

Отчёт по лабораторной работе №5

Простые сети в GNS3. Анализ трафика

Владимир Базлов

Содержание

1 Цель работы	5
2 Выполнение задания	6
2.1 Построение простой сети в GNS3 и анализ трафика	6
2.1.1 Настройка IPv4-адресов на PC-1 и PC-2	6
2.1.2 Настройка PC1-vabazlov	7
2.1.3 Настройка PC2-vabazlov	7
2.2 Анализ ARP-трафика в Wireshark	8
2.2.1 Анализ ICMP-трафика	9
2.2.2 UDP-эхо-запрос	11
2.2.3 TCP-эхо-запрос	12
2.3 Построение сети с маршрутизатором FRR в GNS3 и анализ трафика	13
2.4 Построение сети с маршрутизатором VyOS	16
2.5 Настройка PC1	17
2.6 Настройка маршрутизатора VyOS	17
2.7 Проверка работоспособности сети	19
3 Заключение	20

Список иллюстраций

2.1 Топология сети	6
2.2 Настройка PC1	7
2.3 Настройка PC2	8
2.4 ARP-захват	9
2.5 Ping options	10
2.6 ICMP-анализ	11
2.7 UDP-анализ	12
2.8 Топология сети	13
2.9 Настройка PC1	14
2.10 Настройка FRR	14
2.11 Конфигурация FRR	15
2.12 Интерфейсы FRR	15
2.13 Пинг FRR	16
2.14 ICMP-трафик FRR	16
2.15 Топология сети VyOS	17
2.16 Настройка VyOS	18
2.17 Интерфейсы VyOS	18
2.18 Пинг VyOS	19
2.19 ICMP-трафик VyOS	19

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 задания

2.1 Построение простой сети в GNS3 и анализ трафика

В GNS3 создан новый проект, после чего в рабочей области размещены два узла VPCS и коммутатор Ethernet. Всем устройствам назначены имена, содержащие имя пользователя, согласно требованиям задания. Коммутатору присвоено имя msk-vabazlov-sw-01, а VPCS – PC1-vabazlov и PC2-vabazlov.

Топология после соединения устройств представлена на рисунке ниже.



Рис. 2.1: Топология сети

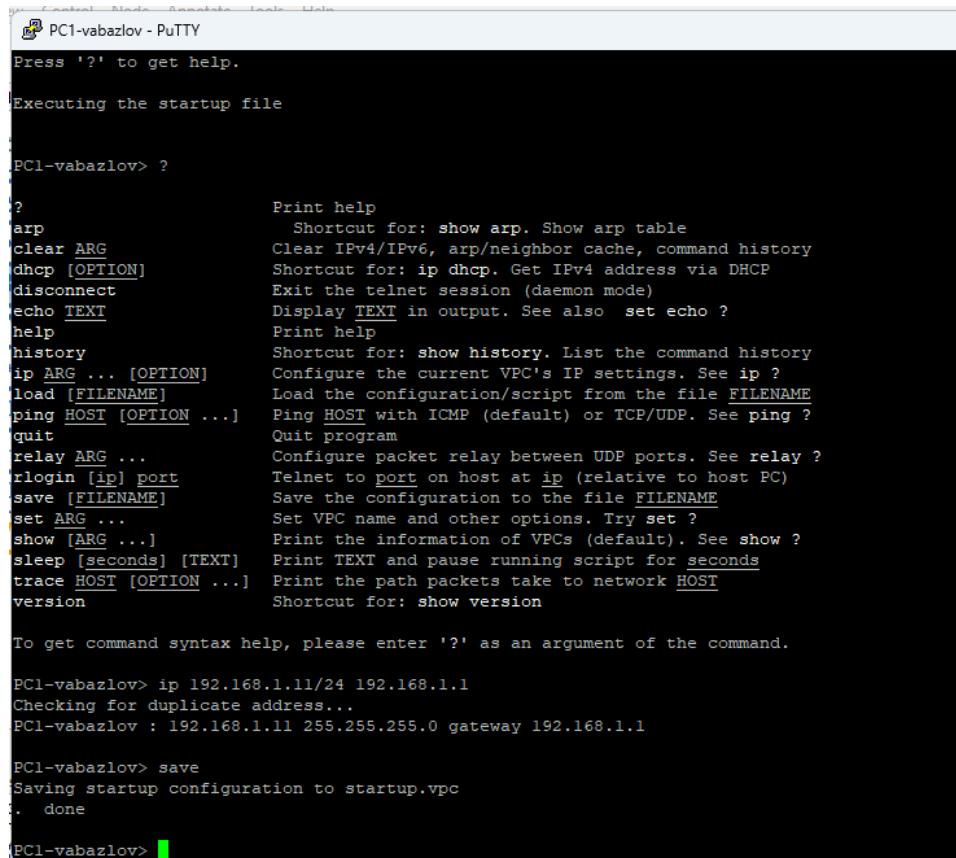
Интерфейсы соединений отображены в соответствии с требованиями лабораторной работы.

