

Отчёт по лабораторной работе 5

Простые сети в GNS3. Анализ трафика

Заур Мустафаев

Содержание

1 Цель работы	5
2 Выполнение работы	6
2.1 Создание топологии сети в GNS3	6
2.2 Настройка IP-адресов узлов VPCS	7
2.3 Анализ трафика ARP в Wireshark	9
2.4 Анализ ICMP-трафика	10
2.5 Анализ UDP-трафика	11
2.6 Анализ TCP-трафика	12
2.7 Моделирование простейшей сети на базе маршрутизатора FRR в GNS3	14
2.7.1 Создание топологии	14
2.7.2 Настройка IP-адресации на узле PC1	14
2.7.3 Настройка маршрутизатора FRR	15
2.7.4 Проверка соединения	16
2.7.5 Анализ ICMP-трафика в Wireshark	17
2.8 Моделирование простейшей сети на базе маршрутизатора VyOS в GNS3	18
2.8.1 Создание топологии	18
2.8.2 Настройка маршрутизатора VyOS	19
2.8.3 Проверка соединения	20
2.8.4 Анализ трафика в Wireshark	21
3 Вывод	22

Список иллюстраций

2.1 Созданная топология в GNS3	7
2.2 Назначение IP-адреса для PC1	8
2.3 Проверка соединения между узлами	9
2.4 Захват ARP-трафика в Wireshark	10
2.5 ICMP Echo Request и Reply в Wireshark	11
2.6 UDP Echo Request и Response в Wireshark	12
2.7 Настройка маршрутизатора FRR	15
2.8 Проверка интерфейсов FRR	16
2.9 Успешная проверка связи с маршрутизатором FRR	17
2.10 ICMP-трафик между PC1 и FRR	18
2.11 Настройка маршрутизатора VyOS	19
2.12 Проверка интерфейсов маршрутизатора VyOS	20
2.13 Успешный обмен ICMP-пакетами с маршрутизатором VyOS	20
2.14 Анализ ICMP и ARP-трафика между PC1 и VyOS	21

Список таблиц

1 Цель работы

Построение простейших моделей сети на базе коммутатора и маршрутизаторов FRR и VyOS в GNS3, анализ трафика посредством Wireshark.

2 Выполнение работы

2.1 Создание топологии сети в GNS3

1. В среде **GNS3** был создан новый проект.

На рабочем поле размещены два устройства **VPCS** и один **Ethernet switch**.

Для всех узлов были изменены имена с указанием фамилии пользователя:

- **PC1-zmustafaev**
- **msk-zmustafaev-sw-01**
- **PC2-zmustafaev**

Узлы соединены между собой с помощью Ethernet-соединений, что обеспечило логическую топологию “звезда”.

Интерфейсы соединений отображены на схеме.

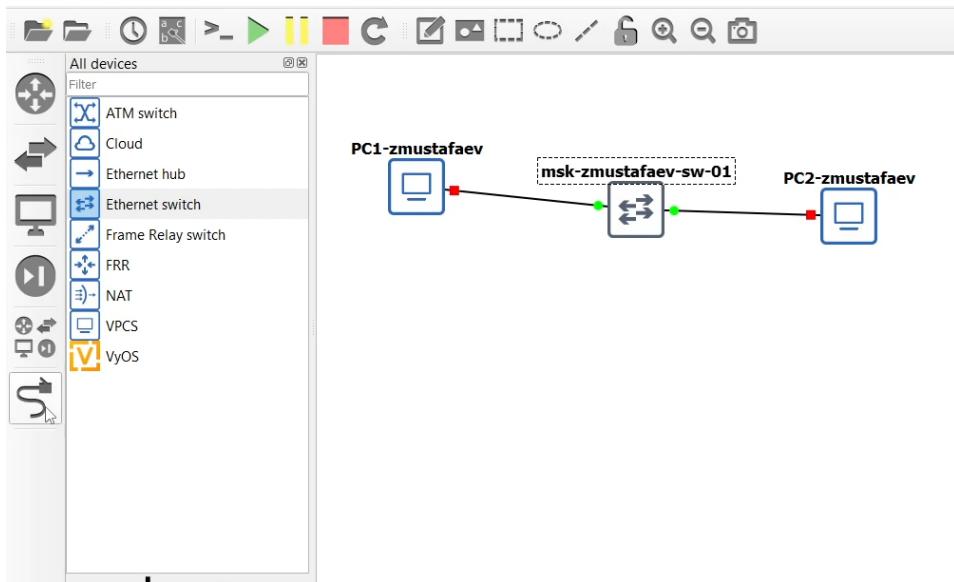


Рис. 2.1: Созданная топология в GNS3

2.2 Настройка IP-адресов узлов VPCS

2. На узле **PC1-zmustafaev** с помощью терминала **Console** выполнена настройка IP-адреса и шлюза по умолчанию.

