Домашнее задание №4 Сети, маршрутизация. Часть 1

Группа: Cyb07-onl

Студент: Парфимович Алексей

1. ifconfig

- отключить интерфейс eth0 на ВМ Kali

```
┌──(user㉿kali)-[~]
└─$ ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.101  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::a00:27ff:fef2:53d0  prefixlen 64  scopeid 0×20<link>
        ether 08:00:27:f2:53:d0  txqueuelen 1000  (Ethernet)
        RX packets 6  bytes 836 (836.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 34  bytes 4348 (4.2 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 0×10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 316  bytes 18920 (18.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 316  bytes 18920 (18.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

┌──(user㉿kali)-[~]
└─$ sudo ifconfig eth0 down
[sudo] password for user:

┌──(user㉿kali)-[~]
└─$ sudo ifconfig eth0
eth0: flags=4098<BROADCAST,MULTICAST>  mtu 1500
        ether 08:00:27:f2:53:d0  txqueuelen 1000  (Ethernet)
        RX packets 6  bytes 836 (836.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 34  bytes 4348 (4.2 KiB)
        TX errors 0  dropped 2 overruns 0  carrier 0  collisions 0
```

- включить интерфейс eth0 на ВМ Kali

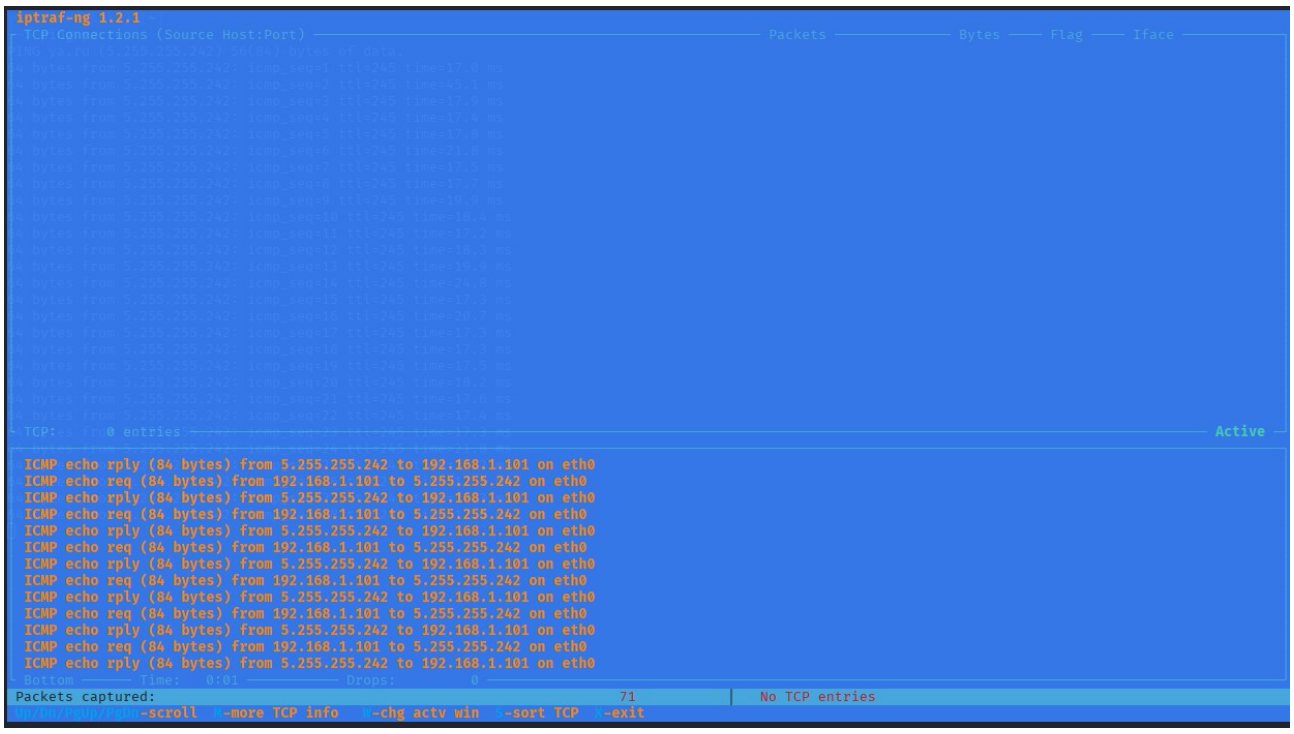
```
┌──(user㉿kali)-[~]
└─$ sudo ifconfig eth0 up

┌──(user㉿kali)-[~]
└─$ sudo ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.101  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::a00:27ff:fef2:53d0  prefixlen 64  scopeid 0×20<link>
        ether 08:00:27:f2:53:d0  txqueuelen 1000  (Ethernet)
        RX packets 8  bytes 1288 (1.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 50  bytes 5876 (5.7 KiB)
        TX errors 0  dropped 2 overruns 0  carrier 0  collisions 0


┌──(user㉿kali)-[~]
└─$
```