2.1.1 Настройка IPv4-адресов на РС-1 и РС-2

После запуска VPCS через Start и открытия консоли командой Console была выполнена настройка IP-адресации.

2.1.2 Настройка PC1-vabazlov

PC-1 получил IP-адрес 192.168.1.11/24 и шлюз 192.168.1.1. Команда для просмотра синтаксиса ip /? использована перед вводом параметров.



```
PC1-vabazlov - PuTTY
Press '?' to get help.

Executing the startup file

PC1-vabazlov> ?

?
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 Display TEXT in output. See also set echo ?
history Print help
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.

PC1-vabazlov> ip 192.168.1.11/24 192.168.1.1
Checking for duplicate address...
PC1-vabazlov : 192.168.1.11 255.255.255.0 gateway 192.168.1.1

PC1-vabazlov> save
Saving startup configuration to startup.vpc
. done

PC1-vabazlov>
```

Рис. 2.2: Настройка PC1

2.1.3 Настройка PC2-vabazlov

PC-2 был настроен с адресом 192.168.1.12/24 и шлюзом 192.168.1.1. После сохранения конфигурации выполнено тестирование доступности PC-1, пинг прошёл успешно.

```
PC2-vabazlov - PuTTY
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file

PC2-vabazlov> ip 192.168.1.12/24 192.168.1.1
Checking for duplicate address...
PC2-vabazlov : 192.168.1.12 255.255.255.0 gateway 192.168.1.1

PC2-vabazlov> save
Saving startup configuration to startup.vpc
. done

PC2-vabazlov> ping 192.168.1.11

84 bytes from 192.168.1.11 icmp_seq=1 ttl=64 time=1.250 ms
84 bytes from 192.168.1.11 icmp_seq=2 ttl=64 time=1.255 ms
84 bytes from 192.168.1.11 icmp_seq=3 ttl=64 time=0.640 ms
84 bytes from 192.168.1.11 icmp_seq=4 ttl=64 time=1.662 ms
84 bytes from 192.168.1.11 icmp_seq=5 ttl=64 time=0.815 ms

PC2-vabazlov>
```

Рис. 2.3: Настройка PC2

2.2 Анализ ARP-трафика в Wireshark

Для анализа ARP-кадров произведён запуск захвата на линии между PC-1 и коммутатором. После включения узлов Wireshark начал фиксировать ARP-трафик, возникший при назначении IP-адресов и выполнении первых запросов.

На снимке отображён обмен Gratuitous ARP и ARP-запросами с указанием MAC-адресов, IP-адресов отправителя и назначения.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	::	ff02::2	ICMPv6	62	Router Solicitation
2	0.001863	::	ff02::2	ICMPv6	62	Router Solicitation
3	0.051094	Private_66:68:00	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.11 (Request)
4	0.052431	Private_66:68:01	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.12 (Request)
5	1.051507	Private_66:68:00	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.11 (Request)
6	1.052965	Private_66:68:01	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.12 (Request)
7	2.052676	Private_66:68:00	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.11 (Request)
8	2.053906	Private_66:68:01	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.12 (Request)

```

> Frame 3: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface -, id 0
✓ Ethernet II, Src: Private_66:68:00 (00:50:79:66:68:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  > Destination: Broadcast (ff:ff:ff:ff:ff:ff)
  > Source: Private_66:68:00 (00:50:79:66:68:00)
    Type: ARP (0x0806)
    [Stream index: 2]
    Padding: 0000000000000000000000000000000000000000000000000000000000000000
    Frame check sequence: 0x00000000 [unverified]
    [FCS Status: Unverified]
  ✓ Address Resolution Protocol (request/gratuitous ARP)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    [Is gratuitous: True]
    Sender MAC address: Private_66:68:00 (00:50:79:66:68:00)
    Sender IP address: 192.168.1.11
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 192.168.1.11

```

Рис. 2.4: ARP-захват

Gratuitous ARP используется для проверки отсутствия конфликта IP, а стандартные ARP-запросы выполняют разрешение MAC-адресов.