Для просмотра доступных команд введена команда `?`.

Затем выполнена настройка IP: `ip 192.168.1.11/24 192.168.1.1`

После проверки отсутствия дубликатов адресов, система подтвердила успешное назначение IP.

```
PC1-zmustafaev - PuTTY
Executing the startup file
Hostname is too long. (Maximum 12 characters)

VPCS> ?

?
arp Print help
clear ARG Shortcut for: show arp. Show arp table
dhcp [OPTION] Clear IPv4/IPv6, arp/neighbor cache, command history
disconnect Shortcut for: ip dhcp. Get IPv4 address via DHCP
echo TEXT Exit the telnet session (daemon mode)
help Print help
history Shortcut for: show history. List the command history
ip ARG ... [OPTION] Configure the current VPC's IP settings. See ip ?
load [FILENAME] Load the configuration/script from the file FILENAME
ping HOST [OPTION ...] Ping HOST with ICMP (default) or TCP/UDP. See ping ?
quit Quit program
relay ARG ... Configure packet relay between UDP ports. See relay ?
rlogin [ip] port Telnet to port on host at ip (relative to host PC)
save [FILENAME] Save the configuration to the file FILENAME
set ARG ... Set VPC name and other options. Try set ?
show [ARG ...] Print the information of VPCs (default). See show ?
sleep [seconds] [TEXT] Print TEXT and pause running script for seconds
trace HOST [OPTION ...] Print the path packets take to network HOST
version Shortcut for: show version

To get command syntax help, please enter '?' as an argument of the command.

VPCS> ip 192.168.1.11/24 192.168.1.1
Checking for duplicate address...
VPCS : 192.168.1.11 255.255.255.0 gateway 192.168.1.1
```

Рис. 2.2: Назначение IP-адреса для PC1

3. Аналогичные действия были выполнены для **PC2-zmustafaev**, которому назначен адрес:

ip 192.168.1.12/24 192.168.1.1

save

После сохранения конфигурации проверено соединение между узлами командой **ping 192.168.1.11**.

Эхо-запросы прошли успешно, что подтверждает работоспособность сети.

```
PC2-zmustafaev - PuTTY
Source code and license can be found at vpcs.sf.net.
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file

Hostname is too long. (Maximum 12 characters)

VPCS> ip 192.168.1.12/24 192.168.1.1
Checking for duplicate address...
VPCS : 192.168.1.12 255.255.255.0 gateway 192.168.1.1

VPCS> save
Saving startup configuration to startup.vpc
. done

VPCS>
VPCS> ping 192.168.1.11

84 bytes from 192.168.1.11 icmp_seq=1 ttl=64 time=0.231 ms
84 bytes from 192.168.1.11 icmp_seq=2 ttl=64 time=1.033 ms
84 bytes from 192.168.1.11 icmp_seq=3 ttl=64 time=0.677 ms
```

Рис. 2.3: Проверка соединения между узлами

2.3 Анализ трафика ARP в Wireshark

- На соединении между **PC1** и коммутатором был запущен захват трафика с помощью **Wireshark**.

После запуска узлов в окне анализа отобразились **ARP-пакеты**, используемые для определения MAC-адресов соседей в локальной сети.

В захваченных данных видны **Gratuitous ARP** и **ARP Request**, отправленные с IP 192.168.1.12 и 192.168.1.11 соответственно.