2. iptraf

- установить приложение iptraf

```
┌──(user㉿kali)-[~]
└─$ sudo apt update
Get:1 http://kali.download/kali kali-rolling InRelease [41.5 kB]
Get:2 http://kali.download/kali kali-rolling/contrib Sources [82.1 kB]
Get:3 http://kali.download/kali kali-rolling/main Sources [17.5 MB]
Get:4 http://kali.download/kali kali-rolling/non-free Sources [123 kB]
Get:5 http://kali.download/kali kali-rolling/main amd64 Packages [21.2 MB]
Get:6 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [51.8 MB]
Get:7 http://kali.download/kali kali-rolling/non-free amd64 Packages [200 kB]
Get:8 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [911 kB]
Get:9 http://kali.download/kali kali-rolling/contrib amd64 Packages [120 kB]
Get:10 http://kali.download/kali kali-rolling/contrib amd64 Contents (deb) [326 kB]
Fetched 92.4 MB in 2min 45s (558 kB/s)
992 packages can be upgraded. Run 'apt list --upgradable' to see them.

┌──(user㉿kali)-[~]
└─$ iptraf
Command 'iptraf' not found, but can be installed with:
sudo apt install iptraf-ng
Do you want to install it? (N/y)y
sudo apt install iptraf-ng
Installing:
  iptraf-ng

Summary:
  Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 992
  Download size: 274 kB
  Space needed: 694 kB / 7,692 MB available

Get:1 http://kali.koyanet.lv/kali kali-rolling/main amd64 iptraf-ng amd64 1:1.2.1-2 [274 kB]
Fetched 274 kB in 9s (29.9 kB/s)
Selecting previously unselected package iptraf-ng.
(Reading database ... 411917 files and directories currently installed.)
Preparing to unpack .../iptraf-ng_1%3a1.2.1-2_amd64.deb ...
Unpacking iptraf-ng (1:1.2.1-2) ...
Setting up iptraf-ng (1:1.2.1-2) ...
Processing triggers for doc-base (0.11.2) ...
Processing 1 added doc-base file...
Processing triggers for man-db (2.13.1-1) ...
Processing triggers for kali-menu (2025.2.7) ...

┌──(user㉿kali)-[~]
└─$
```

