

Лабораторная работа №5

Простые сети в GNS3. Анализ трафика

Демидова Екатерина Алексеевна

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
3.1	Моделирование простейшей сети на базе коммутатора в GNS3 . .	6
3.2	Анализ трафика в GNS3 посредством Wireshark	10
3.3	Моделирование простейшей сети на базе маршрутизатора FRR в GNS3	15
3.4	Моделирование простейшей сети на базе маршрутизатора VyOS в GNS3	21
4	Выводы	25

Список иллюстраций

3.1	Добавление устройств и изменение их имен	7
3.2	Соединение VPCS с коммутатором	7
3.3	Параметры импорта	8
3.4	Задание IP-адреса PC1-eademidova	9
3.5	Задание IP-адреса PC2-eademidova	9
3.6	Проверка соединения между PC-1 и PC-2	10
3.7	Запуск анализатора трафика	11
3.8	ARP пакеты	11
3.9	ARP пакеты	12
3.10	Эхо-запрос в ICMP-модуле	12
3.11	ICMP пакеты	13
3.12	Эхо-запрос в UDP-модуле	13
3.13	UDP пакеты	13
3.14	Эхо-запрос в TCP-модуле	14
3.15	TCP пакеты	14
3.16	Захват трафика на соединении между коммутатором и маршрути- затором	15
3.17	Запуск всех устройств проекта	16
3.18	Консоль всех устройств проекта	16
3.19	Настройка IP-адресации для интерфейса узла PC-1	17
3.20	Настройка IP-адресации для интерфейса локальной сети маршру- тизатора	18
3.21	Проверка конфигурации маршрутизатора и настройки IP-адресации	19
3.22	Проверка подключения	20
3.23	Анализ информации в Wireshark	21
3.24	Захват трафика на соединении между коммутатором и маршрути- затором	22
3.25	Настройка IP-адресации для интерфейса узла PC-1	22
3.26	Режим конфигурации маршрутизатора VyOS	23
3.27	Проверка соединения	24
3.28	Анализ трафика Wireshark	24

1 Цель работы

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

2 Задание

1. Смоделировать простейшую сеть на базе коммутатора в GNS3
2. Проанализировать трафик в GNS3 посредством Wireshark
3. Смоделировать простейшую сеть на базе маршрутизатора FRR в GNS3
4. Смоделировать простейшую сеть на базе маршрутизатора VyOS в GNS3

3 Выполнение лабораторной работы

3.1 Моделирование простейшей сети на базе коммутатора в GNS3

1. Построить в GNS3 топологию сети, состоящей из коммутатора Ethernet и двух оконечных устройств (персональных компьютеров).
2. Задать оконечным устройствам IP-адреса в сети 192.168.1.0/24. Проверить связь.

Запустим GNS3 VM и GNS3 и создадим новый проект. В рабочей области GNS3 разместим коммутатор Ethernet и два VPCS. Щёлкнув на устройстве правой кнопкой мыши в меню `Configure` изменим название устройства, включив в имя устройства имя своей учётной записи. Коммутатору присвоим название `msk-eademidova-sw-01` (рис. 3.1). Затем соединим VPCS с коммутатором и отобразим обозначение интерфейсов соединения (рис. 3.2).

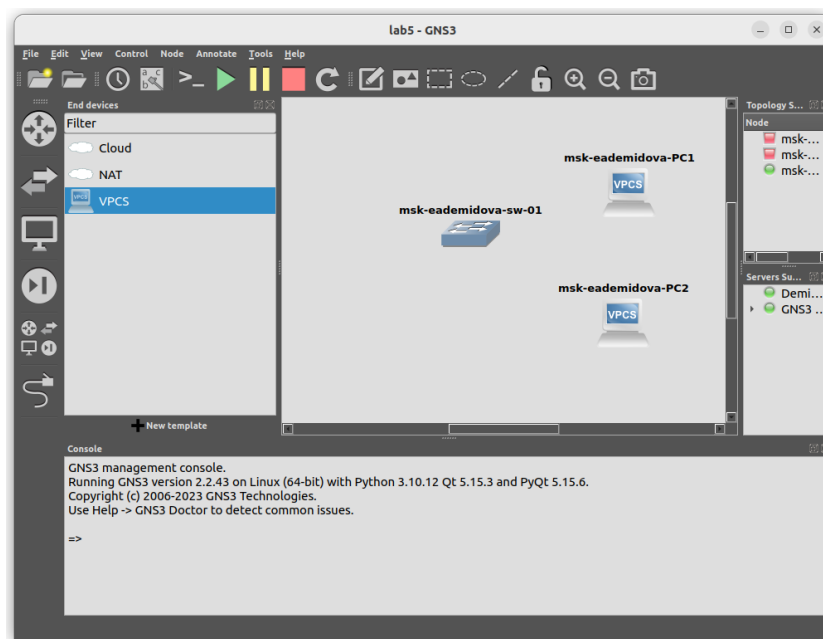


Рис. 3.1: Добавление устройств и изменение их имен

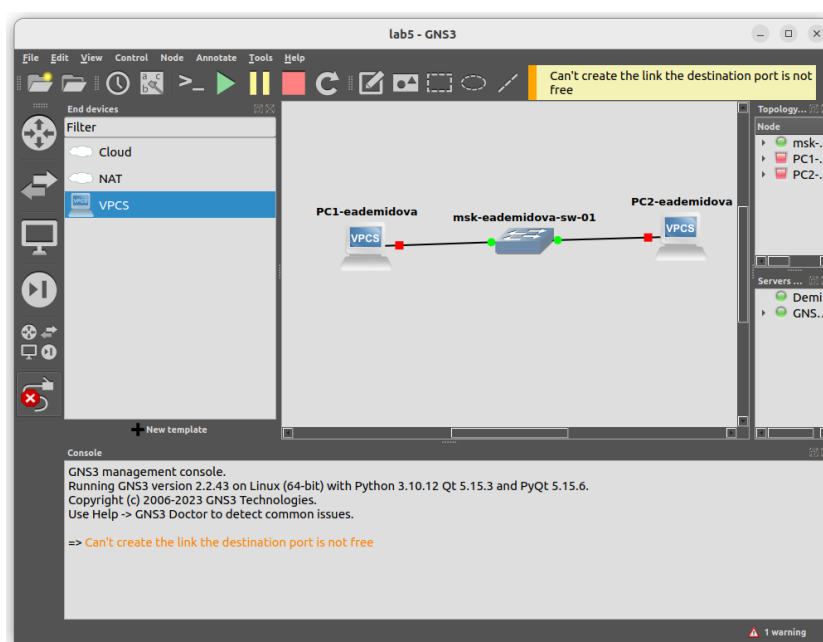
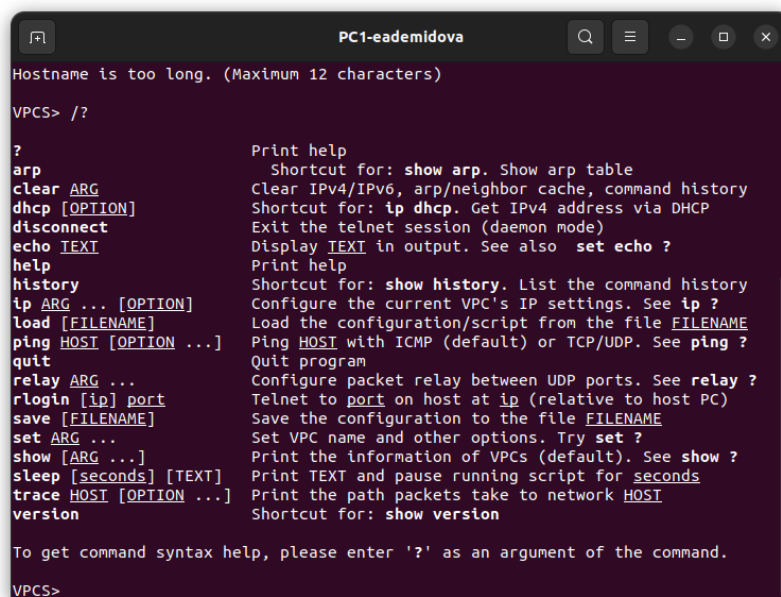