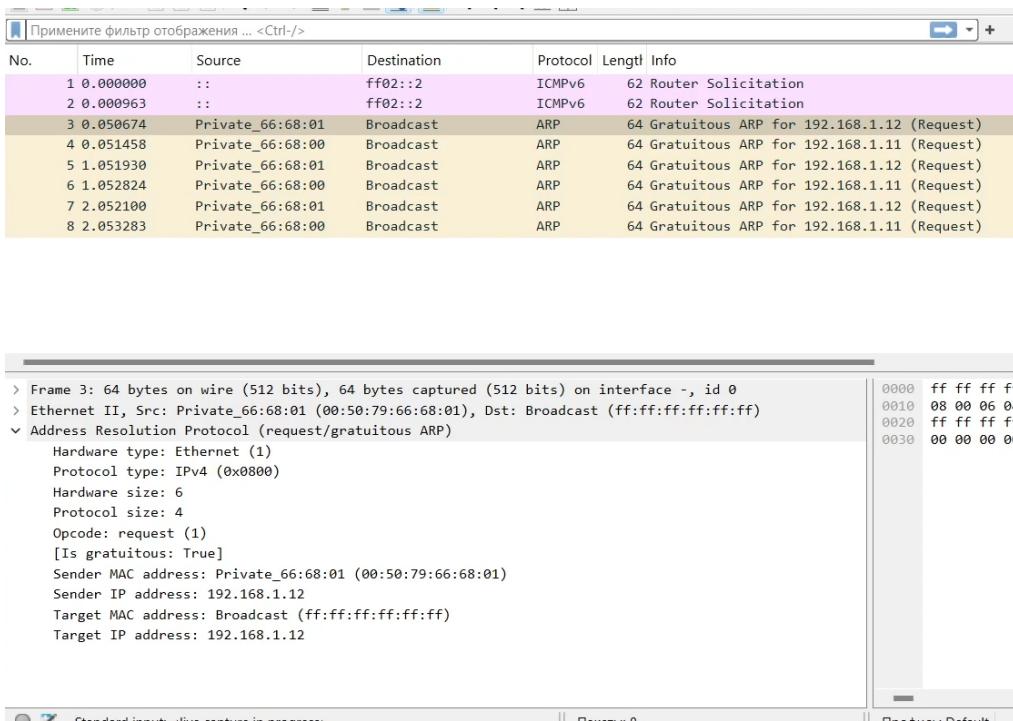


Рис. 2.4: Захват ARP-трафика в Wireshark

2.4 Анализ ICMP-трафика

5. На **PC2** были изучены параметры команды `ping /?`, после чего выполнен один эхо-запрос к **PC1** в **ICMP-режиме**:

```
ping 192.168.1.11 -c 1
```

Запрос прошёл успешно, что подтверждается ответом `ttl=64`.

В **Wireshark** зафиксированы ICMP Echo Request и Echo Reply между адресами 192.168.1.12 и 192.168.1.11.

```

-c count          Packet count, default 5
-D                Set the Don't Fragment bit
-f FLAG          Tcp header FLAG |C|E|U|A|P|R|S|F|
                   bits |7 6 5 4 3 2 1 0|
-i ms            Wait ms milliseconds between sending each packet
-l size          Data size
-P protocol     Use IP protocol in ping packets
                   1 - ICMP (default), 17 - UDP, 6 - TCP
-p port          Destination port
-s port          Source port
-T ttl           Set ttl, default 64
-t                Send packets until interrupted by Ctrl+C
-w ms            Wait ms milliseconds to receive the response

Notes: 1. Using names requires DNS to be set.
       2. Use Ctrl+C to stop the command.

VPCS> ping 192.168.1.11 -c 1
84 bytes from 192.168.1.11 icmp_seq=1 ttl=64 time=0.400 ms
VPCS> ping 192.168.1.11 -c 1 -2
84 bytes from 192.168.1.11 udp_seq=1 ttl=64 time=0.458 ms
VPCS> ping 192.168.1.11 -c 1 -3
Connect    7@192.168.1.11 seq=1 ttl=64 time=1.364 ms
SendData   7@192.168.1.11 seq=1 ttl=64 time=1.331 ms
Close      7@192.168.1.11 seq=1 ttl=64 time=2.609 ms
VPCS>

```

Рис. 2.5: ICMP Echo Request и Reply в Wireshark

2.5 Анализ UDP-трафика

6. Выполнен **UDP-эхо-запрос**:

```
ping 192.168.1.11 -c 1 -2
```

В **Wireshark** отображены пакеты с **протоколом UDP**, использующим **порт 7 (Echo)**.

Видно, что PC2 отправил запрос с порта 32551 на порт 7 узла PC1, после чего был получен ответ.

