

Лабораторная работа №5

Сетевые технологии

Андреева С.В.

Группа НПИбд-01-23

Российский университет дружбы народов, Москва, Россия

Информация

Докладчик

- Андреева Софья Владимировна
- Группа НПИбд-01-23
- Российский университет дружбы народов

Вводная часть

Цель работы

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

Выполнение лабораторной работы

Моделирование простейшей сети на базе коммутатора в GNS3

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-svandreeva-sw-01. Затем соединим VPCS с коммутатором и отобразим обозначение интерфейсов соединения

Выполнение лабораторной работы

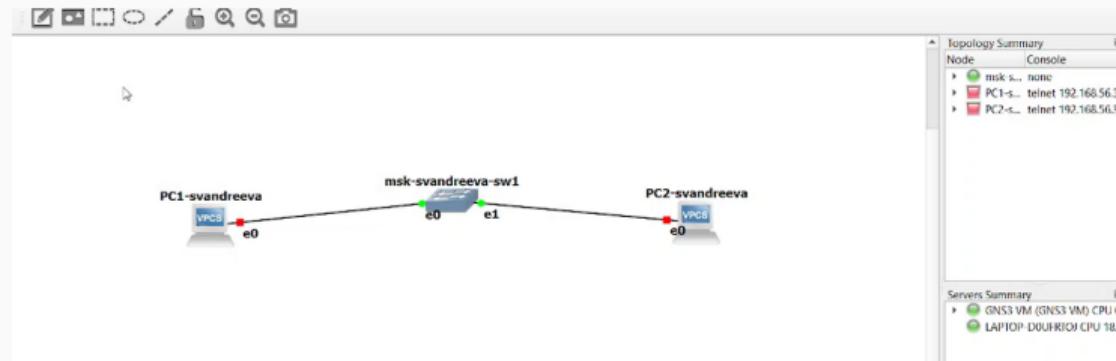


Рис. 1: Добавление устройств и изменение их имен Соединение VPCS с коммутатором

Выполнение лабораторной работы

Зададим IP-адреса VPCS. Для этого с помощью меню, вызываемого правой кнопкой мыши, запустим Start, PC-1, затем вызовим его терминал Console. Для просмотра синтаксиса возможных для ввода команд наберем /?

Выполнение лабораторной работы

```
PC1-svandreeva - PuTTY

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
dhcpp [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. 2: Параметры импорта

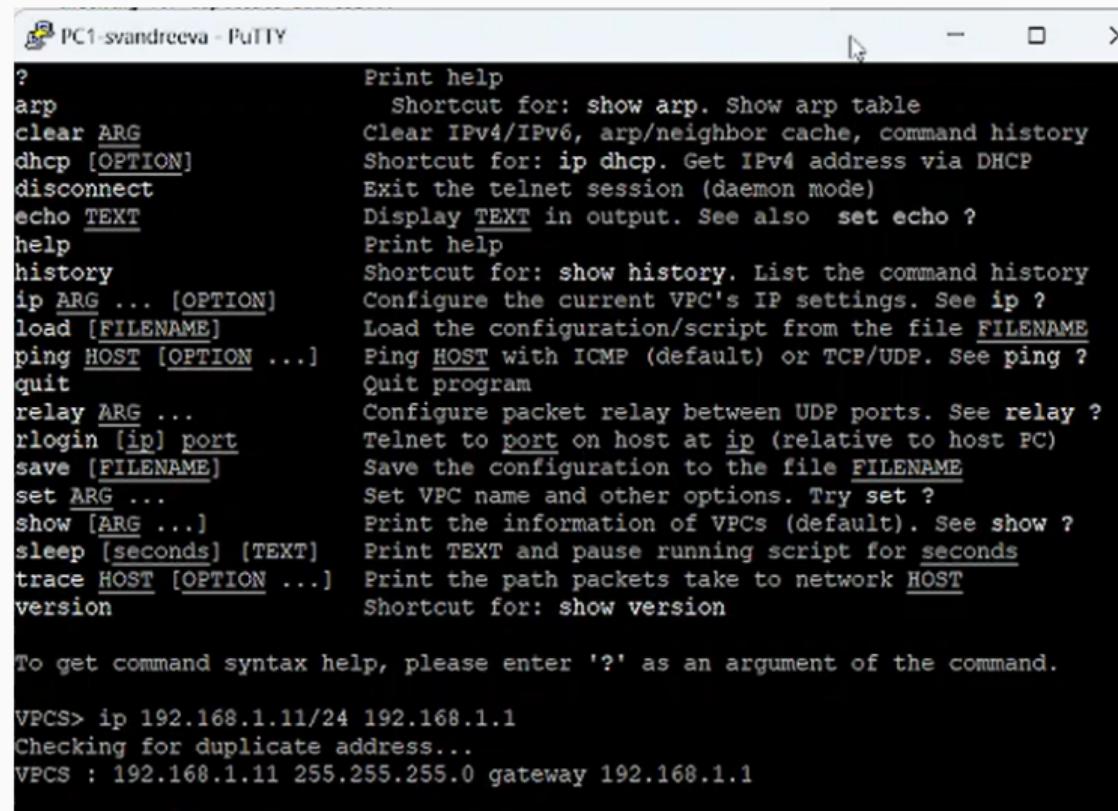
Выполнение лабораторной работы

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.

Выполнение лабораторной работы



```
PC1-svandreeva - PuTTY

?
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> 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. 3: Задание IP-адреса PC1-svandreeva