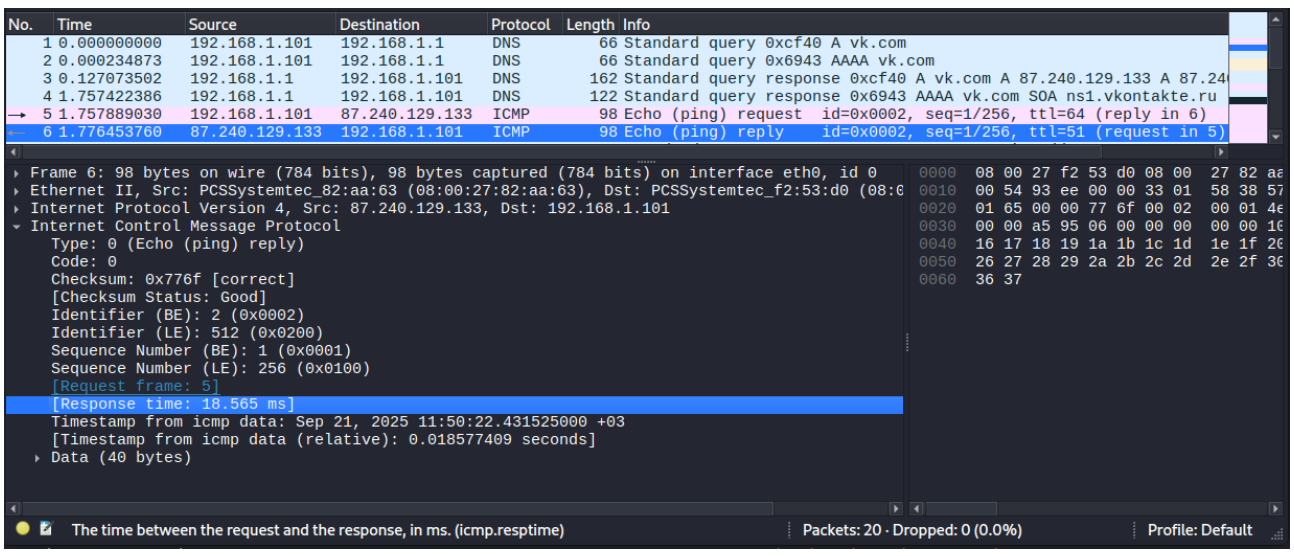
- запустить мониторинг трафика для сетевого интерфеса eth0, с отдельного терминала выполнить команду ping ya.ru, сделать скрин с пакетами в iptraf:



```
iptraf-ng 1.2.1
TCP Connections (Source Host:Port) ———————— Packets ———— Bytes —— Flag —— Iface
TCP:   0 entries ——————————————————————————————————————————————— Active
ICMP echo rply (84 bytes) from 5.255.255.242 to 192.168.1.101 on eth0
ICMP echo req (84 bytes) from 192.168.1.101 to 5.255.255.242 on eth0
ICMP echo rply (84 bytes) from 5.255.255.242 to 192.168.1.101 on eth0
ICMP echo req (84 bytes) from 192.168.1.101 to 5.255.255.242 on eth0
ICMP echo rply (84 bytes) from 5.255.255.242 to 192.168.1.101 on eth0
ICMP echo req (84 bytes) from 192.168.1.101 to 5.255.255.242 on eth0
ICMP echo rply (84 bytes) from 5.255.255.242 to 192.168.1.101 on eth0
ICMP echo req (84 bytes) from 192.168.1.101 to 5.255.255.242 on eth0
ICMP echo rply (84 bytes) from 5.255.255.242 to 192.168.1.101 on eth0
ICMP echo req (84 bytes) from 192.168.1.101 to 5.255.255.242 on eth0
ICMP echo rply (84 bytes) from 5.255.255.242 to 192.168.1.101 on eth0
ICMP echo req (84 bytes) from 192.168.1.101 to 5.255.255.242 on eth0
ICMP echo rply (84 bytes) from 5.255.255.242 to 192.168.1.101 on eth0
Bottom ——— Time:   0:01 ——————— Drops:        0
Packets captured:                                   71   | No TCP entries
Up/Dn/PgUp/PgDn-scroll  M-more TCP info   W-chg actv win  S-sort TCP  X-exit
```