No.	Time	Source	Destination	Protocol	Length	Info
5	1.051930	Private_66:68:01	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.12 (Request)
6	1.052824	Private_66:68:00	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.11 (Request)
7	2.052100	Private_66:68:01	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.12 (Request)
8	2.053283	Private_66:68:00	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.11 (Request)
9	109.469194	Private_66:68:01	Broadcast	ARP	64	Who has 192.168.1.11? Tell 192.168.1.12
10	109.469567	Private_66:68:01	Private_66:68:01	ARP	64	192.168.1.11 is at 00:50:79:66:68:00
→ 11	109.470659	192.168.1.12	192.168.1.11	ICMP	98	Echo (ping) request id=0xb23, seq=1/256,
← 12	109.470882	192.168.1.11	192.168.1.12	ICMP	98	Echo (ping) reply id=0xb23, seq=1/256,
13	125.589410	192.168.1.12	192.168.1.11	ECHO	98	Request
14	125.589631	192.168.1.11	192.168.1.12	ECHO	98	Response
15	133.717388	192.168.1.12	192.168.1.11	TCP	74	51909 → 7 [SYN] Seq=0 Win=2920 Len=0 MSS=1
16	133.717720	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [SYN, ACK] Seq=0 Ack=1 Win=2920
17	133.718814	192.168.1.12	192.168.1.11	TCP	66	51909 → 7 [ACK] Seq=1 Ack=1 Win=2920 Len=0
18	133.718894	192.168.1.12	192.168.1.11	FCHO	122	Request

> Frame 11: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface -, id 0
 > Ethernet II, Src: Private_66:68:01 (00:50:79:66:68:01), Dst: Private_66:68:00 (00:50:79:66:68:00)
 > Internet Protocol Version 4, Src: 192.168.1.12, Dst: 192.168.1.11
 > Internet Control Message Protocol
 Type: 8 (Echo (ping) request)
 Code: 0
 Checksum: 0x94e7 [correct]
 [Checksum Status: Good]
 Identifier (BE): 35619 (0xb23)
 Identifier (LE): 9099 (0x23b)
 Sequence Number (BE): 1 (0x0001)
 Sequence Number (LE): 256 (0x0100)
 [Response frame: 12]
 Data (56 bytes)

0000 00 50 79 66
 0010 00 54 23 88
 0020 01 0b 08 00
 0030 0e 0f 10 12
 0040 1e 1f 20 22
 0050 2e 2f 30 32
 0060 3e 3f

Рис. 2.6: UDP Echo Request и Response в Wireshark

2.6 Анализ TCP-трафика

7. Для проверки TCP-взаимодействия между узлами выполнена команда:

```
ping 192.168.1.11 -c 1 -3
```

В **Wireshark** отображена трёхфазная установка соединения (**SYN** → **SYN/ACK** → **ACK**) и последующий обмен TCP-пакетами.

Также виден корректный **FIN/ACK** при завершении сессии.

No.	Time	Source	Destination	Protocol	Length	Info
11	109.470659	192.168.1.12	192.168.1.11	ICMP	98	Echo (ping) request id=0x8b23, seq=1/256,
12	109.470882	192.168.1.11	192.168.1.12	ICMP	98	Echo (ping) reply id=0x8b23, seq=1/256,
13	125.589410	192.168.1.12	192.168.1.11	ECHO	98	Request
14	125.589631	192.168.1.11	192.168.1.12	ECHO	98	Response
15	133.717388	192.168.1.12	192.168.1.11	TCP	74	51909 → 7 [SYN] Seq=0 Win=2920 Len=0 MSS=1460
16	133.717720	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [SYN, ACK] Seq=0 Ack=1 Win=2920
17	133.718814	192.168.1.12	192.168.1.11	TCP	66	51909 → 7 [ACK] Seq=1 Ack=1 Win=2920 Len=0
18	133.718894	192.168.1.12	192.168.1.11	ECHO	122	Request
19	133.719408	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [ACK] Seq=1 Ack=57 Win=2920 Len=0
20	133.720635	192.168.1.12	192.168.1.11	TCP	66	51909 → 7 [FIN, PSH, ACK] Seq=57 Ack=1 Win=2920
21	133.720738	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [ACK] Seq=1 Ack=58 Win=2920 Len=0
22	133.720749	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [FIN, ACK] Seq=1 Ack=58 Win=2920
23	133.723289	192.168.1.12	192.168.1.11	TCP	66	51909 → 7 [ACK] Seq=58 Ack=2 Win=2920 Len=0

> Frame 13: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface -, id 0
> Ethernet II, Src: Private_66:68:01 (00:50:79:66:68:01), Dst: Private_66:68:00 (00:50:79:66:68:00)
> Internet Protocol Version 4, Src: 192.168.1.12, Dst: 192.168.1.11
< User Datagram Protocol, Src Port: 32551, Dst Port: 7
 Source Port: 32551
 Destination Port: 7
 Length: 64
 Checksum: 0x614d [unverified]
 [Checksum Status: Unverified]
 [Stream index: 0]
 [Stream Packet Number: 1]
 > [Timestamps]
 UDP payload (56 bytes)
< Echo
 Echo data: 0050796668010e0f101112131415161718191a1b1c1d1e1f202122232425262728292a2b2c2d2e2f3031323...