Выполнение лабораторной работы

Те же действия проделаем для второго VPCS

The screenshot shows a PuTTY terminal window with the title 'PC2-svandreeva - PuTTY'. The window displays the command-line interface of a VPCS device. At the top, there is a help menu listing various commands such as ip, load, ping, quit, relay, rlogin, save, set, show, sleep, trace, and version, each with a brief description. Below the help menu, a message instructs the user to enter '?' as an argument for command syntax help. The main session area shows the following sequence of commands:

```
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. 4: Задание IP-адреса PC2-svandreeva

Выполнение лабораторной работы

Проверим работоспособность соединения между PC-1 и PC-2 с помощью команды ping. В конце остановим в проекте все узлы(меню GNS3 Control Stop all nodes).

The image shows two terminal windows side-by-side. The left window is titled 'PC2-svandreeva - PUTTY' and displays the configuration of a VPC host. It shows commands like 'ip login [ip] port', 'save [FILENAME]', and 'show [ARG ...]'. Below these, it lists network interfaces: 'ipcs : 192.168.1.12 255.255.255.0 gateway 192.168.1.1'. The user then saves the configuration: 'VPCS> save' followed by 'Saving startup configuration to startup.vpc' and '. done'. Finally, a ping command is run: 'VPCS> ping 192.168.1.11', which returns several ICMP echo replies from the target IP.

The right window shows the output of the ping command. It starts with '64 bytes from 192.168.1.12 icmp_seq=1 ttl=64 time=0.874 ms', followed by three more replies. Then it continues with '84 bytes from 192.168.1.12 icmp_seq=2 ttl=64 time=0.537 ms', '84 bytes from 192.168.1.12 icmp_seq=3 ttl=64 time=0.433 ms', and five more replies. The output ends with '84 bytes from 192.168.1.12 icmp_seq=5 ttl=64 time=0.552 ms'.

Рис. 5: Проверка соединения между PC-1 и PC-2

Анализ трафика в GNS3 посредством Wireshark

Запустим на соединении между PC-1 и коммутатором анализатор трафика. Для этого щёлкнём правой кнопкой мыши на соединении, выберем в меню Start capture

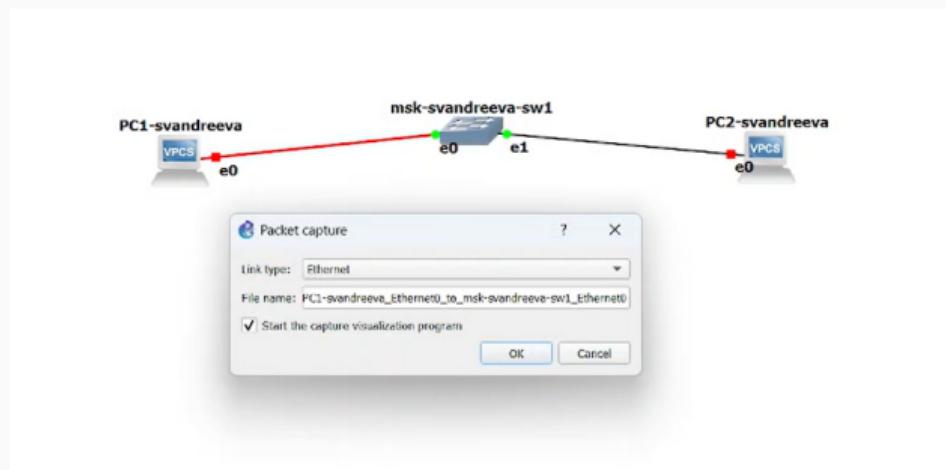


Рис. 6: Запуск анализатора трафика

Выполнение лабораторной работы

Запустился Wireshark, а в проекте GNS3 на соединении появился значок

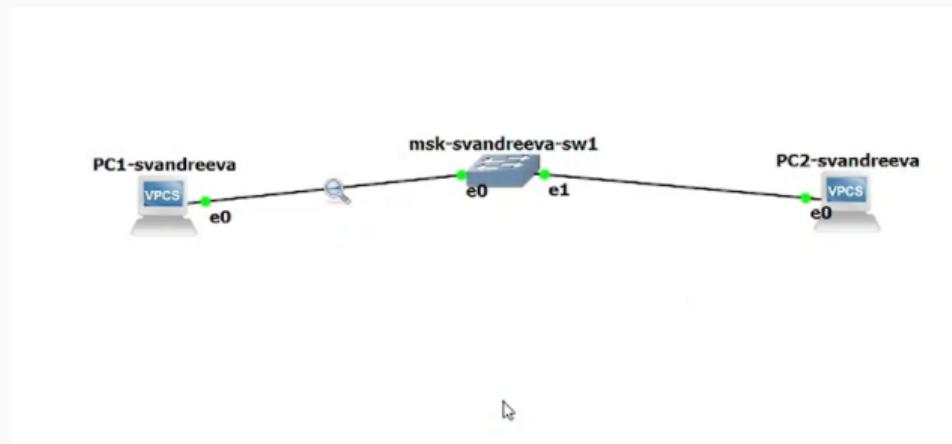


Рис. 7: ARP пакеты

Выполнение лабораторной работы

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. В обоих случаях длина кадра равняется 64 байт. В начале сформировались запросы безвоздмездных 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.

Выполнение лабораторной работы

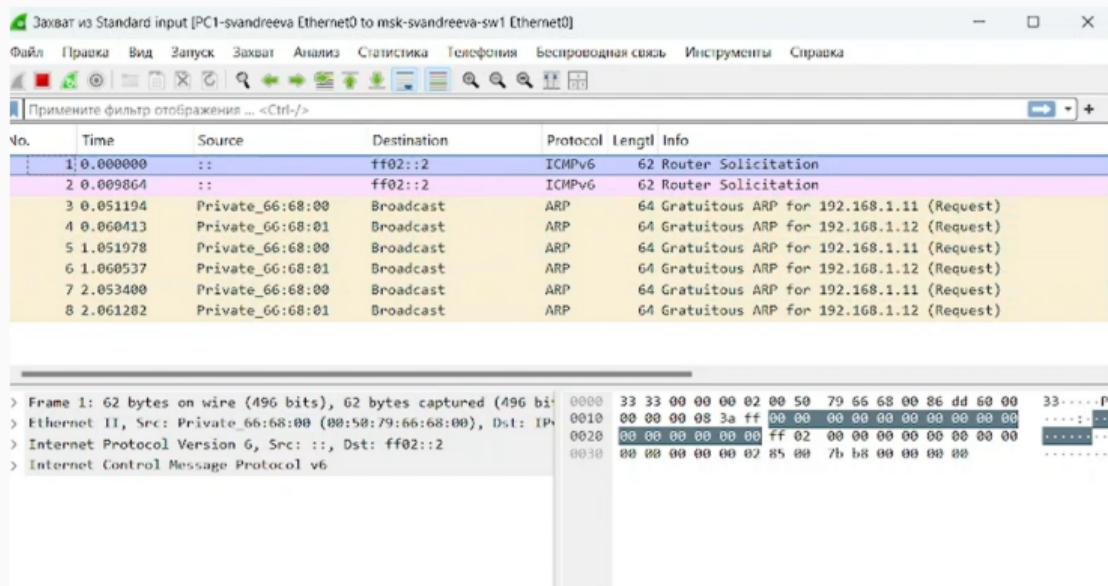


Рис. 8: ARP пакеты

Выполнение лабораторной работы

В терминале PC-2 посмотрим информацию по опциям команды ping, введя ping ?. Затем сделаем один эхо-запрос в ICMP-моде к узлу PC-1. Изучим эхо-запрос и эхо-ответ ICMP в программе Wireshark. 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.

Выполнение лабораторной работы

PC2-svandreeva - PuTTY