Рис. 3.2: Соединение VPCS с коммутатором

Зададим IP-адреса VPCS. Для этого с помощью меню, вызываемого правой кнопкой мыши, запустим Start, PC-1, затем вызовем его терминал Console. Для

просмотра синтаксиса возможных для ввода команд наберем /?(рис. 3.3).



```
PC1-eademidova
Hostname is too long. (Maximum 12 characters)
VPCS> /?
?
? Print help
arp Shortcut for: show arp. Show arp table
clear ARG Clear IPv4/IPv6, arp/neighbor cache, command history
dhcp [OPTION] Shortcut for: ip dhcp. Get IPv4 address via DHCP
disconnect Exit the telnet session (daemon mode)
echo TEXT Display TEXT in output. See also set echo ?
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

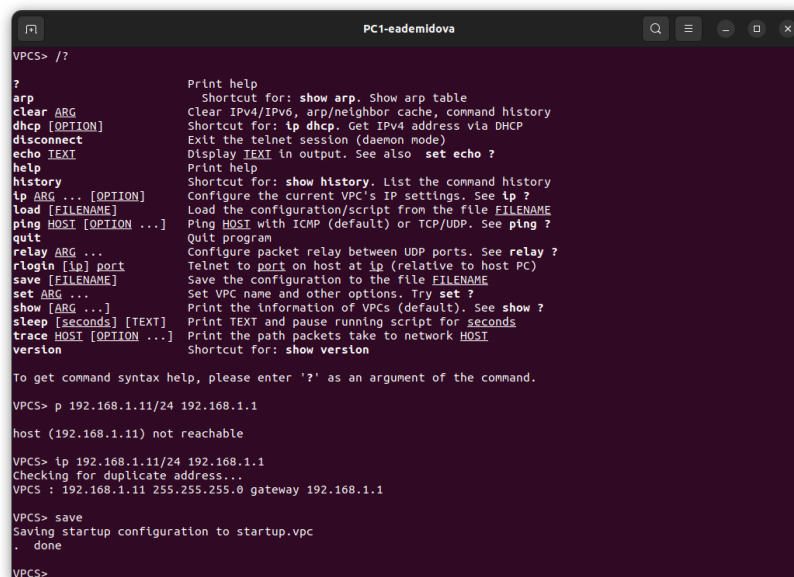
To get command syntax help, please enter '?' as an argument of the command.
VPCS>
```

Рис. 3.3: Параметры импорта

Для задания IP-адреса 192.168.1.11 в сети 192.168.1.0/24 введем:

```
ip 192.168.1.11/24 192.168.1.1
```

А для сохранения конфигураций введём команду save(рис. 3.4).



```
VPCS> /?
?          Print help
arp        Shortcut for: show arp. Show arp table
clear ARG  clear IPv4/IPv6, arp/neighbor cache, command history
dhcp [OPTION]  Shortcut for: ip dhcp. Get IPv4 address via DHCP
disconnect Exit the telnet session (daemon mode)
echo TEXT  Display TEXT in output. See also set echo ?
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> p 192.168.1.11/24 192.168.1.1

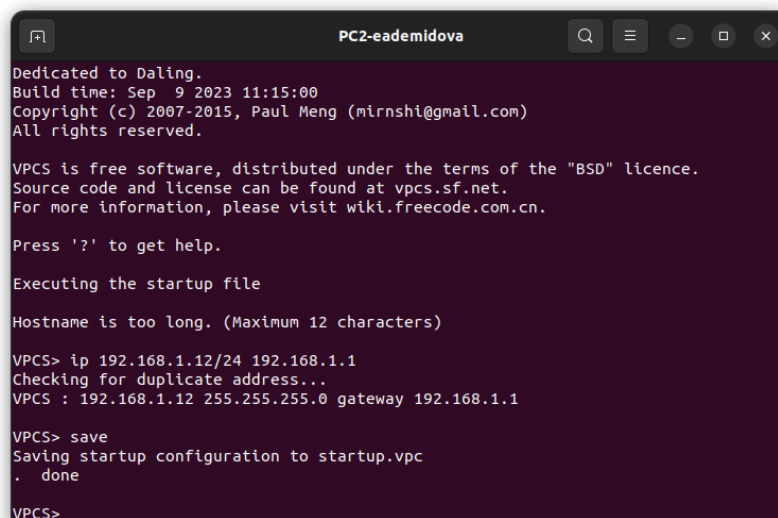
host (192.168.1.11) not reachable

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

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

Рис. 3.4: Задание IP-адреса PC1-eademidova

Те же действия продоелаем для второго VPCS(рис. 3.5):



```
PC2-eademidova
Dedicated to Daling.
Build time: Sep  9 2023 11:15:00
Copyright (c) 2007-2015, Paul Meng (mirnshi@gmail.com)
All rights reserved.