2.2.1 Анализ ICMP-трафика

Перед выполнением ICMP-эхо-запроса на PC-2 были изучены параметры команды ping.

```
PC2-vabazlov - PuTTY
PC2-vabazlov> ping ?

ping HOST [OPTION ...]
  Ping the network HOST. HOST can be an ip address or name
  Options:
    -1          ICMP mode, default
    -2          UDP mode
    -3          TCP mode
    -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.

PC2-vabazlov> ping 192.168.1.11 -c 1
84 bytes from 192.168.1.11 icmp_seq=1 ttl=64 time=1.651 ms
PC2-vabazlov> ping 192.168.1.11 -c 1 -2
84 bytes from 192.168.1.11 udp_seq=1 ttl=64 time=2.590 ms
PC2-vabazlov> ping 192.168.1.11 -c 1 -3
Connect    7@192.168.1.11 seq=1 ttl=64 time=3.012 ms
SendData  7@192.168.1.11 seq=1 ttl=64 time=3.403 ms
Close      7@192.168.1.11 seq=1 ttl=64 time=3.040 ms
PC2-vabazlov>
```

Рис. 2.5: Ping options

Выполнен один ICMP-эхо-запрос к узлу PC-1. В Wireshark зафиксированы кадры типа Echo Request и Echo Reply, содержащие идентификатор, TTL, последовательные номера и полезную нагрузку.

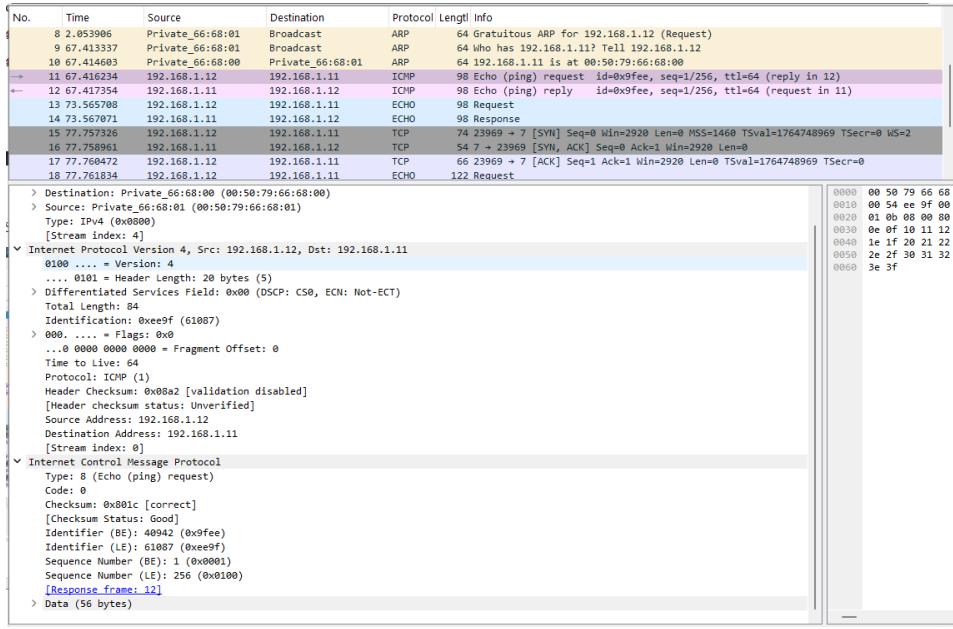


Рис. 2.6: ICMP-анализ

2.2.2 UDP-эхо-запрос

Выполнена отправка одного echo-пакета в UDP-режиме. В Wireshark отображается пакет с протоколом UDP, случайным исходным портом и портом назначения 7, который используется службой echo.