```
-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 -l
84 bytes from 192.168.1.11 icmp_seq=1 ttl=64 time=0.532 ms
84 bytes from 192.168.1.11 icmp_seq=2 ttl=64 time=1.029 ms
84 bytes from 192.168.1.11 icmp_seq=3 ttl=64 time=0.635 ms
84 bytes from 192.168.1.11 icmp_seq=4 ttl=64 time=0.600 ms
84 bytes from 192.168.1.11 icmp_seq=5 ttl=64 time=0.649 ms

VPCS>
```

Рис. 9: Эхо-запрос в ICMP-моде

Выполнение лабораторной работы

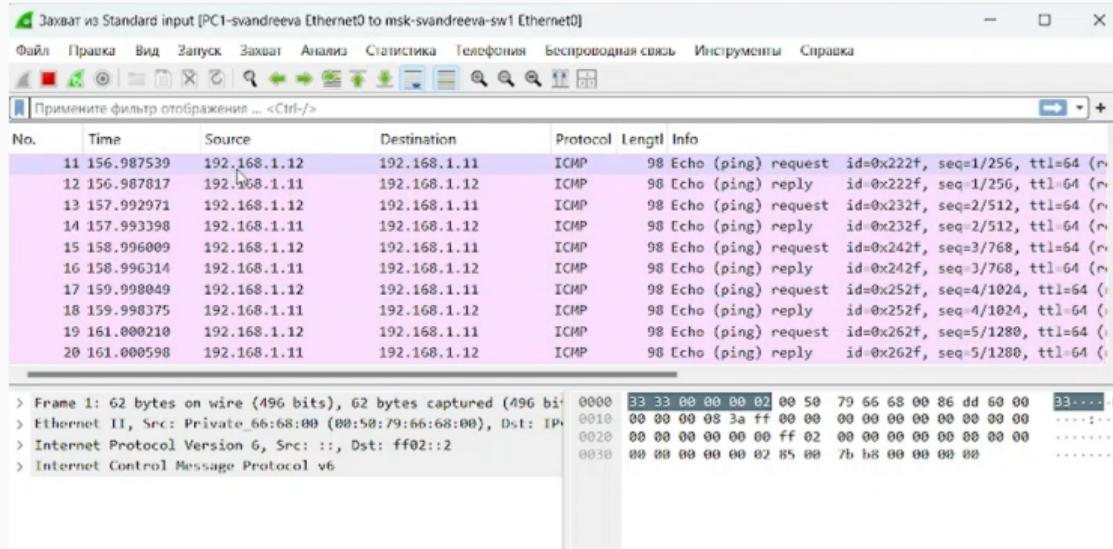
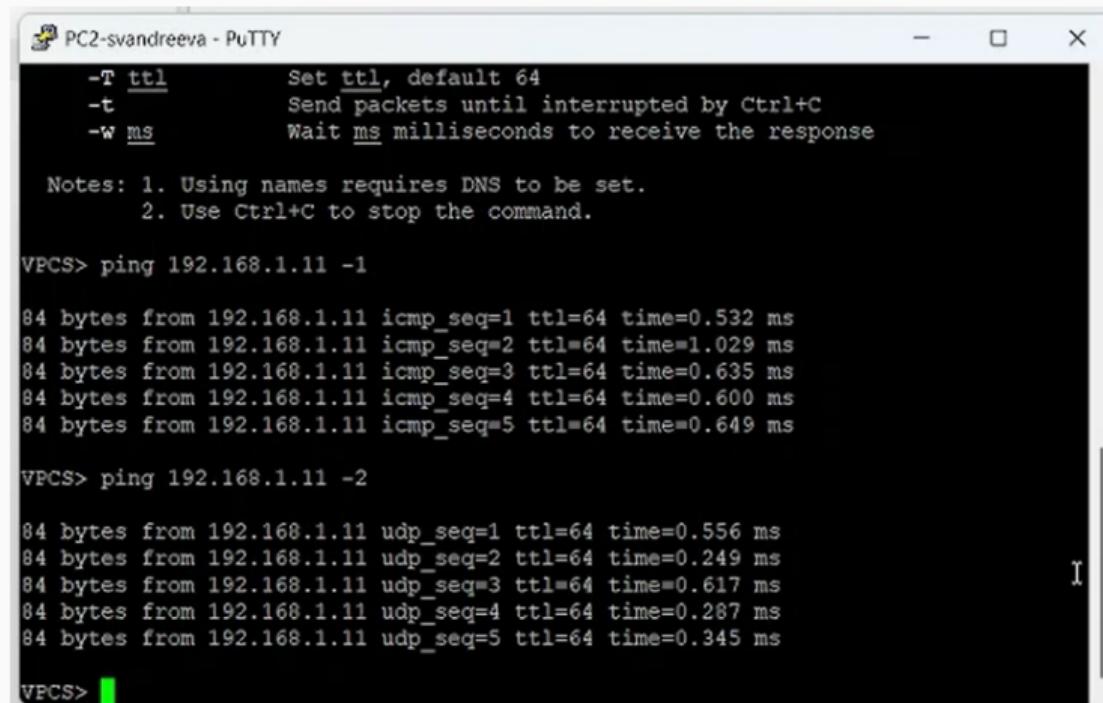


Рис. 10: ICMP пакеты

Выполнение лабораторной работы

Сделаем один эхо-запрос в UDP-моде к узлу PC-1. В окне Wireshark проанализируем полученную информацию. 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.

Выполнение лабораторной работы



The screenshot shows a PuTTY terminal window titled "PC2-svandreeva - PuTTY". The window displays a series of echo requests sent from a host to an IP address. The host's prompt is "VPCS>". The first set of requests is labeled "-1" and uses ICMP. The second set is labeled "-2" and uses UDP. Both sets show five consecutive requests with their sequence numbers (seq=1 to seq=5), Time-to-Live (ttl) of 64, and response times in milliseconds.