3. wireshark

3.1 Начать захват трафика, через консоль отправить пинг по адресу vk.com, остановить захват трафика, сделать скрин экрана wireshark, предоставить описание перехваченных пакетов для протоколов DNS и ICMP:



```
No. Time          Source         Destination    Protocol Length Info
  1 0.000000000   192.168.1.101  192.168.1.1    DNS        66 Standard query 0xcf40 A vk.com
  2 0.000234873   192.168.1.101  192.168.1.1    DNS        66 Standard query 0x6943 AAAA vk.com
  3 0.127073502   192.168.1.1    192.168.1.101  DNS       162 Standard query response 0xcf40 A vk.com A 87.240.129.133 A 87.24
  4 1.757422386   192.168.1.1    192.168.1.101  DNS       122 Standard query response 0x6943 AAAA vk.com SOA ns1.vkontakte.ru
→ 5 1.757889030   192.168.1.101  87.240.129.133 ICMP       98 Echo (ping) request  id=0x0002, seq=1/256, ttl=64 (reply in 6)
← 6 1.776453760   87.240.129.133 192.168.1.101  ICMP       98 Echo (ping) reply    id=0x0002, seq=1/256, ttl=51 (request in 5)

› Frame 6: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface eth0, id 0
› Ethernet II, Src: PCSSystemtec_82:aa:63 (08:00:27:82:aa:63), Dst: PCSSystemtec_f2:53:d0 (08:6
› Internet Protocol Version 4, Src: 87.240.129.133, Dst: 192.168.1.101
⌄ Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0x776f [correct]
    [Checksum Status: Good]
    Identifier (BE): 2 (0x0002)
    Identifier (LE): 512 (0x0200)
    Sequence Number (BE): 1 (0x0001)
    Sequence Number (LE): 256 (0x0100)
    [Request frame: 5]
    [Response time: 18.565 ms]
    Timestamp from icmp data: Sep 21, 2025 11:50:22.431525000 +03
    [Timestamp from icmp data (relative): 0.018577409 seconds]
  › Data (40 bytes)

0000  08 00 27 f2 53 d0 08 00  27 82 aa
0010  00 54 93 ee 00 00 33 01  58 38 57
0020  01 65 00 00 77 6f 00 02  00 01 4e
0030  00 00 a5 95 06 00 00 00  00 00 10
0040  16 17 18 19 1a 1b 1c 1d  1e 1f 20
0050  26 27 28 29 2a 2b 2c 2d  2e 2f 30
0060  36 37

The time between the request and the response, in ms. (icmp.resptime)   Packets: 20 · Dropped: 0 (0.0%)   Profile: Default
```

Пакет 1: Запрос типа A для vk.com

- **Отправитель:** 192.168.1.101 (ВМ Kali в локальной сети)
- **Адресат:** 192.168.1.1 (локальный DNS-сервер)
- **Протокол:** DNS
- **Тип запроса:** A — запрос адреса IPv4
- **ID запроса:** 0xcf40
- **Запрашиваемое доменное имя:** vk.com

→ Выполняется запрос на получение IPv4-адреса сайта vk.com.

Пакет 2: Запрос типа AAAA для vk.com

- **Тип запроса:** AAAA — запрос адреса IPv6
- **ID запроса:** 0x6943

→ Выполняется запрос на получение IPv6-адрес сайта vk.com.

Пакет 3: Ответ на запрос типа A (IPv4)

- **Отправитель:** 192.168.1.1 (локальный DNS-сервер)
- **ID ответа:** 0xcf40 — соответствует ID пакета №1
- **Тип ответа:** A — возвращение адресов IPv4
- **Адреса:** 87.240.129.133, 87.240.132.72, 87.240.132.67, 93.186.225.194, 87.240.137.164, 87.240.132.78

→ Это IP-адреса серверов ВКонтакте.

Пакет 4: Ответ на запрос AAAA (IPv6)

- **ID ответа**: 0x6943 → соответствует ID пакета №2
- **Ответ**: SOA ns1.vkontakte.ru – в ответе возвращается код отказа!

→ SOA (Start of Authority) в ответе означает: «на DNS сервере нет AAAA-записи для данного доменого имени, передается информация об авторитативном сервере».