0000	00 50 79 66
0010	00 54 23 a0
0020	01 0b 7f 2
0030	0e 0f 10 1
0040	1e 1f 20 2
0050	2e 2f 30 3
0060	3e 3f

Standard input: <live capture in progress> || Пакеты: 23 || Профиль: Default

18	133.718894	192.168.1.12	192.168.1.11	ECHO	122	Request
19	133.719408	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [ACK] Seq=1 Ack=57 Win=2920 Len=0
20	133.720635	192.168.1.12	192.168.1.11	TCP	66	51909 → 7 [FIN, PSH, ACK] Seq=57 Ack=1 Win=2920
21	133.720738	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [ACK] Seq=1 Ack=58 Win=2920 Len=0
22	133.720749	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [FIN, ACK] Seq=1 Ack=58 Win=2920
23	133.723289	192.168.1.12	192.168.1.11	TCP	66	51909 → 7 [ACK] Seq=58 Ack=2 Win=2920 Len=0

> Transmission Control Protocol, Src Port: 7, Dst Port: 51909, Seq: 1, Ack: 58, Len: 0
 Source Port: 7
 Destination Port: 51909
 [Stream index: 0]
 [Stream Packet Number: 7]
 > [Conversation completeness: Complete, WITH_DATA (31)]
 [TCP Segment Len: 0]
 Sequence Number: 1 (relative sequence number)
 Sequence Number (raw): 1978392413
 [Next Sequence Number: 1 (relative sequence number)]
 Acknowledgment Number: 58 (relative ack number)
 Acknowledgment number (raw): 519200666
 0101 = Header Length: 20 bytes (5)
 > Flags: 0x010 (ACK)
 Window: 2920
 [Calculated window size: 2920]
 [Window size scaling factor: -2 (no window scaling used)]
 Checksum: 0xef59 [unverified]
 [Checksum Status: Unverified]
 Urgent Pointer: 0
 > [Timestamps]
 > [SEQ/ACK analysis]

0000	00 50 79 66
0010	00 28 23 a0
0020	01 0c 00 01
0030	0b 68 ef 59

2.7 Моделирование простейшей сети на базе маршрутизатора FRR в GNS3

2.7.1 Создание топологии

1. В среде **GNS3** был создан новый проект.

В рабочей области размещены три устройства:

- **PC1-zmustafaev**
- **msk-zmustafaev-sw-01**
- **msk-zmustafaev-gw-01**

Все устройства соединены через коммутатор, что позволило сформировать простейшую сеть с одним маршрутизатором.

2.7.2 Настройка IP-адресации на узле PC1

2. Для устройства **PC1-zmustafaev** задан IP-адрес, шлюз и сохранена конфигурация.

Проверка текущих параметров сети выполнена командой `show ip`.

Назначенные параметры:

- **IP-адрес:** 192.168.1.10/24
- **Шлюз:** 192.168.1.1

```

PC1-zmustafaev - PuTTY

Checking for duplicate address...
VPCS : 192.168.1.11 255.255.255.0 gateway 192.168.1.1

VPCS> ip 192.168.1.10/24 192.168.1.1
Checking for duplicate address...
VPCS : 192.168.1.10 255.255.255.0 gateway 192.168.1.1

VPCS> save
Saving startup configuration to startup.vpc
. done

VPCS> show ip

NAME      : VPCS[1]
IP/MASK   : 192.168.1.10/24
GATEWAY   : 192.168.1.1
DNS       :
MAC       : 00:50:79:66:68:00
LPORT     : 10004
RHOST:PORT : 127.0.0.1:10005
MTU       : 1500

VPCS>

```

2.7.3 Настройка маршрутизатора FRR

3. На маршрутизаторе **msk-zmustafaev-gw-01** выполнена базовая конфигурация.

Установлено имя узла, активирован интерфейс **eth0**, которому присвоен адрес **192.168.1.1/24**, и включено его состояние (**no shutdown**).