VPCS is free software, distributed under the terms of the "BSD" licence.
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

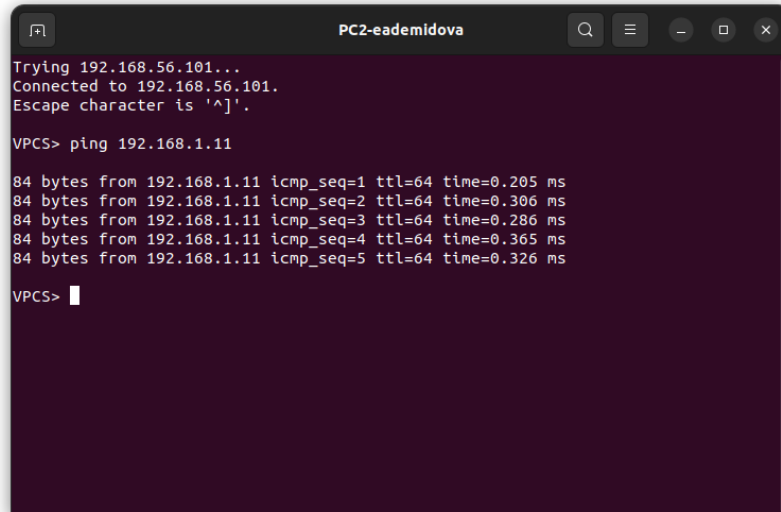
Hostname is too long. (Maximum 12 characters)

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>
```

Рис. 3.5: Задание IP-адреса PC2-eademidova

Проверим работоспособность соединения между PC-1 и PC-2 с помощью команды ping(рис. 3.6).



```
PC2-eademidova
Trying 192.168.56.101...
Connected to 192.168.56.101.
Escape character is '^]'.

VPCS> ping 192.168.1.11

84 bytes from 192.168.1.11 icmp_seq=1 ttl=64 time=0.205 ms
84 bytes from 192.168.1.11 icmp_seq=2 ttl=64 time=0.306 ms
84 bytes from 192.168.1.11 icmp_seq=3 ttl=64 time=0.286 ms
84 bytes from 192.168.1.11 icmp_seq=4 ttl=64 time=0.365 ms
84 bytes from 192.168.1.11 icmp_seq=5 ttl=64 time=0.326 ms