```
-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 -1

84 bytes from 192.168.1.11 icmp_seq=1 ttl=64 time=0.532 ms
84 bytes from 192.168.1.11 icmp_seq=2 ttl=64 time=1.029 ms
84 bytes from 192.168.1.11 icmp_seq=3 ttl=64 time=0.635 ms
84 bytes from 192.168.1.11 icmp_seq=4 ttl=64 time=0.600 ms
84 bytes from 192.168.1.11 icmp_seq=5 ttl=64 time=0.649 ms

VPCS> ping 192.168.1.11 -2

84 bytes from 192.168.1.11 udp_seq=1 ttl=64 time=0.556 ms
84 bytes from 192.168.1.11 udp_seq=2 ttl=64 time=0.249 ms
84 bytes from 192.168.1.11 udp_seq=3 ttl=64 time=0.617 ms
84 bytes from 192.168.1.11 udp_seq=4 ttl=64 time=0.287 ms
84 bytes from 192.168.1.11 udp_seq=5 ttl=64 time=0.345 ms

VPCS>
```

Рис. 11: Эхо-запрос в UDP-моде

Выполнение лабораторной работы

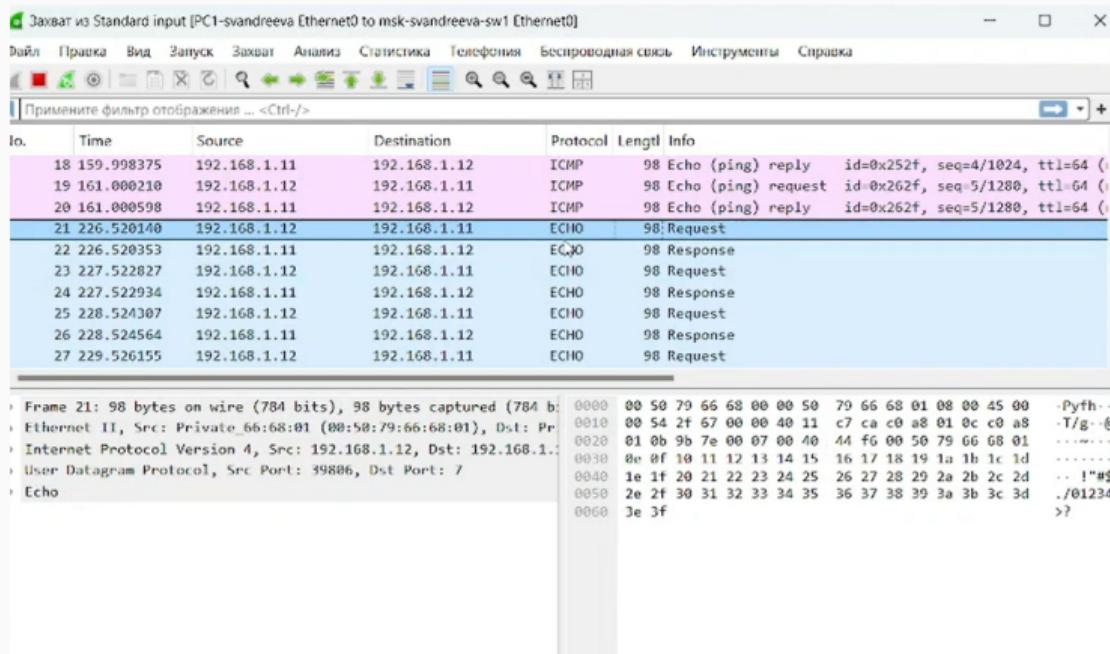
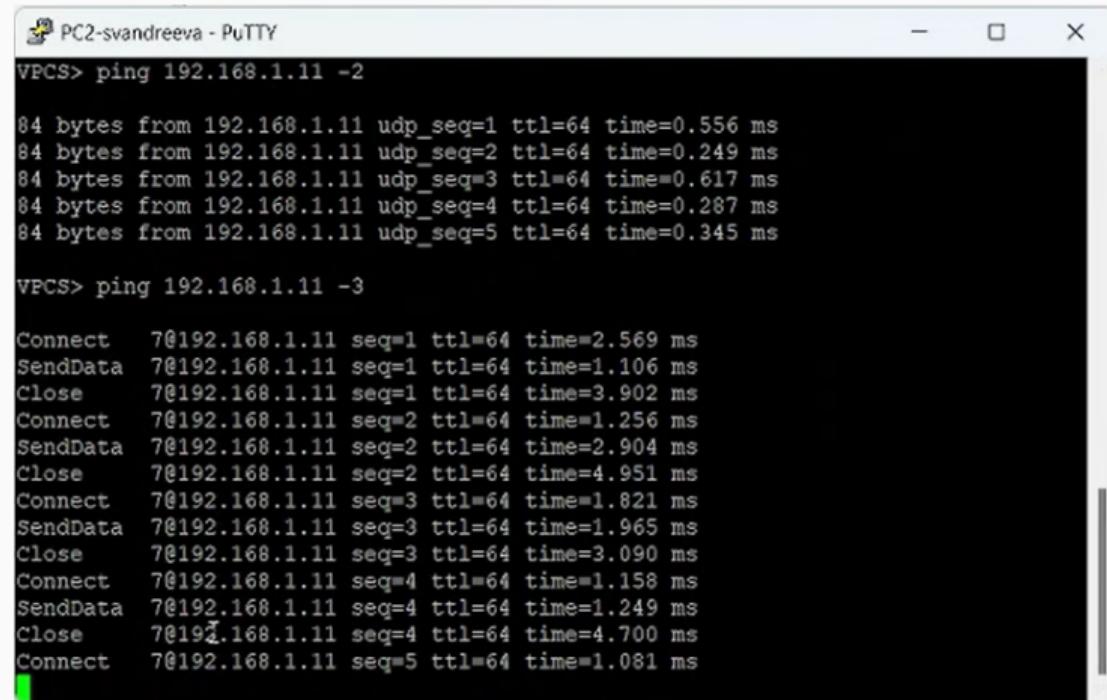


Рис. 12: UDP пакеты

Выполнение лабораторной работы

Сделаем один эхо-запрос в TCP-моде к узлу PC-1. В окне Wireshark проанализируем полученную информацию. 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, Acknowldgment: set). А в следующем пакете установлен бит ACK(Acknowledgment: set). Также есть пакеты с битом FIN, завершающим handshake.

Выполнение лабораторной работы



The screenshot shows a PuTTY terminal window titled "PC2-svandreeva - PuTTY". The session identifier "VPCS>" is visible at the top left. The terminal displays two sets of network traffic logs.

The first set of logs shows ICMP echo requests (ping) sent to 192.168.1.11:

```
VPCS> ping 192.168.1.11 -2
84 bytes from 192.168.1.11 udp_seq=1 ttl=64 time=0.556 ms
84 bytes from 192.168.1.11 udp_seq=2 ttl=64 time=0.249 ms
84 bytes from 192.168.1.11 udp_seq=3 ttl=64 time=0.617 ms
84 bytes from 192.168.1.11 udp_seq=4 ttl=64 time=0.287 ms
84 bytes from 192.168.1.11 udp_seq=5 ttl=64 time=0.345 ms
```

The second set of logs shows TCP echo requests (echo -3) sent to 192.168.1.11, showing the connection state transitions (Connect, SendData, Close) for each request:

```
VPCS> ping 192.168.1.11 -3
Connect 7@192.168.1.11 seq=1 ttl=64 time=2.569 ms
SendData 7@192.168.1.11 seq=1 ttl=64 time=1.106 ms
Close 7@192.168.1.11 seq=1 ttl=64 time=3.902 ms
Connect 7@192.168.1.11 seq=2 ttl=64 time=1.256 ms
SendData 7@192.168.1.11 seq=2 ttl=64 time=2.904 ms
Close 7@192.168.1.11 seq=2 ttl=64 time=4.951 ms
Connect 7@192.168.1.11 seq=3 ttl=64 time=1.821 ms
SendData 7@192.168.1.11 seq=3 ttl=64 time=1.965 ms
Close 7@192.168.1.11 seq=3 ttl=64 time=3.090 ms
Connect 7@192.168.1.11 seq=4 ttl=64 time=1.158 ms
SendData 7@192.168.1.11 seq=4 ttl=64 time=1.249 ms
Close 7@192.168.1.11 seq=4 ttl=64 time=4.700 ms
Connect 7@192.168.1.11 seq=5 ttl=64 time=1.081 ms
```

Рис. 13: Эхо-запрос в TCP-моде

Выполнение лабораторной работы

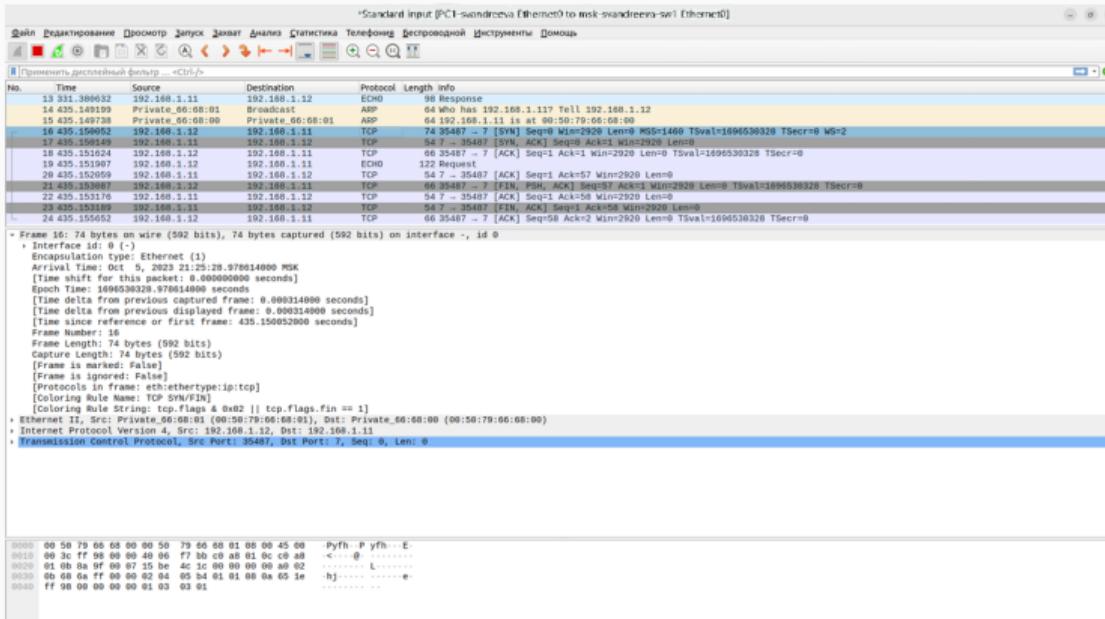


Рис. 14: TCP пакеты

Моделирование простейшей сети на базе маршрутизатора FRR в GNS3

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. Затем запустим все устройства проекта. Откроем консоль всех устройств проекта.

Выполнение лабораторной работы

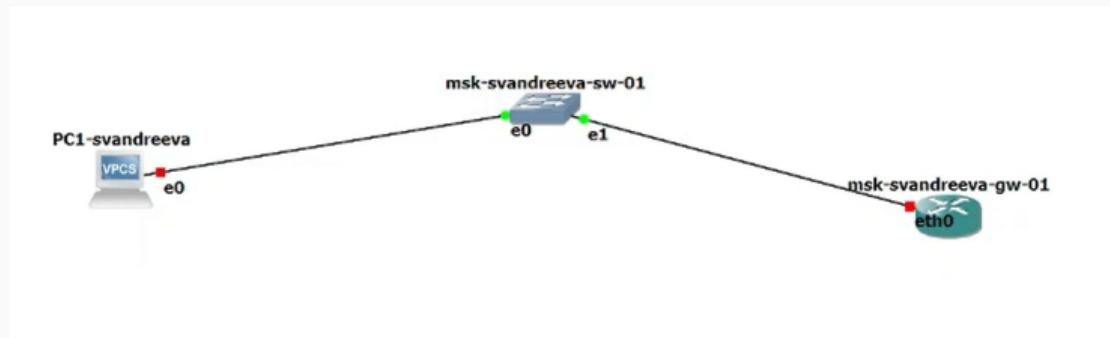


Рис. 15: Рабочая область

Выполнение лабораторной работы

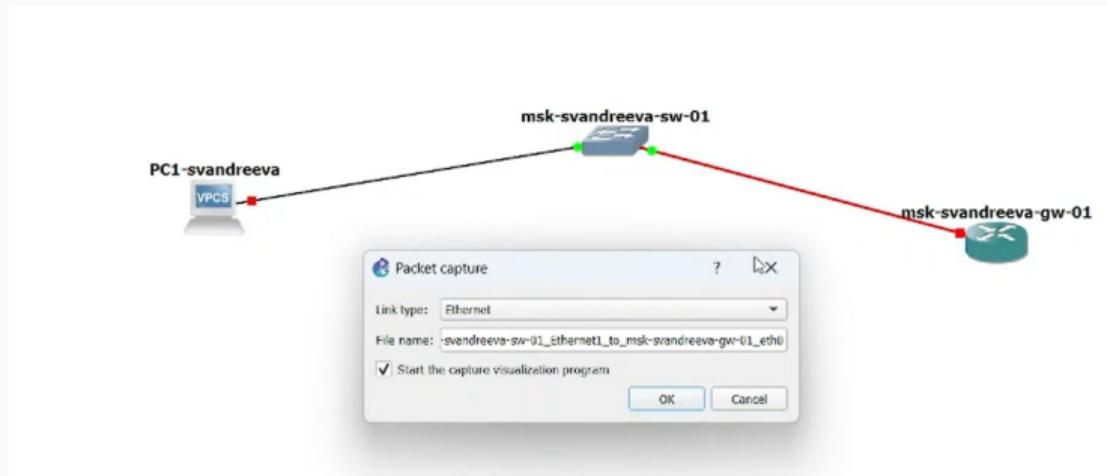
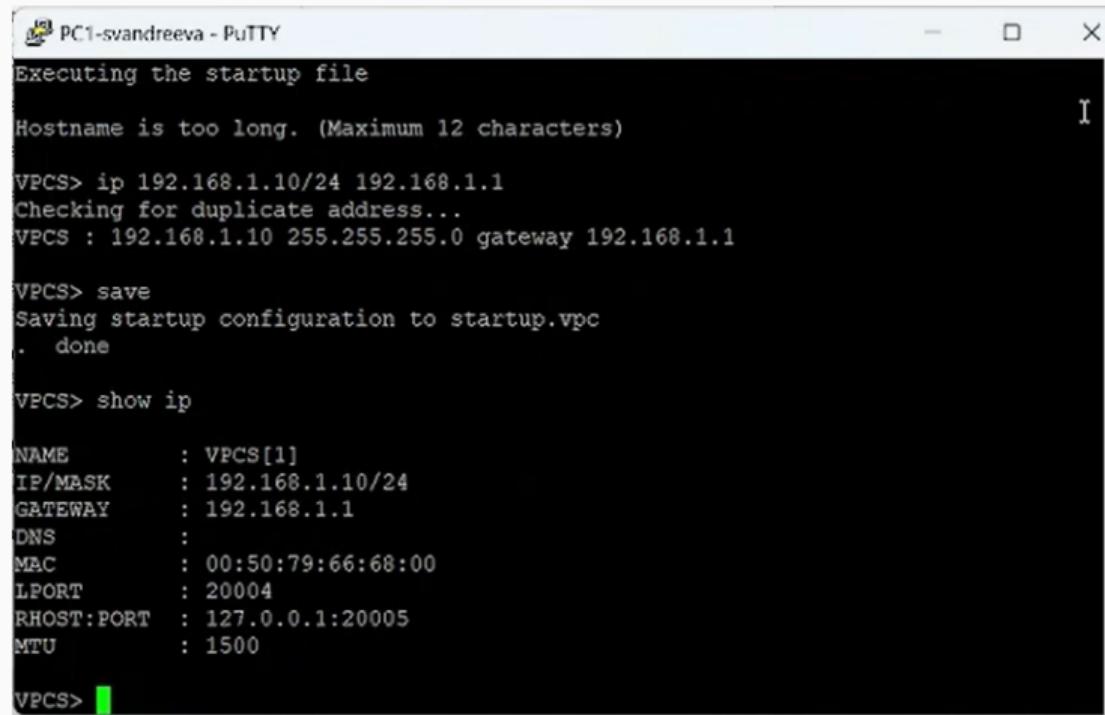


Рис. 16: Захват трафика

Выполнение лабораторной работы

Настроим IP-адресацию для интерфейса узла PC1:



```
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     : 20004
RHOST:PORT : 127.0.0.1:20005
MTU       : 1500

VPCS>
```

Рис. 17: Настройка IP-адресации для интерфейса узла PC1

Выполнение лабораторной работы

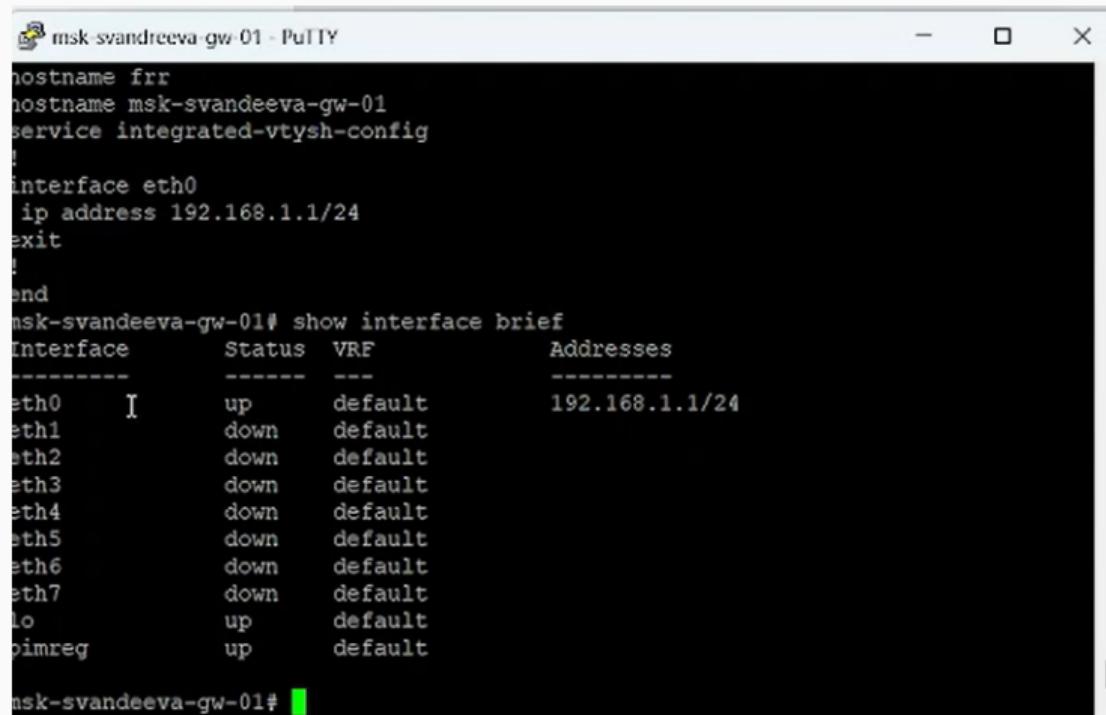
Настроим IP-адресацию для интерфейса локальной сети маршрутизатора:

```
msk-svandreeva-gw-01 - PuTTY

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

Выполнение лабораторной работы

Проверим конфигурацию маршрутизатора и настройки IP-адресации:

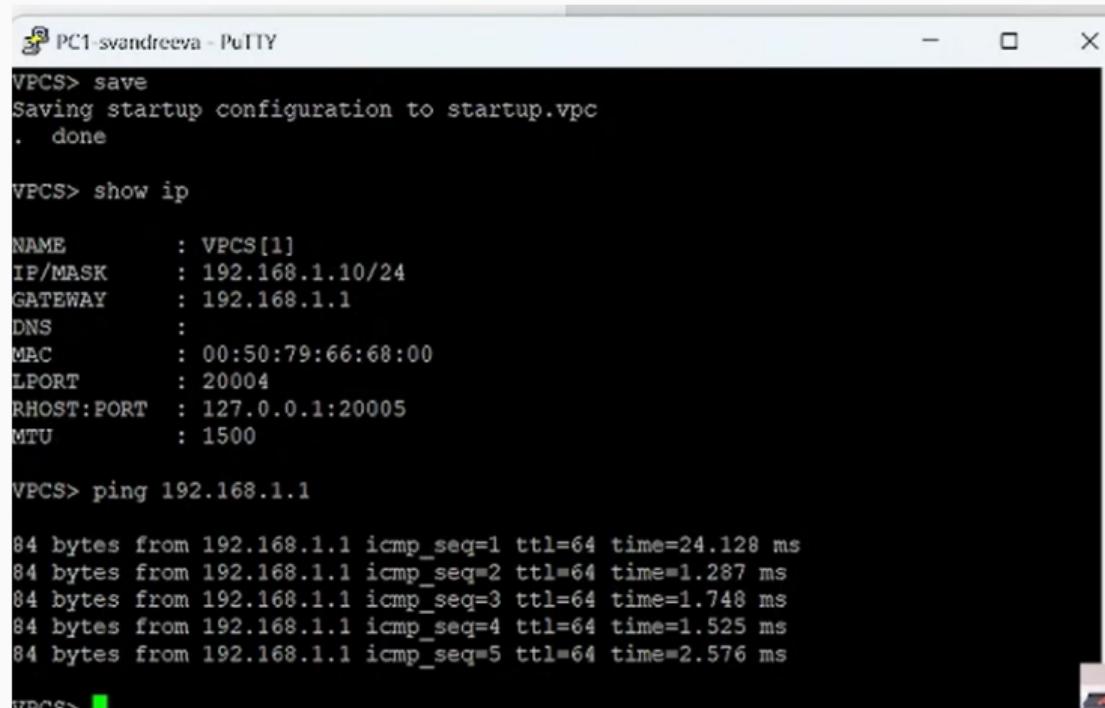


The screenshot shows a PuTTY terminal window titled "msk svandreeva-gw-01 - PuTTY". The session content displays the following Cisco IOS configuration and interface status:

```
hostname frr
hostname msk-svandreeva-gw-01
service integrated-vtysh-config
!
interface eth0
 ip address 192.168.1.1/24
exit
!
end
msk-svandreeva-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-svandreeva-gw-01#
```

Выполнение лабораторной работы

Проверим подключение. Узел PC1 успешно отправлять эхо-запросы ICMP на адрес маршрутизатора 192.168.1.1



```
PC1 svandreeva - PuTTY

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     : 20004
RHOST:PORT : 127.0.0.1:20005
MTU       : 1500

VPCS> ping 192.168.1.1

84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=24.128 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=1.287 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=1.748 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=1.525 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=2.576 ms

VPCS>
```

Выполнение лабораторной работы

В окне Wireshark проанализируем полученную информацию. Было отправлено 10 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 сообщающий этот MAC-адрес PC-1, а затем запрашивающие MAC-адрес PC-1 и сообщающие его шлюзу.

Выполнение лабораторной работы

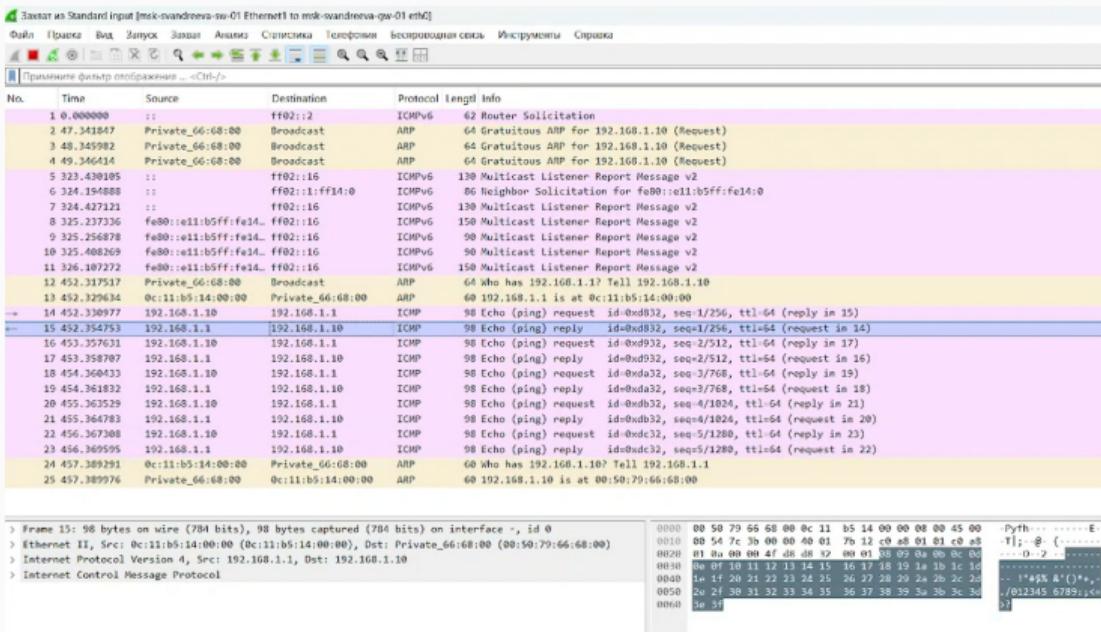


Рис. 21: Анализ информации в Wireshark

Моделирование простейшей сети на базе маршрутизатора VyOS в GNS3

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.