No.	Time	Source	Destination	Protocol	Length	Info
8	2.053906	Private_66:68:01	Broadcast	ARP	64	Gratuitous ARP for 192.168.1.12 (Request)
9	67.413337	Private_66:68:01	Broadcast	ARP	64	iwh has 192.168.1.11? Tell 192.168.1.12
10	67.414603	Private_66:68:00	Private_66:68:01	ARP	64	192.168.1.11 is at 00:50:79:66:68:00
11	67.416234	192.168.1.12	192.168.1.11	ICMP	98	Echo (ping) request id=0x9fee, seq=1/256, ttl=64 (reply in
12	67.417354	192.168.1.11	192.168.1.12	ICMP	98	Echo (ping) reply id=0x9fee, seq=1/256, ttl=64 (request i
13	73.565708	192.168.1.12	192.168.1.11	ECHO	98	Request
14	73.567071	192.168.1.11	192.168.1.12	ECHO	98	Response
15	77.757326	192.168.1.12	192.168.1.11	TCP	74	23969 → 7 [SYN] Seq=0 Win=2920 Len=0 MSS=1460 TSval=1764748969
16	77.758961	192.168.1.11	192.168.1.12	TCP	54	7 → 23969 [SYN, ACK] Seq=0 Ack=1 Win=2920 Len=0
17	77.760472	192.168.1.12	192.168.1.11	TCP	66	23969 → 7 [ACK] Seq=1 Ack=1 Win=0 TSval=1764748969
18	77.761834	192.168.1.12	192.168.1.11	ECHO	122	Request

> Source: Private_66:68:01 (00:50:79:66:68:01)
Type: IPv4 (0x0800)
[Stream index: 4]
`- Internet Protocol Version 4, Src: 192.168.1.12, Dst: 192.168.1.11
` 0100 = Version: 4
.... 0101 = Header Length: 20 bytes (5)
`- Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 84
Identification: 0xeea5 (61093)
` 000 = Flags: 0x0
...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 64
Protocol: UDP (17)
Header Checksum: 0x088c [validation disabled]
[Header checksum status: Unverified]
Source Address: 192.168.1.12
Destination Address: 192.168.1.11
[Stream index: 0]
`- User Datagram Protocol, Src Port: 18673, Dst Port: 7
Source Port: 18673
Destination Port: 7
Length: 64
Checksum: 0x9783 [unverified]
[Checksum Status: Unverified]
[Stream index: 0]
[Stream Packet Number: 1]
`- [Timestamps]
UDP payload (56 bytes)
`- Echo

Рис. 2.7: UDP-анализ

UDP-пакет не устанавливает соединение, содержит только заголовок транспортного уровня и данные.

2.2.3 TCP-эхо-запрос

При отправке echo-запроса в TCP-режиме фиксируется трёхстороннее рукопожатие SYN → SYN/ACK → ACK. Далее передаются данные и соединение завершается.

Такой обмен соответствует требованиям TCP, который обеспечивает установление соединения перед отправкой данных.

2.3 Построение сети с маршрутизатором FRR в GNS3 и анализ трафика

В GNS3 создан новый проект, после чего на рабочем поле размещены устройства: оконечный узел PC1, коммутатор Ethernet и маршрутизатор FRR. Им присвоены имена согласно требованиям: PC1-vabazlov, msk-vabazlov-sw-01 и msk-vabazlov-gw-01. Устройства соединены в единую топологию.



Рис. 2.8: Топология сети

На узле PC1-vabazlov выполнена настройка IPv4-адреса. Устройству назначен адрес 192.168.1.10/24 и шлюз 192.168.1.1. После сохранения конфигурации была выполнена проверка параметров.

```

PC1-vabazlov>
PC1-vabazlov>
PC1-vabazlov> ip 192.168.1.10/24 192.168.1.1
Checking for duplicate address...
PC1-vabazlov : 192.168.1.10 255.255.255.0 gateway 192.168.1.1

PC1-vabazlov> save
Saving startup configuration to startup.vpc
. done
u

PC1-vabazlov> show ip

NAME      : PC1-vabazlov[1]
IP/MASK   : 192.168.1.10/24
GATEWAY   : 192.168.1.1
DNS       :
MAC       : 00:50:79:66:68:00
LPORT     : 10006
RHOST:PORT: 127.0.0.1:10007
MTU       : 1500

PC1-vabazlov>

```

Рис. 2.9: Настройка PC1

Маршрутизатору присвоено имя msk-vabazlov-gw-01.