VPCS> 
```

Рис. 3.6: Проверка соединения между PC-1 и PC-2

В конце остановим в проекте все узлы(меню GNS3 Control Stop all nodes).

3.2 Анализ трафика в GNS3 посредством Wireshark

1. С помощью Wireshark захватить и проанализировать ARP-сообщения.
2. С помощью Wireshark захватить и проанализировать ICMP-сообщения.

Запустим на соединении между PC-1 и коммутатором анализатор трафика. Для этого щёлкнем правой кнопкой мыши на соединении, выберем в меню Start capture(рис. 3.7).

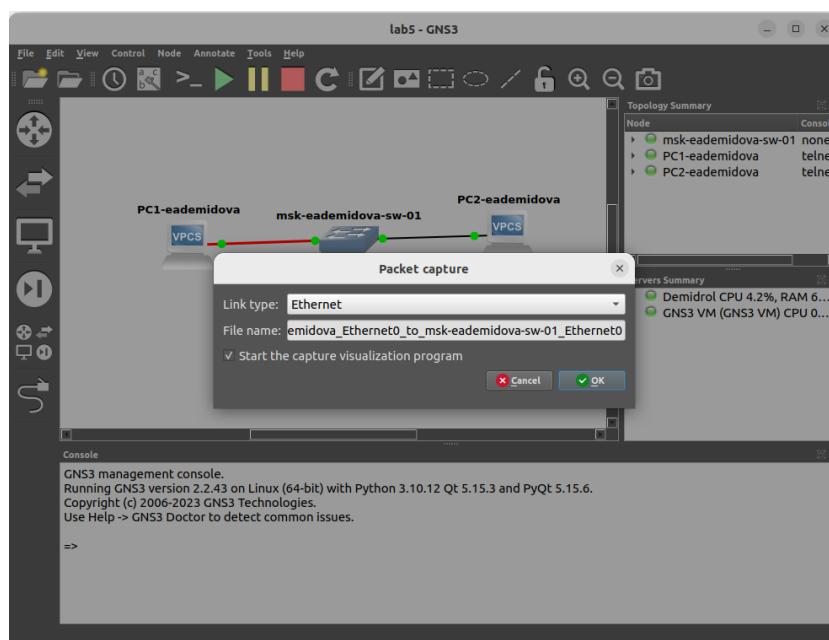


Рис. 3.7: Запуск анализатора трафика

Запустился Wireshark, а в проекте GNS3 на соединении появился значок (рис. 3.8).

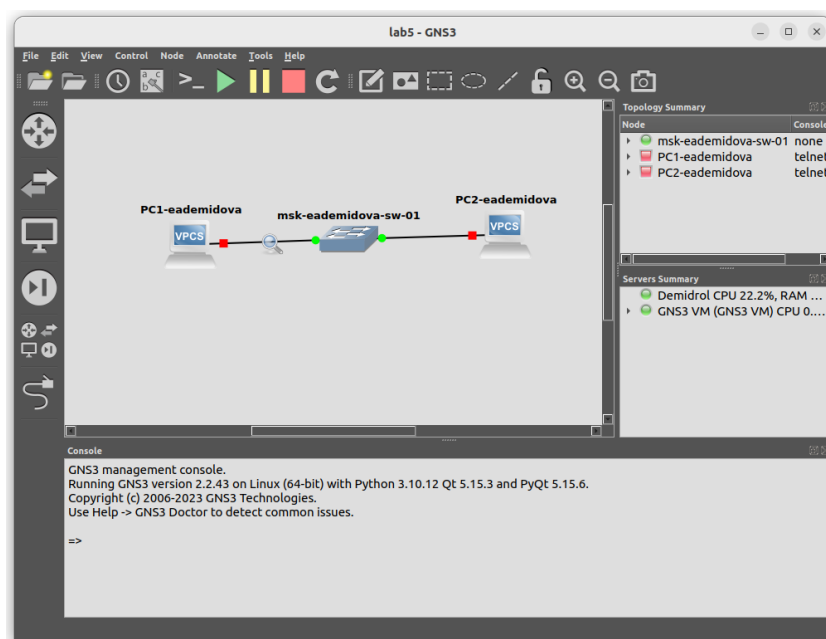


Рис. 3.8: ARP пакеты

В проекте GNS3 стартуем все узлы (меню GNS3 Control Start/Resume all nodes). В окне Wireshark отобразилась информация по протоколу ARP.

Изучим запрос и ответ ARP в программе Wireshark(рис. 3.9). В обоих случаях длина кадра равняется 98 байт. В начале сформировались запросы безвозмездных пакетов ARP для PC-1(в этом случае источник – Private_66:68:00, а пункт назначения - Broadcast) и для PC-2(в этом случае источник – Private_66:68:01, а пункт назначения - Broadcast). Затем был сформирован запрос от PC-2 на передачу MAC-адреса PC-1 и получен ответ - MAC-адрес.

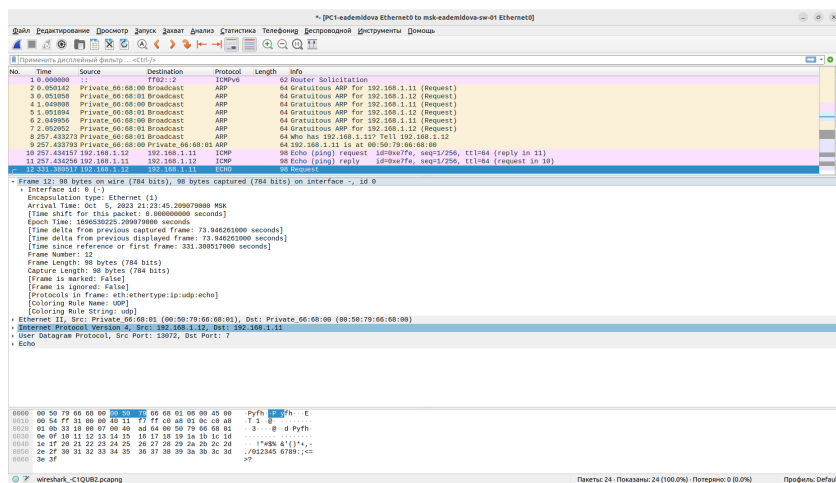


Рис. 3.9: ARP пакеты

В терминале PC-2 посмотрим информацию по опциям команды ping, введя ping /?. Затем сделаем один эхо-запрос в ICMP-моду к узлу PC-1(рис. 3.10). Изучим эхо-запрос и эхо-ответ ICMP в программе Wireshark(рис. 3.11). В обоих случаях длина кадра равняется 98 байт. В случае эхо-запроса точка назначения – PC-1, а источник – PC-2, в случае же эхо-ответа – наоборот.

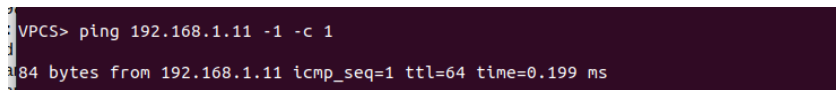


Рис. 3.10: Эхо-запрос в ICMP-моду

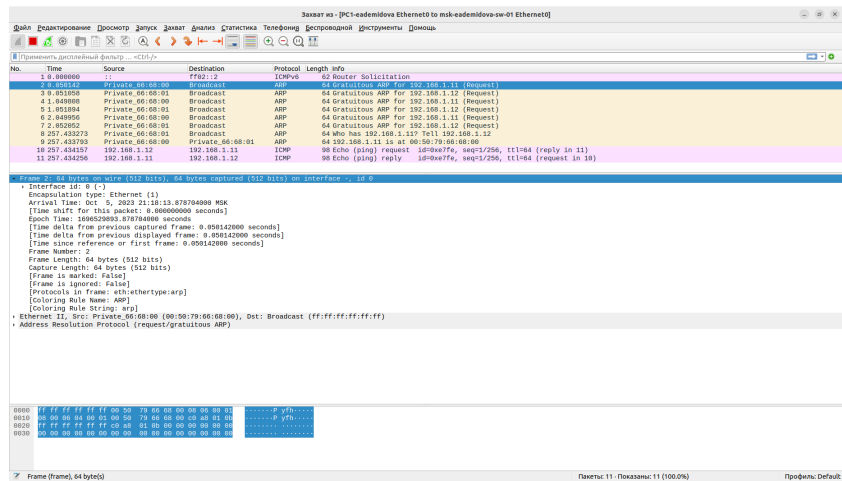


Рис. 3.11: ICMP пакеты

Сделаем один эхо-запрос в UDP-модe к узлу PC-1(рис. 3.12). В окне Wireshark проанализируем полученную информацию(рис. 3.13). В обоих случаях длина кадра равняется 98 байт. В случае эхо-запроса точка назначения – PC-1, а источник – PC-2, в случае же эхо-ответа – наоборот.

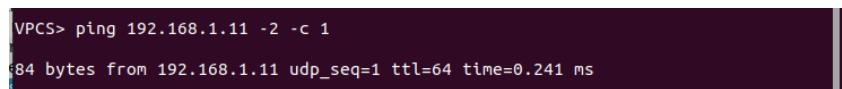


Рис. 3.12: Эхо-запрос в UDP-модe

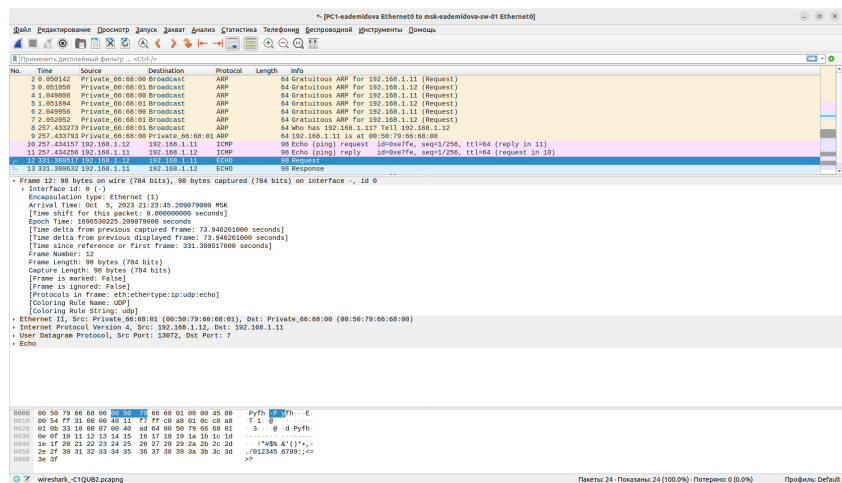


Рис. 3.13: UDP пакеты

Сделаем один эхо-запрос в TCP-моду к узлу PC-1(3.12). В окне Wireshark проанализируем полученную информацию(3.13). Порт источника задан случайно равен 35487, порт назначения равен 7. В случае ответа порты заданы наоборот. Также можно увидеть handshake протокола TCP. В первом пакете установлен бит SYN(Syn: set). Во втором пакете установлены биты SYN и ACK(Syn: set, Acknowledgment: set). А в следующем пакете установлен бит ACK(Acknowledgment: set). Также есть пакеты с битом FIN, завершающим handshake.