Конфигурация сохранена в **/etc/frr/frr.conf**.

```

msk-zmustafaev-gw-01 - PuTTY

frr# hostname msk-zmustafaev-gw-01
% Unknown command: hostname msk-zmustafaev-gw-01
frr# configure terminal
frr(config)# hostname msk-zmustafaev-gw-01
msk-zmustafaev-gw-01(config)# exit
msk-zmustafaev-gw-01# write memory
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Integrated configuration saved to /etc/frr/frr.conf
[OK]
msk-zmustafaev-gw-01# configure terminal
msk-zmustafaev-gw-01(config)# interface eth0
msk-zmustafaev-gw-01(config-if)# ip address 192.168.1.1/24
msk-zmustafaev-gw-01(config-if)# no shutdown
msk-zmustafaev-gw-01(config-if)# exit
msk-zmustafaev-gw-01(config)# exit
msk-zmustafaev-gw-01# write memory
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Integrated configuration saved to /etc/frr/frr.conf
[OK]
msk-zmustafaev-gw-01#
msk-zmustafaev-gw-01# show running-config

```

Рис. 2.7: Настройка маршрутизатора FRR

4. Проверка конфигурации показала, что интерфейс **eth0** находится в состоянии **up** и имеет адрес **192.168.1.1/24**.

```
hostname frr
hostname msk-zmustafaev-gw-01
service integrated-vtysh-config
!
interface eth0
 ip address 192.168.1.1/24
exit
tc!
end
msk-zmustafaev-gw-01# show interface brief
Interface      Status   VRF   Addresses
-----  -----  ---  -----
eth0        up      default  192.168.1.1/24
eth1        down    default
eth2        down    default
eth3        down    default
eth4        down    default
eth5        down    default
eth6        down    default
eth7        down    default
lo          up      default
pimreg      up      default
msk-zmustafaev-gw-01#
```

Рис. 2.8: Проверка интерфейсов FRR

2.7.4 Проверка соединения

5. С узла **PC1-zmustafaev** выполнена проверка связи с маршрутизатором:

```
ping 192.168.1.1
```

Все пакеты успешно получены, что подтверждает корректность IP-настройки и функционирование маршрутизации.

```
PC1-zmustafaev - PuTTY
bits |7 6 5 4 3 2 1 0|
-i ms      Wait ms milliseconds between sending each packet
-l size     Data size
-P protocol Use IP protocol in ping packets
             1 - ICMP (default), 17 - UDP, 6 - TCP
-p port     Destination port
-s port     Source port
-T ttl      Set ttl, default 64
-t          Send packets until interrupted by Ctrl+C
-w ms      Wait ms milliseconds to receive the response

Notes: 1. Using names requires DNS to be set.
       2. Use Ctrl+C to stop the command.

VPCS> I
VPCS> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=2.692 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=1.311 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=2.059 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=1.205 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=2.205 ms

VPCS> █
```

Рис. 2.9: Успешная проверка связи с маршрутизатором FRR

2.7.5 Анализ ICMP-трафика в Wireshark

6. В **Wireshark** был запущен захват пакетов на канале между коммутатором и маршрутизатором.

На графике видны **ICMP Echo Request** и **ICMP Echo Reply** между IP-адресами 192.168.1.10 и 192.168.1.1, что свидетельствует о корректной работе сетевого обмена на уровне ICMP.