На интерфейсе eth0 настроен адрес 192.168.1.1/24. Интерфейс активирован, а конфигурация сохранена.

```

Hello, this is FRRouting (version 8.2.2).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

frr# configure terminal
frr(config)# hostname msk-vabazlov-gw-01
msk-vabazlov-gw-01(config)# exit
msk-vabazlov-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-vabazlov-gw-01# configure terminal
msk-vabazlov-gw-01(config)# interface eth0
msk-vabazlov-gw-01(config-if)# ip address 192.168.1.1/24
msk-vabazlov-gw-01(config-if)# no shutdown
msk-vabazlov-gw-01(config-if)# exit
msk-vabazlov-gw-01(config)# exit
msk-vabazlov-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-vabazlov-gw-01#

```

Рис. 2.10: Настройка FRR

Проверка конфигурации показывает корректно настроенные параметры, включая IP-адрес интерфейса eth0.

```
msk-vabazlov-gw-01# show running-config
Building configuration...

Current configuration:
c!
frr version 8.2.2
frr defaults traditional
hostname frr
hostname msk-vabazlov-gw-01
service integrated-vtysh-config
!
interface eth0
 ip address 192.168.1.1/24
exit
!
end
msk-vabazlov-gw-01#
```

Рис. 2.11: Конфигурация FRR

Сводка по интерфейсам предоставляет подтверждение, что интерфейс eth0 находится в состоянии up.

```
msk-vabazlov-gw-01#
msk-vabazlov-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-vabazlov-gw-01#
```

Рис. 2.12: Интерфейсы FRR

На PC1-vabazlov выполнена отправка ICMP-эхо-запросов на адрес маршрутиза-

тока 192.168.1.1. Узел успешно отвечает, что подтверждает корректность настроек.

```
PC1-vabazlov> ping 192.168.1.1
84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=4.140 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=2.658 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=3.151 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=3.275 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=0.898 ms

PC1-vabazlov>
```

Рис. 2.13: Пинг FRR

В захвате Wireshark отображаются ICMP Echo Request и Echo Reply между PC1 и маршрутизатором.

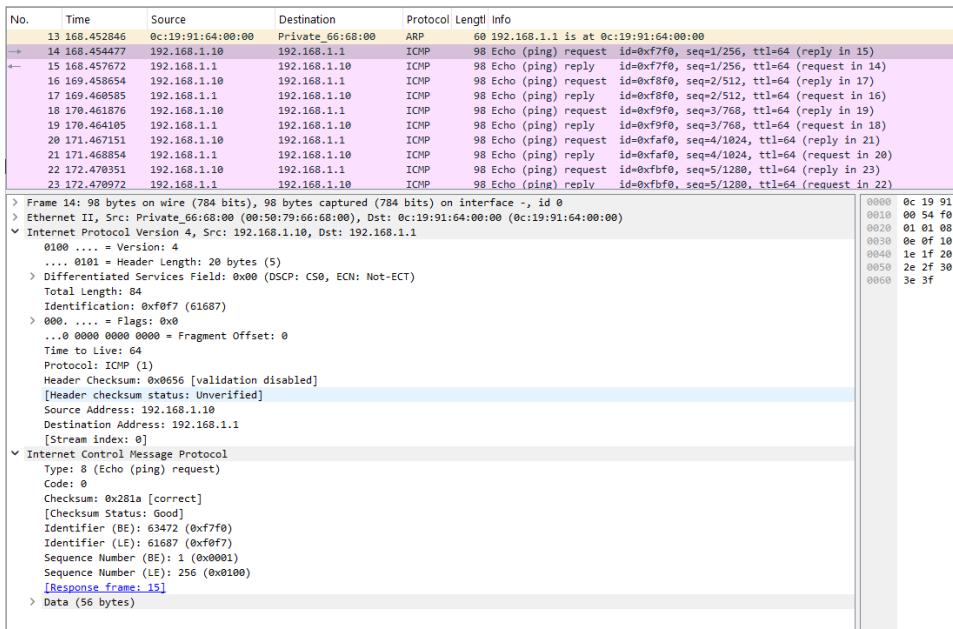


Рис. 2.14: ICMP-трафик FRR

2.4 Построение сети с маршрутизатором VyOS

В GNS3 создан новый проект, на рабочей области размещены VPCS, коммутатор Ethernet и маршрутизатор VyOS.

Устройствам присвоены имена: PC1-vabazlov, msk-vabazlov-sw-01 и msk-vabazlov-gw-01. Устройства соединены в топологию.



Рис. 2.15: Топология сети VyOS

2.5 Настройка PC1

Узел PC1 получил адрес 192.168.1.10/24 с шлюзом 192.168.1.1.