```
VPCS> ping 192.168.1.11 -3 -c 1
Connect 7@192.168.1.11 seq=1 ttl=64 time=1.379 ms
SendData 7@192.168.1.11 seq=1 ttl=64 time=1.088 ms
Close 7@192.168.1.11 seq=1 ttl=64 time=2.322 ms
```

Рис. 3.14: Эхо-запрос в TCP-моду

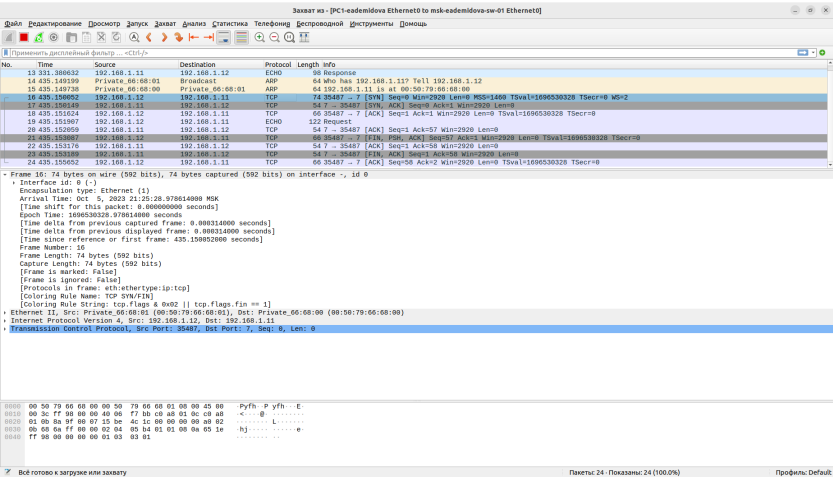


Рис. 3.15: TCP пакеты

Остановим захват пакетов в Wireshark.

3.3 Моделирование простейшей сети на базе маршрутизатора FRR в GNS3

1. Построить в GNS3 топологию сети, состоящей из маршрутизатора FRR, коммутатора Ethernet и оконечного устройства.
2. Задать оконечному устройству IP-адрес в сети 192.168.1.0/24.
3. Присвоить интерфейсу маршрутизатора адрес 192.168.1.1/24
4. Проверить связь.

Запустим GNS3 VM и GNS3. Создадим новый проект. В рабочей области GNS3 разместим VPCS, коммутатор Ethernet и маршрутизатор FRR. Изменим отображаемые названия устройств. Включим захват трафика на соединении между коммутатором и маршрутизатором (3.14). Затем запустим все устройства проекта (3.15). Откроем консоль всех устройств проекта (3.16).

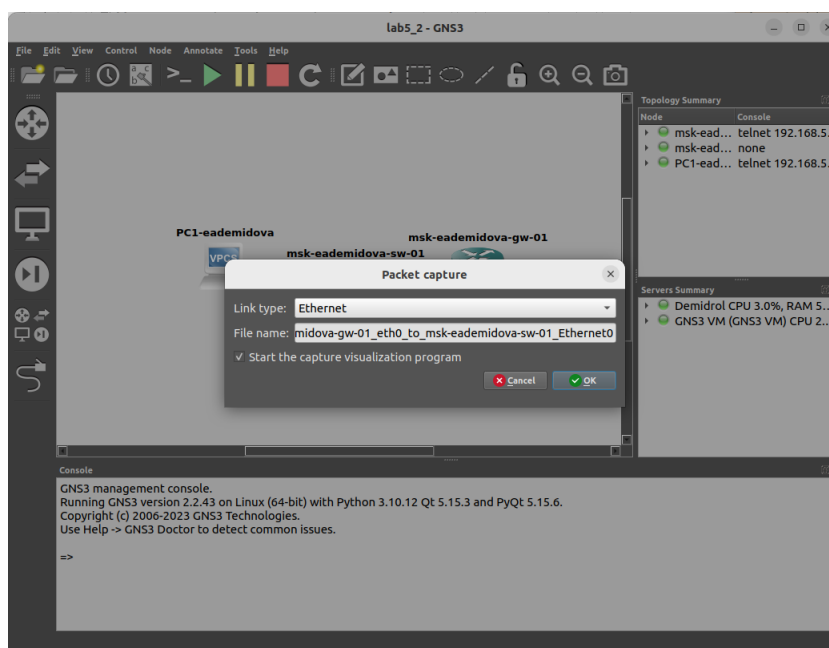


Рис. 3.16: Захват трафика на соединении между коммутатором и маршрутизатором

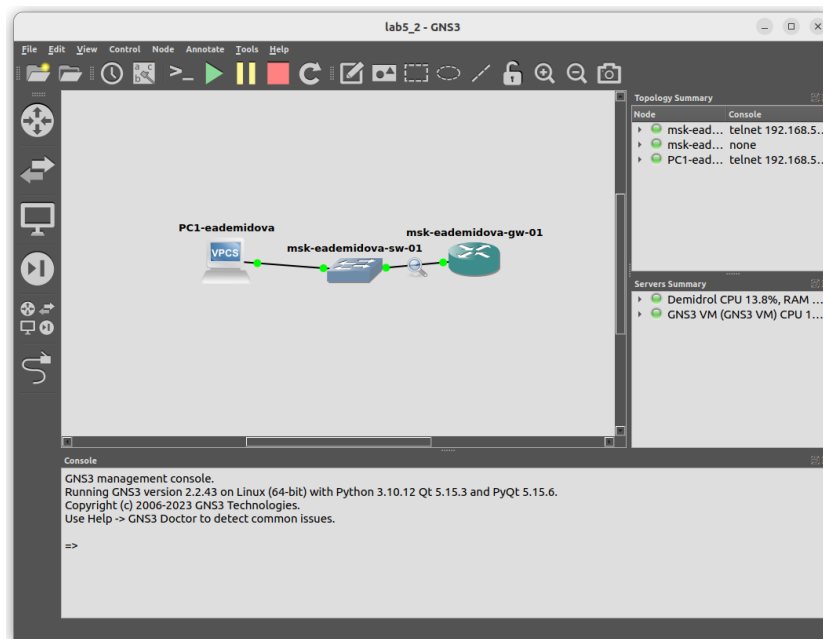


Рис. 3.17: Запуск всех устройств проекта

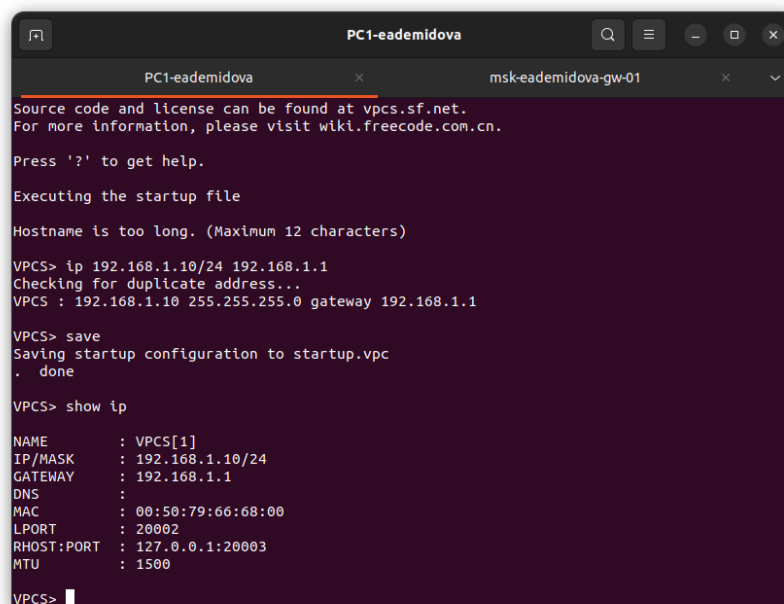
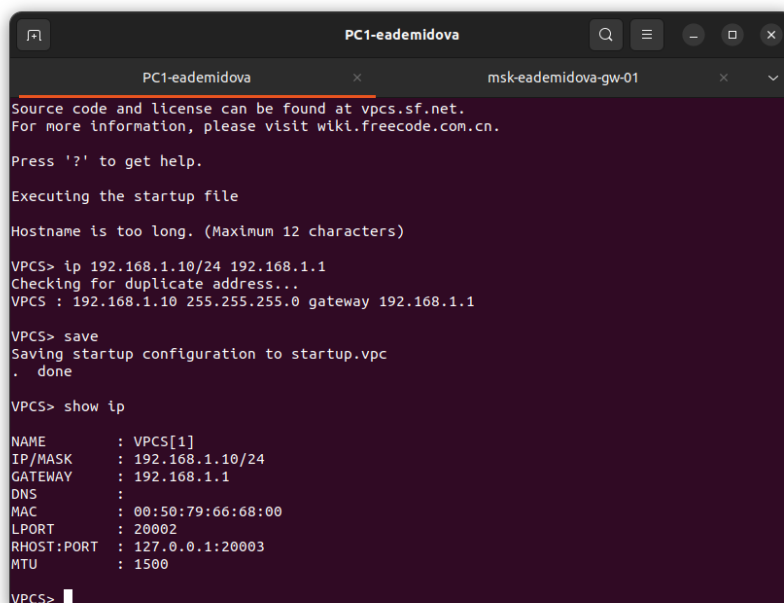


Рис. 3.18: Консоль всех устройств проекта

Настроим IP-адресацию для интерфейса узла PC1(3.19):