Пакет 5: Запрос типа ICMP Echo Request

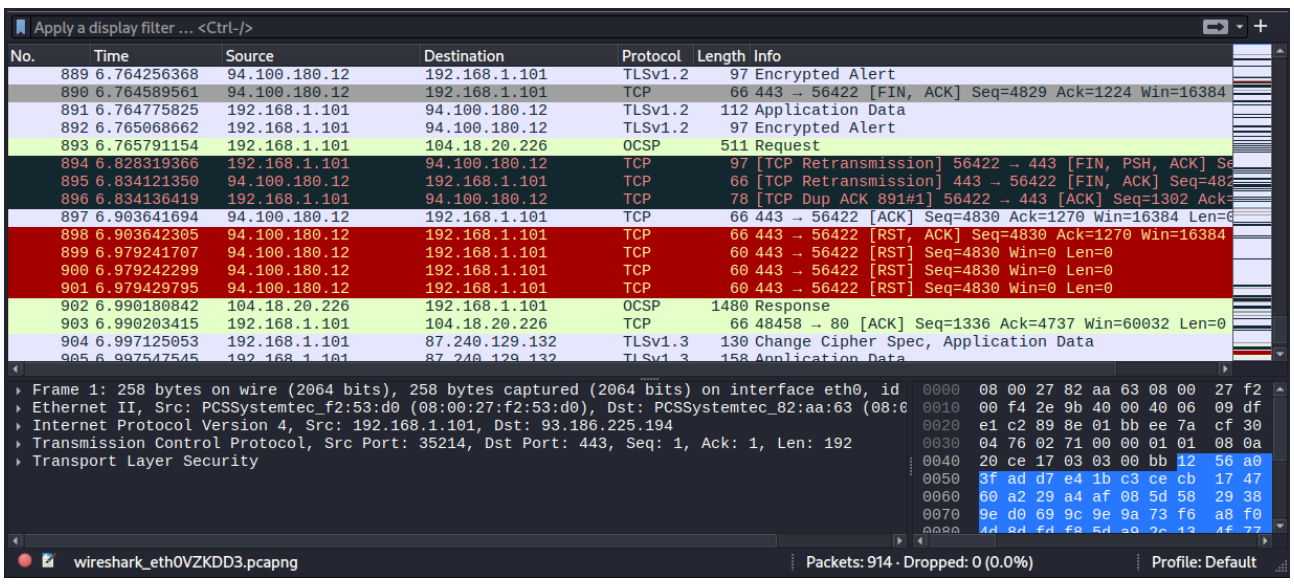
- **Отправитель**: 192.168.1.101 (ВМ Kali в локальной сети)
- **Адресат**: 87.240.129.133 (Один из серверов ВКонтакте полученный от DNS в пакете №3))
- Протокол: ICMP
- **Тип запроса**: Echo (ping) request – запрос "ping" (проверка доступности хоста)
- **ID запроса**: 0x0002
- **seq: 1/256** – Номер последовательности (1-й пакет в серии)
- **ttl: 64** – Time To Live (Максимальное число прыжков, которое может пройти пакет. 64 это стандартное значение для Linux/macOS, 128 для Windows)
- **(reply in 6)** – Ссылка на ответ (пакете №6)