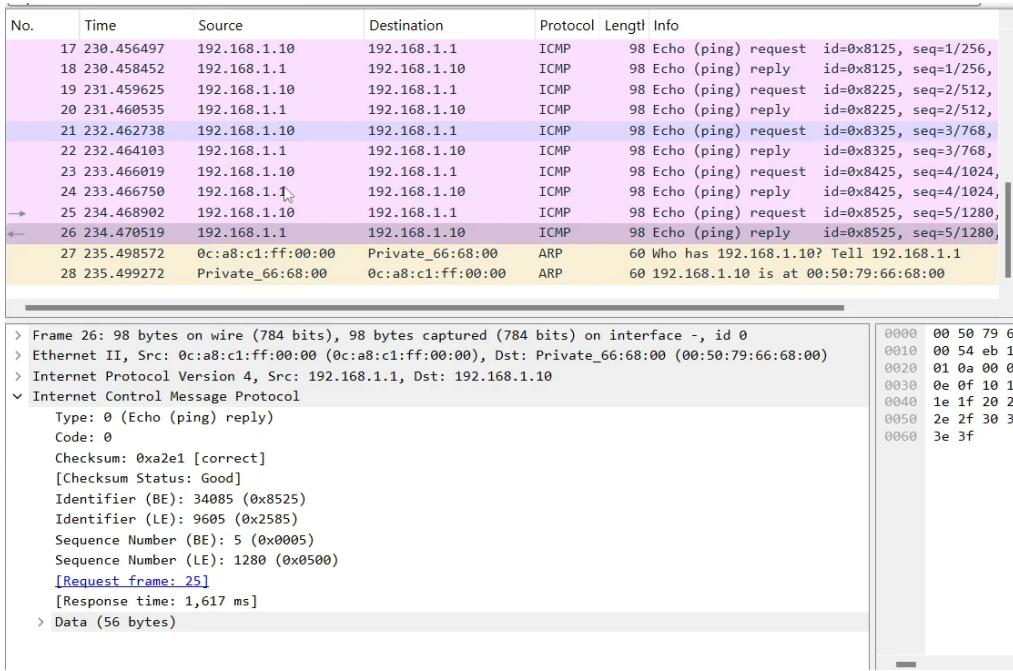


Рис. 2.10: ICMP-трафик между PC1 и FRR

2.8 Моделирование простейшей сети на базе маршрутизатора VyOS в GNS3

2.8.1 Создание топологии

1. В среде **GNS3** создана аналогичная топология с маршрутизатором **VyOS**.

В схему включены устройства:

- **PC1-zmustafaev**

- **msk-zmustafaev-sw-01**

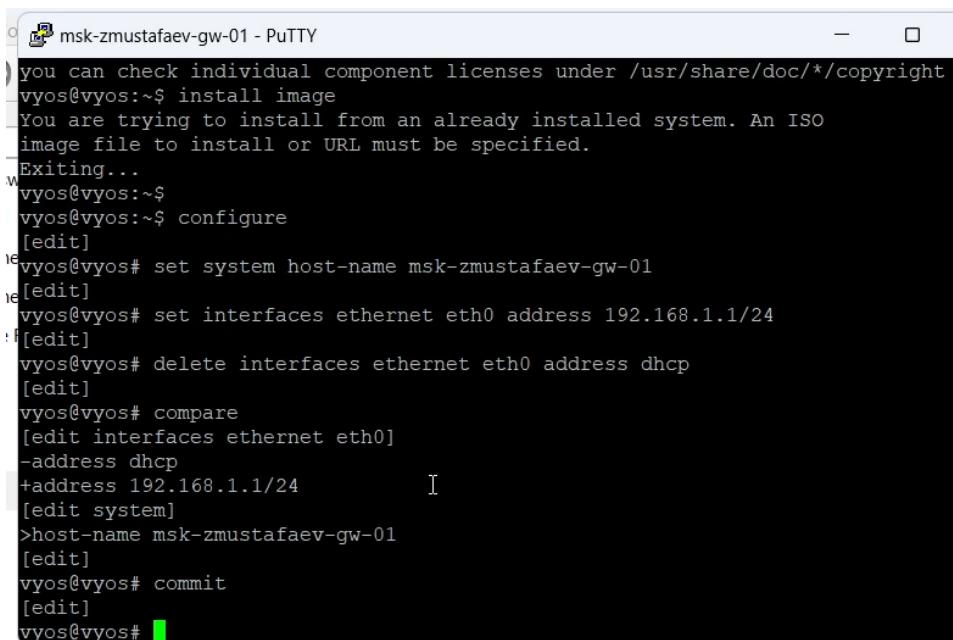
- **msk-zmustafaev-gw-01 (VyOS)**

Устройства соединены через коммутатор Ethernet.

2.8.2 Настройка маршрутизатора VyOS

2. После входа под пользователем **vyos** выполнена настройка маршрутизатора:

- Изменено имя устройства: `msk-zmustafaev-gw-01`
- Задан IP-адрес интерфейса **eth0**: `192.168.1.1/24`
- Удалён DHCP-адрес с интерфейса
- Конфигурация сохранена и применена командами `commit` и `save`



```
you can check individual component licenses under /usr/share/doc/*copyright
vyos@vyos:~$ install image
You are trying to install from an already installed system. An ISO
image file to install or URL must be specified.
Exiting...
vyos@vyos:~$ configure
[edit]
vyos@vyos# set system host-name msk-zmustafaev-gw-01
[edit]
vyos@vyos# set interfaces ethernet eth0 address 192.168.1.1/24
[edit]
vyos@vyos# delete interfaces ethernet eth0 address dhcp
[edit]
vyos@vyos# compare
[edit interfaces ethernet eth0]
-address dhcp
+address 192.168.1.1/24
[edit system]
>host-name msk-zmustafaev-gw-01
[edit]
vyos@vyos# commit
[edit]
vyos@vyos#
```

