply from 192.168.0.104: bytes=32 time=6ms TTL=64

Ping statistics for 192.168.0.104:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 6ms, Average = 2ms
```

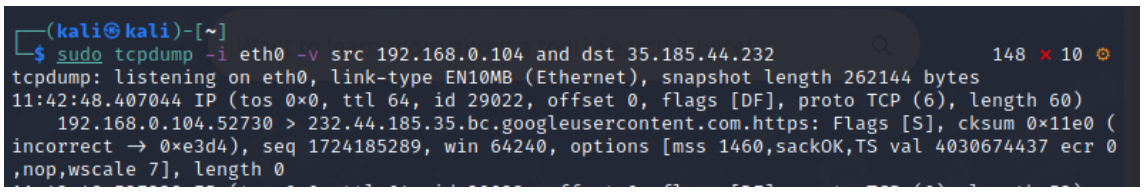
```
┌──(kali㉿kali)-[~]
└─$ sudo tcpdump -i eth0 -v host 192.168.0.105
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
11:14:05.009960 IP (tos 0x0, ttl 128, id 56311, offset 0, flags [none], proto ICMP (1), length 60
    192.168.0.105 > 192.168.0.104: ICMP echo request, id 1, seq 5, length 40
11:14:05.009983 IP (tos 0x0, ttl 64, id 11528, offset 0, flags [none], proto ICMP (1), length 60)
    192.168.0.104 > 192.168.0.105: ICMP echo reply, id 1, seq 5, length 40
11:14:05.500743 ARP, Ethernet (len 6), IPv4 (len 4), Request who-has 192.168.0.1 tell 192.168.0.1
11:14:06.018882 IP (tos 0x0, ttl 128, id 56312, offset 0, flags [none], proto ICMP (1), length 60
    192.168.0.105 > 192.168.0.104: ICMP echo request, id 1, seq 6, length 40
11:14:06.018906 IP (tos 0x0, ttl 64, id 11775, offset 0, flags [none], proto ICMP (1), length 60)
    192.168.0.104 > 192.168.0.105: ICMP echo reply, id 1, seq 6, length 40
11:14:07.026916 IP (tos 0x0, ttl 128, id 56313, offset 0, flags [none], proto ICMP (1), length 60
    192.168.0.105 > 192.168.0.104: ICMP echo request, id 1, seq 7, length 40
11:14:07.026939 IP (tos 0x0, ttl 64, id 11894, offset 0, flags [none], proto ICMP (1), length 60)
    192.168.0.104 > 192.168.0.105: ICMP echo reply, id 1, seq 7, length 40
11:14:08.048003 IP (tos 0x0, ttl 128, id 56314, offset 0, flags [none], proto ICMP (1), length 60
    192.168.0.105 > 192.168.0.104: ICMP echo request, id 1, seq 8, length 40
11:14:08.048022 IP (tos 0x0, ttl 64, id 11939, offset 0, flags [none], proto ICMP (1), length 60)
    192.168.0.104 > 192.168.0.105: ICMP echo reply, id 1, seq 8, length 40