Пакет 6: Ответ на предыдущий запрос

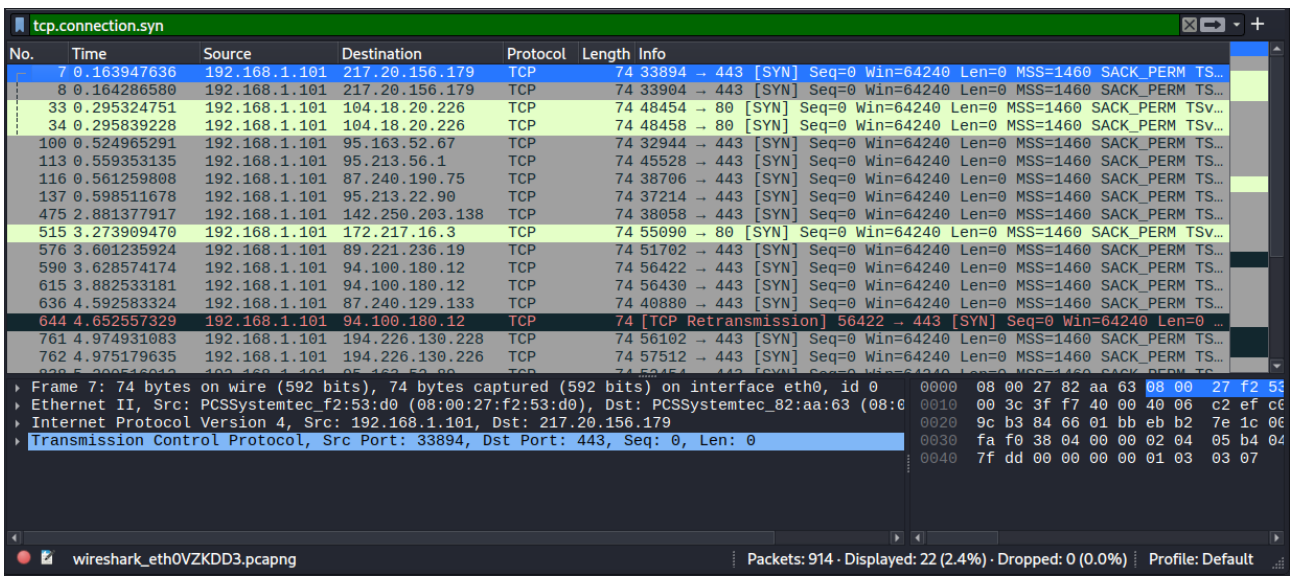
- **ttl: 54** – пакет прошёл 10 прыжков (64 - 54 = 10).

3.2 Начать захват трафика, перейти в браузере на страницу vk.com, остановить захват трафика в wireshark и найти, где устанавливается TCP connect (3-х стороннее рукопожатие):

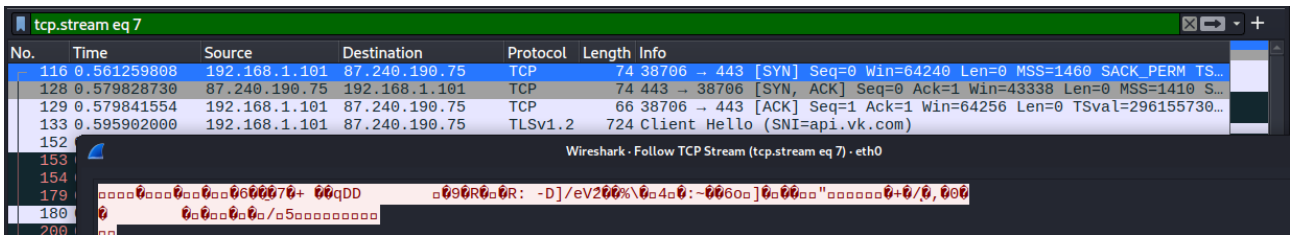
В процессе перехода на сайт vk.com и загрузки страницы входа, wireshark перехватывает 100500+ пакетов:



Используем специальный встроенный семантический фильтр **tcp.connection.syn** – специально для поиска начала TCP-соединений. Wireshark автоматически найдёт и отобразит первые пакеты (SYN) из каждой тройки «рукопожатий»:



Выберем пакет отправленный на адрес 87.240.190.75, правой кнопкой мыши откроем контекстное меню и далее → Follow → TCP Stream. Wireshark автоматически выделит все пакеты этого соединения, включая 3-way handshake:



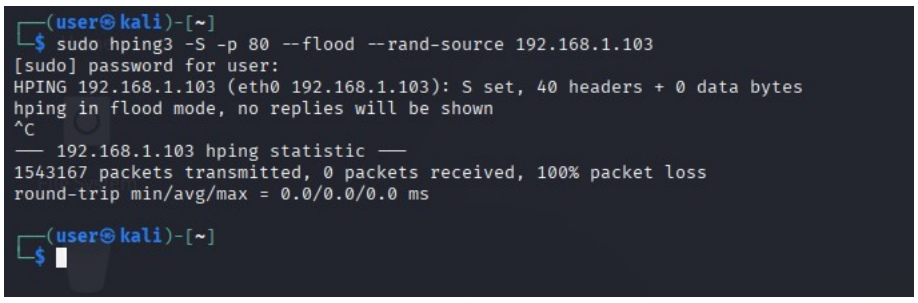
3.3 Начать захват трафика, провести Tcp SYN FLOOD с BM Kali на BM Metasploitable, используя утилиту hping3, отловить пакеты в Wireshark и описать, как работает эта атака.