Рис. 2.11: Настройка маршрутизатора VyOS

3. Проверка конфигурации маршрутизатора показала наличие активного интерфейса **eth0** с назначенным IP-адресом **192.168.1.1/24**.

```
[edit]
vyos@vyos# commit
[edit]
vyos@vyos# save
Saving configuration to '/config/config.boot'...
Done
[edit]
vyos@vyos# show interfaces
+-----+
ethernet eth0 {
    address 192.168.1.1/24
    hw-id 0c:77:69:cf:00:00
}
ethernet eth1 {
    hw-id 0c:77:69:cf:00:01
}
ethernet eth2 {
    hw-id 0c:77:69:cf:00:02
}
loopback lo {
}
[edit]
vyos@vyos# exit
exit
vyos@vyos:~$
```

Рис. 2.12: Проверка интерфейсов маршрутизатора VyOS

2.8.3 Проверка соединения

- На узле **PC1-zmustafaev** выполнена команда `ping 192.168.1.1`.

Все эхо-запросы успешно доставлены, что подтверждает корректную работу маршрутизатора VyOS и связь между устройствами.

```
VPCS> ping 192.168.1.1
84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=3.789 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=2.306 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=2.276 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=1.114 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=2.472 ms
VPCS>
```

Рис. 2.13: Успешный обмен ICMP-пакетами с маршрутизатором VyOS

2.8.4 Анализ трафика в Wireshark

5. При анализе пакетов с помощью **Wireshark** зафиксированы обмены ICMP-запросами и ответами между узлами **192.168.1.10** и **192.168.1.1**, а также ARP-запросы, направленные на определение MAC-адресов узлов. Это подтверждает правильную работу стека протоколов и успешное взаимодействие устройств.

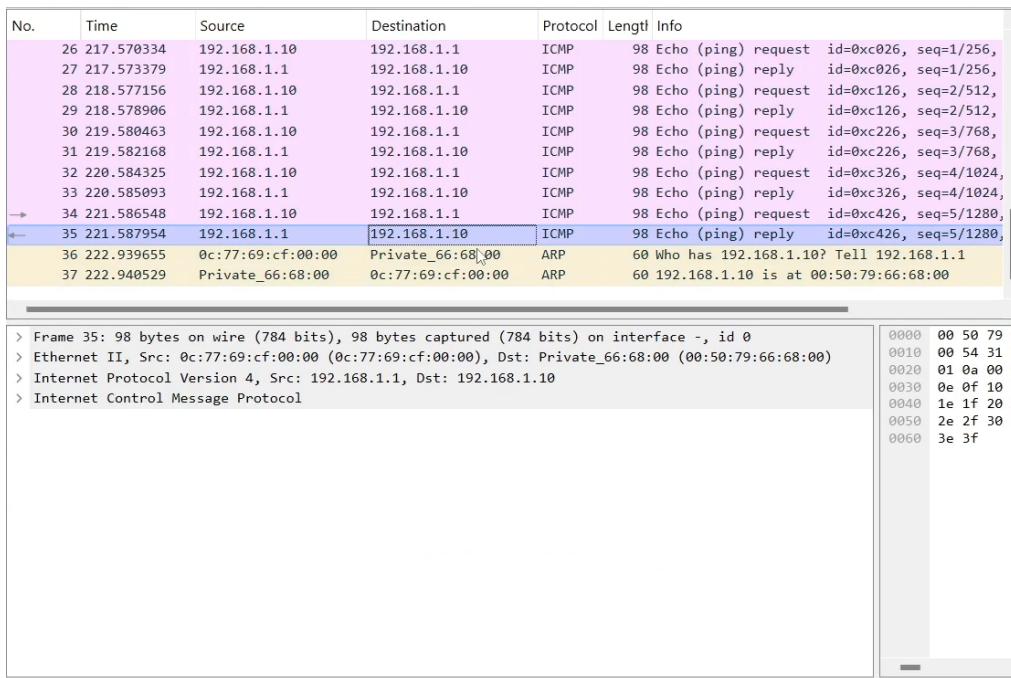


Рис. 2.14: Анализ ICMP и ARP-трафика между PC1 и VyOS

3 Вывод

В результате лабораторной работы были смоделированы две простейшие сети на базе маршрутизаторов **FRR** и **VyOS**.

Для обеих схем успешно выполнена IP-настройка, проверена связность и проведён анализ ICMP- и ARP-трафика.

Результаты подтверждают корректную работу сетевой инфраструктуры, функционирование протоколов нижних уровней и правильную конфигурацию маршрутизаторов.