2.6 Настройка маршрутизатора VyOS

На маршрутизаторе после установки образа выполнены следующие операции:

- устройство переведено в режим конфигурации
- присвоено имя msk-vabazlov-gw-01
- интерфейсу eth0 задан адрес 192.168.1.1/24
- изменения применены и сохранены

```
you can check individual component licenses under /usr/share/doc/ /copyright
vyos@vyos:~$ configure
[edit]
vyos@vyos# set system host-name msk-vabazlov-gw-01
[edit]
vyos@vyos# delete interfaces ethernet eth0 address dhcp
[edit]
vyos@vyos# set interfaces ethernet eth0 address 192.168.1.1/24
[edit]
vyos@vyos# compare
[edit interfaces ethernet eth0]
+address dhcp
+address 192.168.1.1/24
[edit system]
>host-name msk-vabazlov-gw-01
[edit]
vyos@vyos# commit
[edit]
vyos@vyos# save
Saving configuration to '/config/config.boot'...
Done
[edit]
vyos@vyos#
```

Рис. 2.16: Настройка VyOS

Просмотр информации об интерфейсах подтверждает корректность настройки адреса.

```
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:e5:cb:5f:00:00
}
ethernet eth1 {
    hw-id 0c:e5:cb:5f:00:01
}
ethernet eth2 {
    hw-id 0c:e5:cb:5f:00:02
}
loopback lo {
}
[edit]
vyos@vyos#
```

Рис. 2.17: Интерфейсы VyOS

2.7 Проверка работоспособности сети

PC1-vabazlov успешно отправляет ICMP-эхо-запросы на маршрутизатор VyOS по адресу 192.168.1.1.

```
PC1-vabazlov>
PC1-vabazlov> ping 192.168.1.1

 84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=3.694 ms
 84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=1.838 ms
 84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=3.181 ms
 84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=5.053 ms
 84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=1.938 ms
6
6PC1-vabazlov>
```

Рис. 2.18: Пинг VyOS

В окне Wireshark видны ICMP Echo Request / Echo Reply между устройствами.

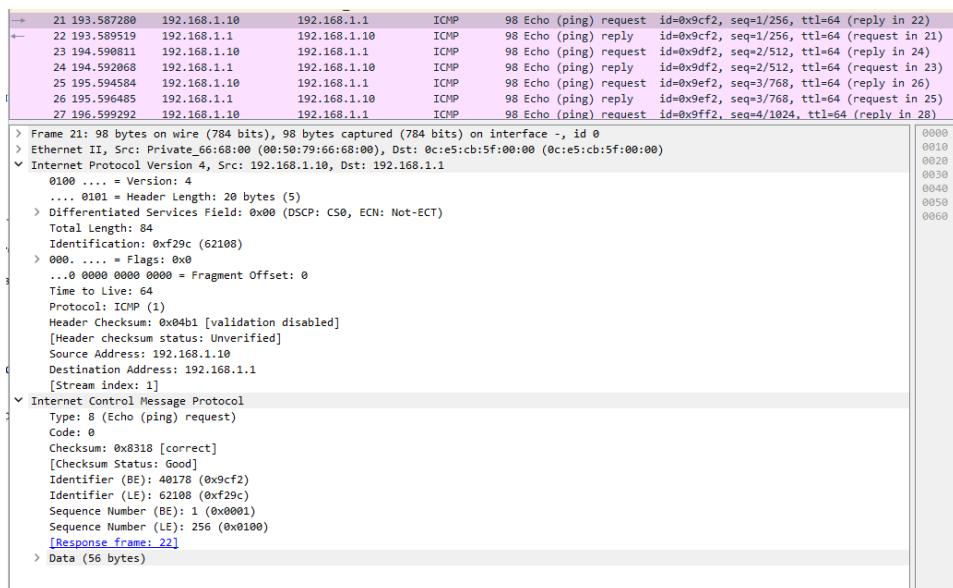


Рис. 2.19: ICMP-трафик VyOS

3 Заключение

В ходе работы была построена простейшая сеть в GNS3 на базе коммутатора, маршрутизаторов FRR и VyOS, а также оконечного узла VPCS. Для всех устройств выполнена настройка IPv4-адресации, после чего проведена проверка связности между узлом PC1 и шлюзом. На обоих вариантах маршрутизаторов подтверждена корректная работа сети — эхо-запросы успешно достигали адреса 192.168.1.1, что фиксировалось как в терминале, так и в захваченных пакетах Wireshark.

Захват трафика позволил проанализировать последовательность ARP-обращений при первом обращении к шлюзу и структуру ICMP-пакетов на этапах обмена Echo Request и Echo Reply. Полученные данные демонстрируют корректное разрешение адресов, формирование ARP-таблицы и стабильную передачу ICMP-сообщений.