```
PC1-eademidova
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 192.168.1.10/24 192.168.1.1
Checking for duplicate address...
VPCS : 192.168.1.10 255.255.255.0 gateway 192.168.1.1

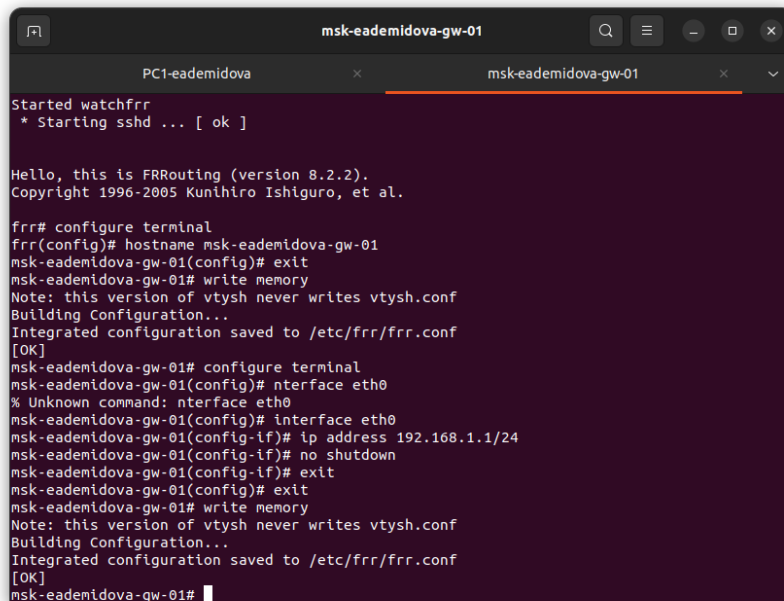
VPCS> save
Saving startup configuration to startup.vpc
. done

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    : 20002
RHOST:PORT : 127.0.0.1:20003
MTU       : 1500

VPCS>
```

Рис. 3.19: Настройка IP-адресации для интерфейса узла PC-1

Настроим IP-адресацию для интерфейса локальной сети маршрутизатора(3.20):



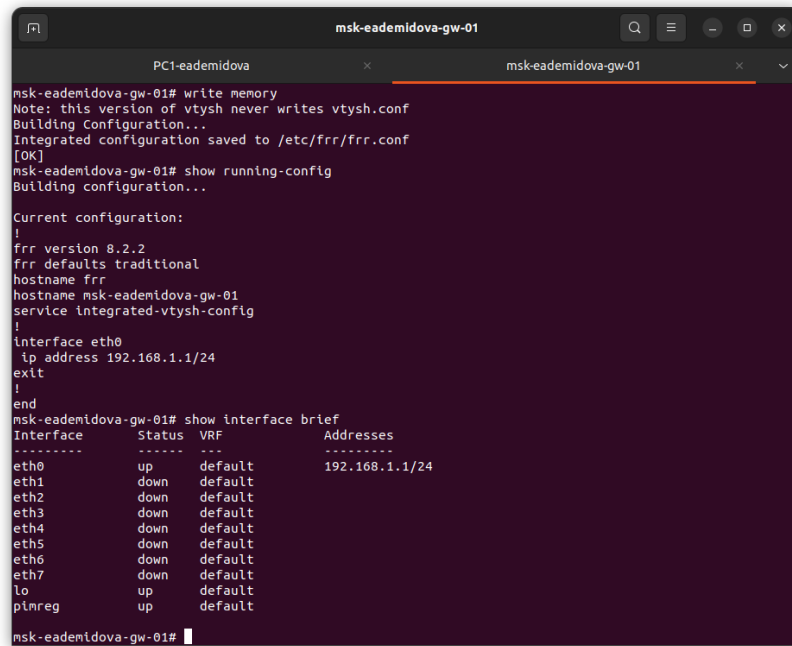
```
msk-eademidova-gw-01
PC1-eademidova x msk-eademidova-gw-01 x
Started watchfrr
* Starting sshd ... [ ok ]