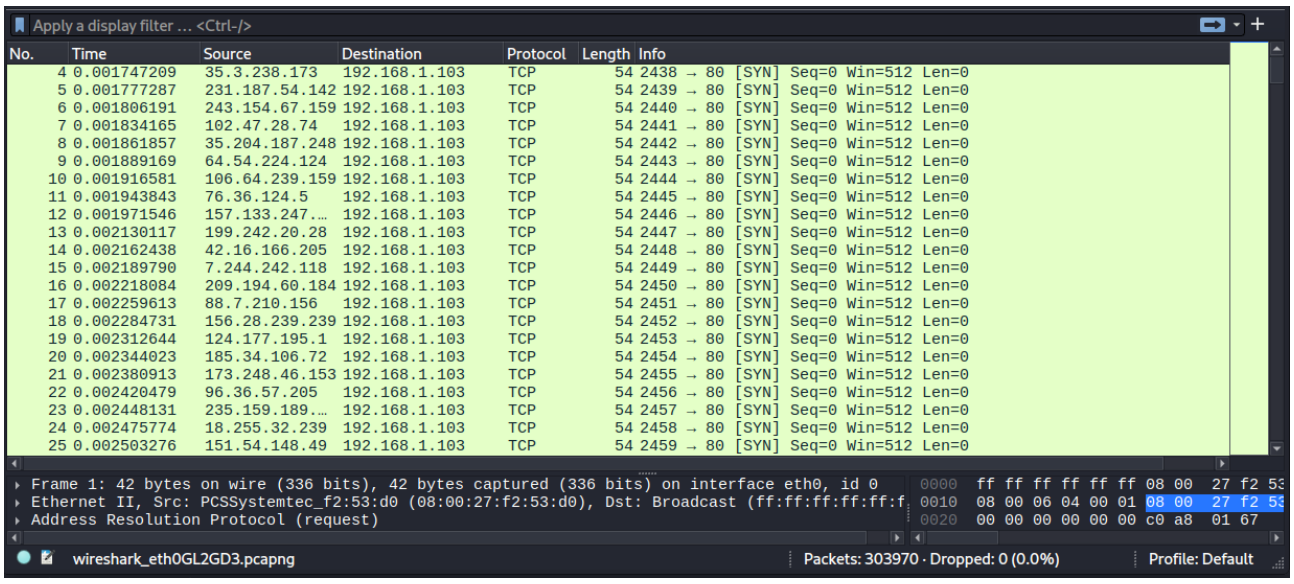
TCP SYN Flood — это вид DoS-атаки, при которой злоумышленник отправляет огромное количество TCP-пакетов с флагом SYN на целевой сервер, не завершая рукопожатие. Сервер выделяет ресурсы под каждое "полуоткрытое" соединение, исчерпывает память и перестаёт отвечать легитимным клиентам.

Пример атаки BM metasploitable (IP 192.168.1.103), на которой запущен веб-сервер (порт 80):

```
sudo hping3 -S -p 80 --flood --rand-source 192.168.1.103
```

→ hping3 начнёт слать SYN-пакеты с поддельных IP на порт 80 цели:





4. Cisco Packet Tracer

Настроить сеть локальную сеть: 2 switch, 4 компьютера, 2 vlan (по 2 компьютера в каждой), открыть эмуляцию терминала для одного из компьютеров и выполнить команду ping к компьютеру в одном vlan и в другом, прислать скрин схемы сети и терминала с выполненными командами.