11:14:09.966627 ARP, Ethernet (len 6), IPv4 (len 4), Request who-has 192.168.0.104 (00:0c:29:76:a
11:14:09.966639 ARP, Ethernet (len 6), IPv4 (len 4), Reply 192.168.0.104 is-at 00:0c:29:76:a2:97
11:14:11.027425 ARP, Ethernet (len 6), IPv4 (len 4), Request who-has 192.168.0.105 tell 192.168.0
11:14:11.028367 ARP, Ethernet (len 6), IPv4 (len 4), Reply 192.168.0.105 is-at 64:6c:80:db:e2:2d
```

### 2.2 Ping do bramy domyślnej



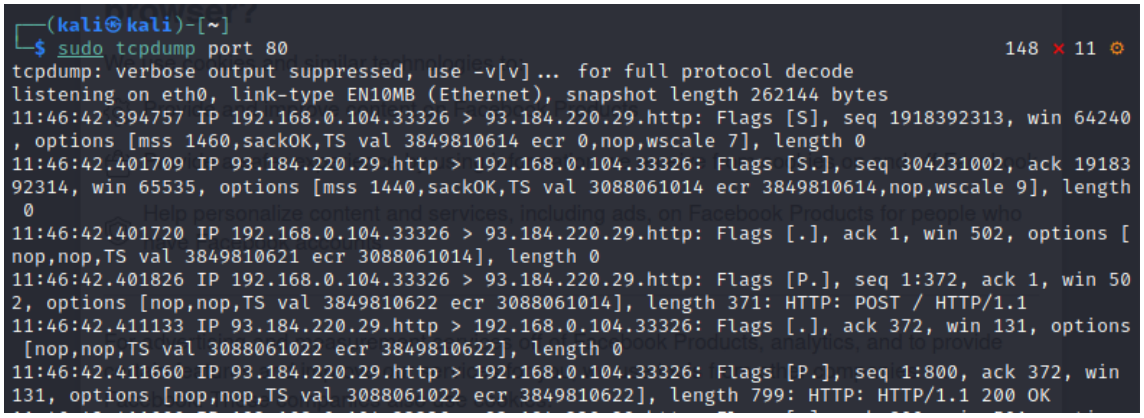

### 2.3 Wyłapanie zapytania do jakiejś strony internetowej

Strona: https://www.kali.org/tools/

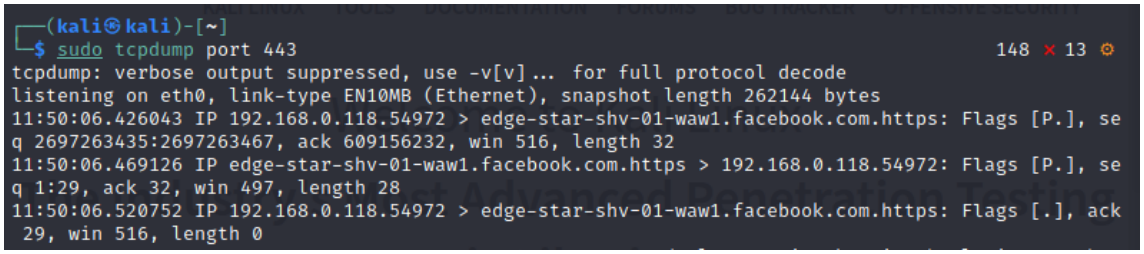



### 2.4 Wyłapanie ruchu na porcie 80

Strona: https://www.facebook.com



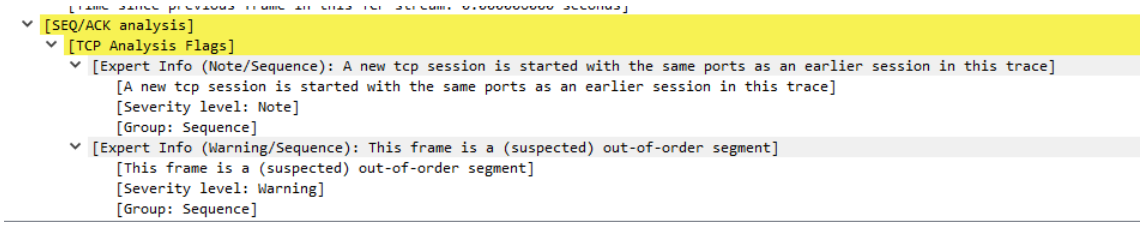

### 2.5 Wyłapanie ruchu na porcie 443




## 3 Wireshark

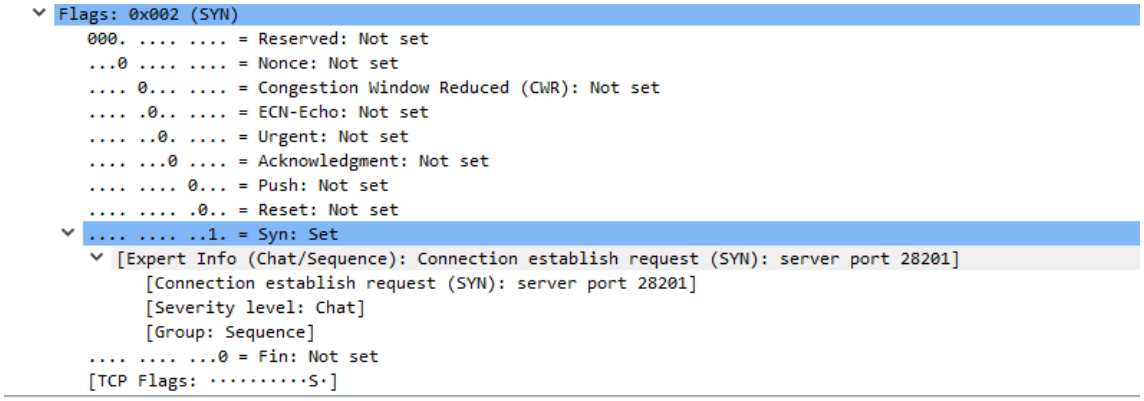
### 3.1 Skanowanie stealth scan

W Wiresharku na masyznie skanowanej pojawiły się liczne zapytania TCP Out-of-Order, które program analizuje jako podejrzane i nadaje im severity level: warning.




### 3.2 Skanowanie z wykorzystaniem fragmentacji pakietu

W tym przypadku wireshark nie wykrył niczego podejrzanego:




Druga metoda zdecydowanie mniej widoczna dla osoby analizującej ruch.

## 4 Analiza pliku

Adres IP komputera poddanego analizie: 172.16.17.131
Gateway: 172.16.17.2
Zostały wykorzystane maszyny wirtualne, analizując dowolny pakiet możemy zobaczyć, że adresy MAC należą do maszyn VMware.
Maszyna była skanowana z adresu 172.16.17.128, świadczą o tym liczbe połączenia TCP na ten sam adres, ale różne porty. Porty były sprawdzane w losowej kolejności metodą stealth scan.
Adres MAC 00:0c:29:ec:8a:14 należy do atakującego.
Z pakietu nr 2252 można wyciągnąć plik exe, który okazuje się być trojanem.
Do przesłania danych wykorzystano port 49162.
Nazwa komputer: Komputer/Kamil.
NetworkMiner nie odnajduje zainfekowanego pliku.