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

frr# configure terminal
frr(config)# hostname msk-eademidova-gw-01
msk-eademidova-gw-01(config)# exit
msk-eademidova-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-eademidova-gw-01# configure terminal
msk-eademidova-gw-01(config)# nterface eth0
% Unknown command: nterface eth0
msk-eademidova-gw-01(config)# interface eth0
msk-eademidova-gw-01(config-if)# ip address 192.168.1.1/24
msk-eademidova-gw-01(config-if)# no shutdown
msk-eademidova-gw-01(config-if)# exit
msk-eademidova-gw-01(config)# exit
msk-eademidova-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-eademidova-gw-01#
```

Рис. 3.20: Настройка IP-адресации для интерфейса локальной сети маршрутизатора

Проверим конфигурацию маршрутизатора и настройки IP-адресации(3.21):



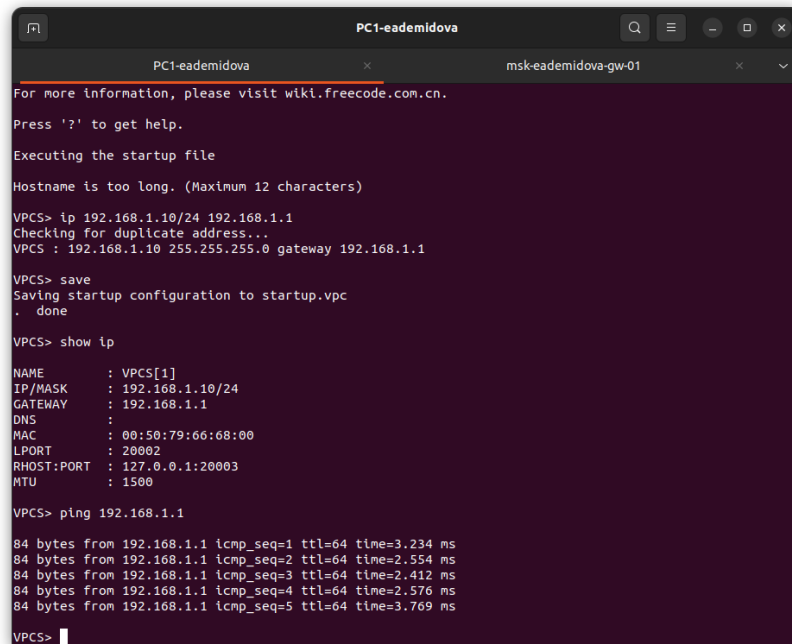
```
msk-eademidova-gw-01# write memory
Note: this version of vtysh never writes vtysh.conf
Building Configuration...
Integrated configuration saved to /etc/frr/frr.conf
[OK]
msk-eademidova-gw-01# show running-config
Building configuration...

Current configuration:
!
frr version 8.2.2
frr defaults traditional
hostname frr
hostname msk-eademidova-gw-01
service integrated-vtysh-config
!
interface eth0
 ip address 192.168.1.1/24
exit
!
end
msk-eademidova-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
p1nreg        up     default

msk-eademidova-gw-01#
```

Рис. 3.21: Проверка конфигурации маршрутизатора и настройки IP-адресации

Проверим подключение. Узел PC1 успешно отправляет эхо-запросы ICMP на адрес маршрутизатора 192.168.1.1(3.22).



```
PC1-eademidova
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 192.168.1.10/24 192.168.1.1
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     : 20002
RHOST:PORT : 127.0.0.1:20003
MTU        : 1500

VPCS> ping 192.168.1.1
84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=3.234 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=2.554 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=2.412 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=2.576 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=3.769 ms

VPCS>
```

Рис. 3.22: Проверка подключения

В окне Wireshark проанализируем полученную информацию(3.23). Было отправлено 5 пакетов формата ICMP. В эхо-запросе источником является IP-адрес PC-1, а пунктом назначения – IP-адрес шлюза маршрутизатора. В эхо-ответе – наоборот. Также были сформированы ARP пакеты запрашивающий MAC-адрес шлюза маршрутизатора перед пингованием его и сообщаящий этот MAC-адрес PC-1, а затем запрашивающие MAC-адрес PC-1 и сообщаящие его шлюзу.

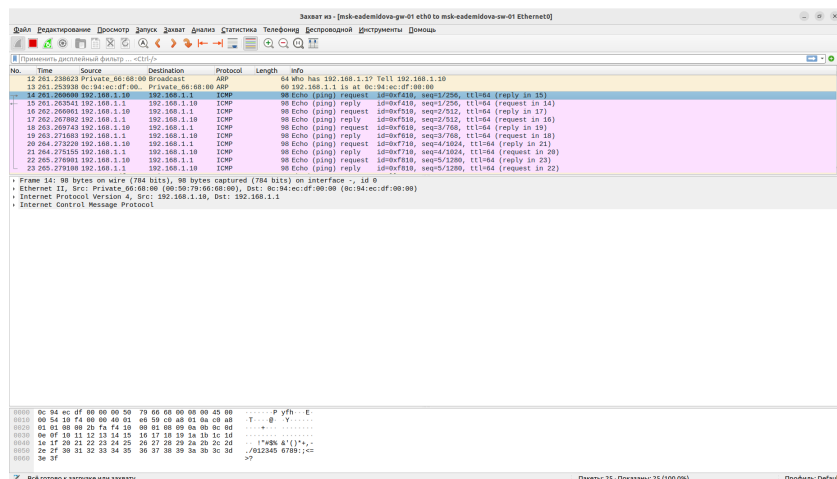


Рис. 3.23: Анализ информации в Wireshark

В конце остановим захват пакетов в Wireshark и остановим все устройства в проекте.

3.4 Моделирование простейшей сети на базе маршрутизатора VyOS в GNS3

1. Построить в GNS3 топологию сети, состоящей из маршрутизатора VyOS, коммутатора Ethernet и оконечного устройства.
2. Задать оконечному устройству IP-адрес в сети 192.168.1.0/24.
3. Присвоить интерфейсу маршрутизатора адрес 192.168.1.1/24
4. Проверить связь.

Запустим GNS3 VM и GNS3. Создадим новый проект.

В рабочей области GNS3 разместим VPCS, коммутатор Ethernet и маршрутизатор VyOS.

Изменим отображаемые названия устройств. Включим захват трафика на соединении между коммутатором и маршрутизатором. Запустим все устройства проекта. Откройте консоль всех устройств проекта(3.24).

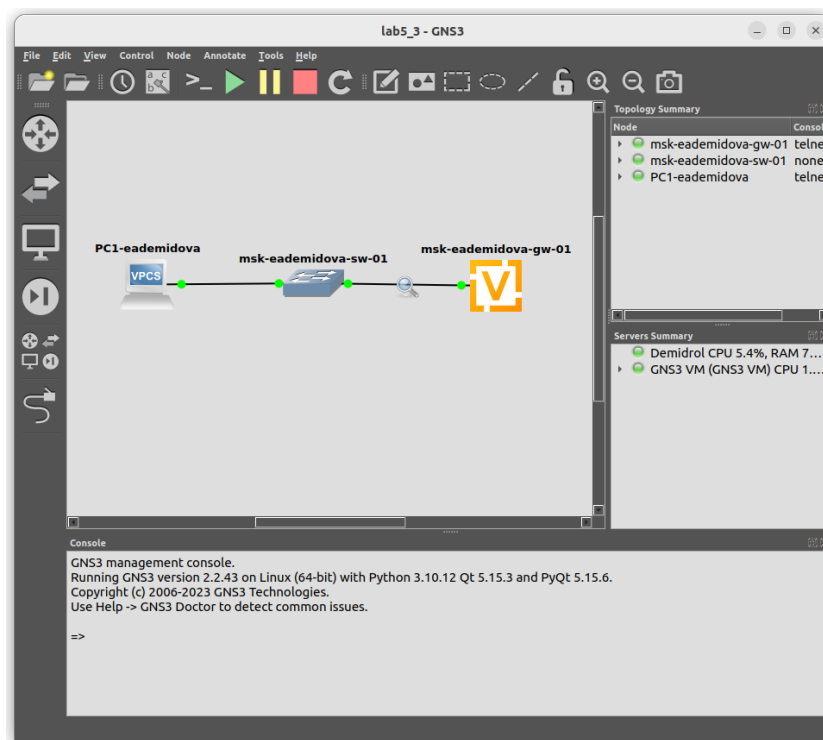


Рис. 3.24: Захват трафика на соединении между коммутатором и маршрутизатором

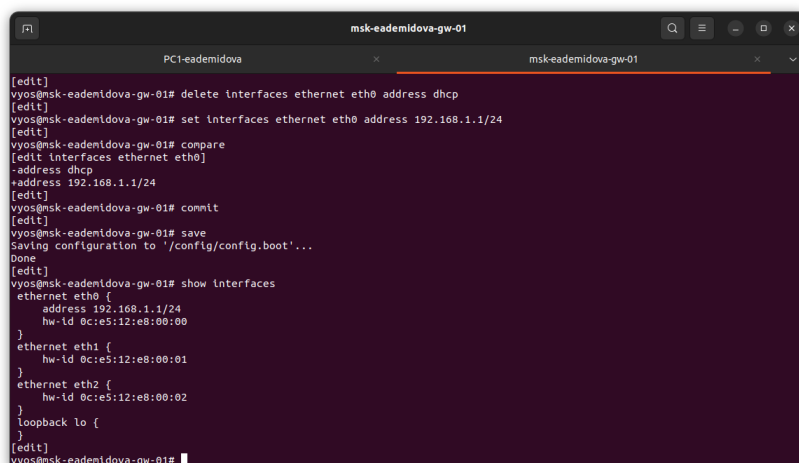
Настроим IP-адресацию для интерфейса узла PC1(3.25).



Рис. 3.25: Настройка IP-адресации для интерфейса узла PC-1

Настроим маршрутизатор VyOS. После загрузки введем логин `vyos` и пароль `vyos`: В рабочем режиме в командной строке отображается символ `$`. Перейдем в режим конфигурирования с помощью команды `configure`. Изменим имя устройства с помощью команды `set system host-name msk-eademidova-gw-01`. Изменения в имени устройства вступят в силу после применения и сохранения конфигурации и перезапуска устройства.

Затем зададим IP-адрес на интерфейсе `eth0`. Но так как на моем устройстве уже был установлен `dhcp` адрес, то сначала я удалила его, а затем установила адрес IPv4. Также посмотрим внесённые в конфигурацию изменения, применим эти изменения и сохраним. В конце выйдем из режима конфигурирования(3.26).



```
[edit]
vyos@msk-eademidova-gw-01# delete interfaces ethernet eth0 address dhcp
[edit]
vyos@msk-eademidova-gw-01# set interfaces ethernet eth0 address 192.168.1.1/24
[edit]
vyos@msk-eademidova-gw-01# compare
[edit interfaces ethernet eth0]
-address dhcp
+address 192.168.1.1/24
[edit]
vyos@msk-eademidova-gw-01# commit
[edit]
vyos@msk-eademidova-gw-01# save
Saving configuration to '/config/config.boot'...
Done
[edit]
vyos@msk-eademidova-gw-01# show interfaces
 ethernet eth0 {
    address 192.168.1.1/24
    hw-id 0c:e5:12:e8:00:00
  }
 ethernet eth1 {
    hw-id 0c:e5:12:e8:00:01
  }
 ethernet eth2 {
    hw-id 0c:e5:12:e8:00:02
  }
 loopback lo {
  }
[edit]
vyos@msk-eademidova-gw-01#
```

Рис. 3.26: Режим конфигурации маршрутизатора VyOS

Проверим подключение. Узел PC1 дупешно отправлять эхо-запросы на адрес маршрутизатора 192.168.1.1(3.27).

```
PC1-eademidova
mskeademidova-gw-01

Trying 192.168.56.101...
Connected to 192.168.56.101.
Escape character is '^['.

Welcome to Virtual PC Simulator, version 0.8.3
Dedicated to Daling.
Build time: Sep  9 2023 11:15:00
Copyright (c) 2007-2015, Paul Meng (mirnshi@gmail.com)
All rights reserved.

VPCS is free software, distributed under the terms of the "BSD" licence.
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

Checking for duplicate address...
VPCS: 192.168.1.10 255.255.255.0 gateway 192.168.1.1

VPCS> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=4.069 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=1.603 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=1.865 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=1.890 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=1.522 ms

VPCS>
```

Рис. 3.27: Проверка соединения

В окне Wireshark проанализируйте полученную информацию(3.28). Было от-
првлено 5 пакетов формата ICMP. В эхо-запросе источником является IP-адрес
PC-1, а пунктом назначения – IP-адрес шлюза маршрутизатора. В эхо-ответе –
наоборот. Также были сформированы ARP пакеты запрашивающий MAC-адрес
шлюза маршрутизатора перед пингованием его и сообщаящий этот MAC-адрес
PC-1, а затем запрашивающие MAC-адрес PC-1 и сообщающие его шлюзу.

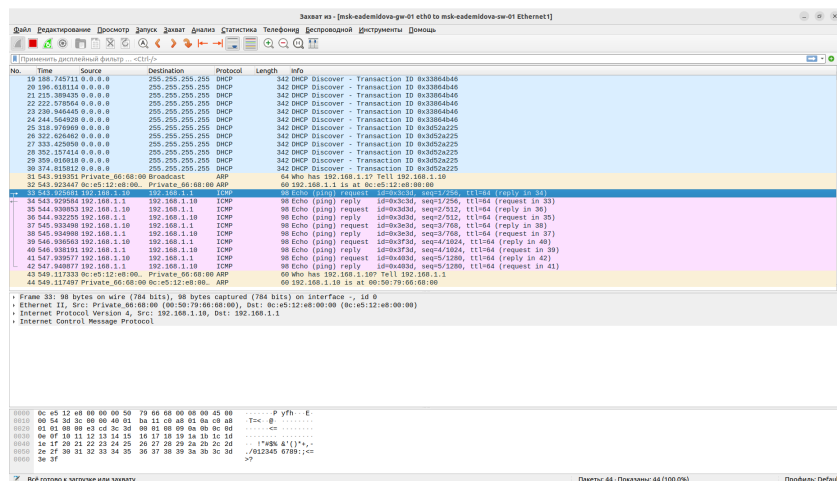


Рис. 3.28: Анализ трафика Wireshark

4 Выводы

В результате выполнения лабораторной работы были построены простейшие модели сети на базе коммутатора и маршрутизаторов FRR и VyOS в GNS3, а также проанализирован трафик посредством Wireshark.