Выполнение лабораторной работы

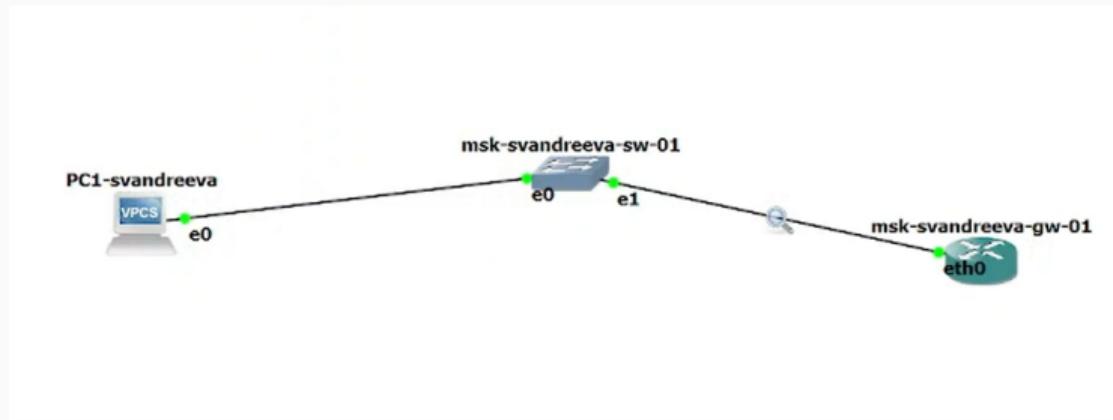
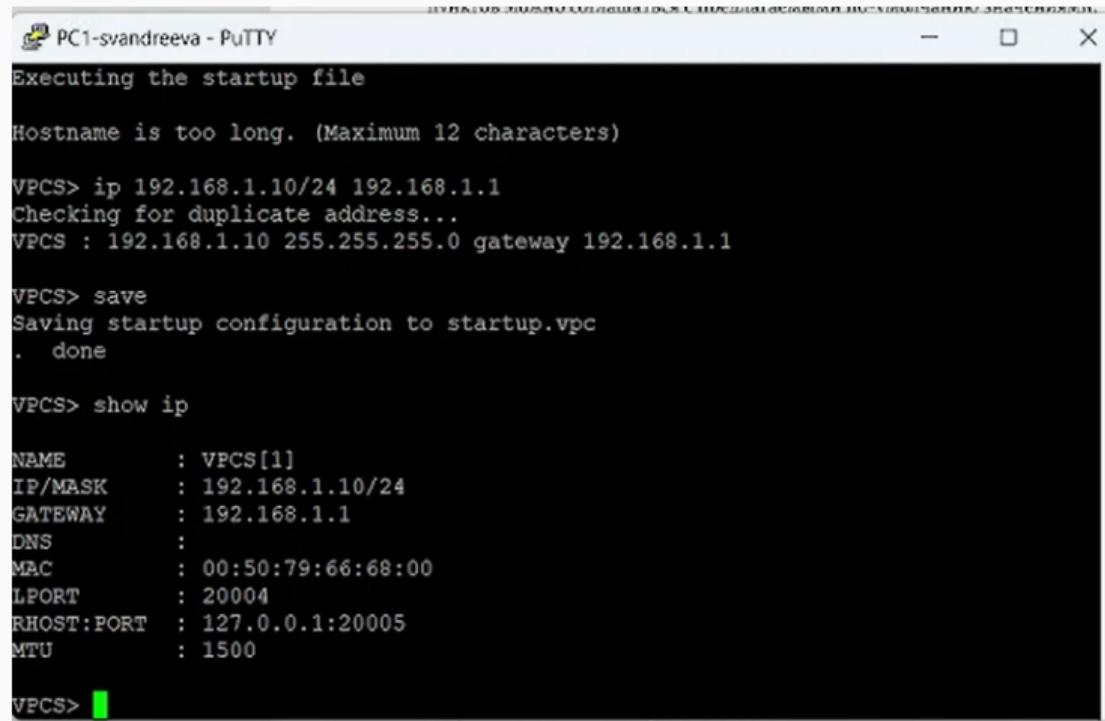


Рис. 22: Захват трафика на соединении между коммутатором и маршрутизатором

Выполнение лабораторной работы

Настроим IP-адресацию для интерфейса узла PC1.



```
PC1-svandreeva - PuTTY
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     : 20004
RHOST:PORT:PORT : 127.0.0.1:20005
MTU       : 1500

VPCS>
```

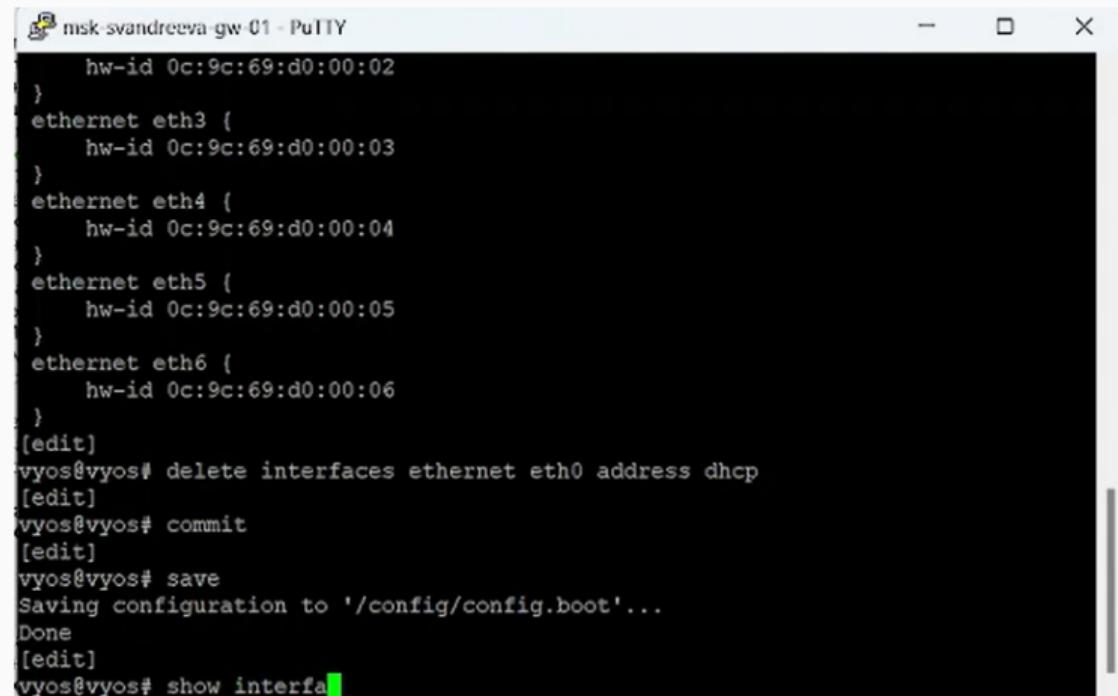
Выполнение лабораторной работы

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-svandreeva-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 просмотрим внесённые в конфигурацию изменения, применим эти изменения и сохраним. В конце выйдем из режима конфигурирования

Выполнение лабораторной работы

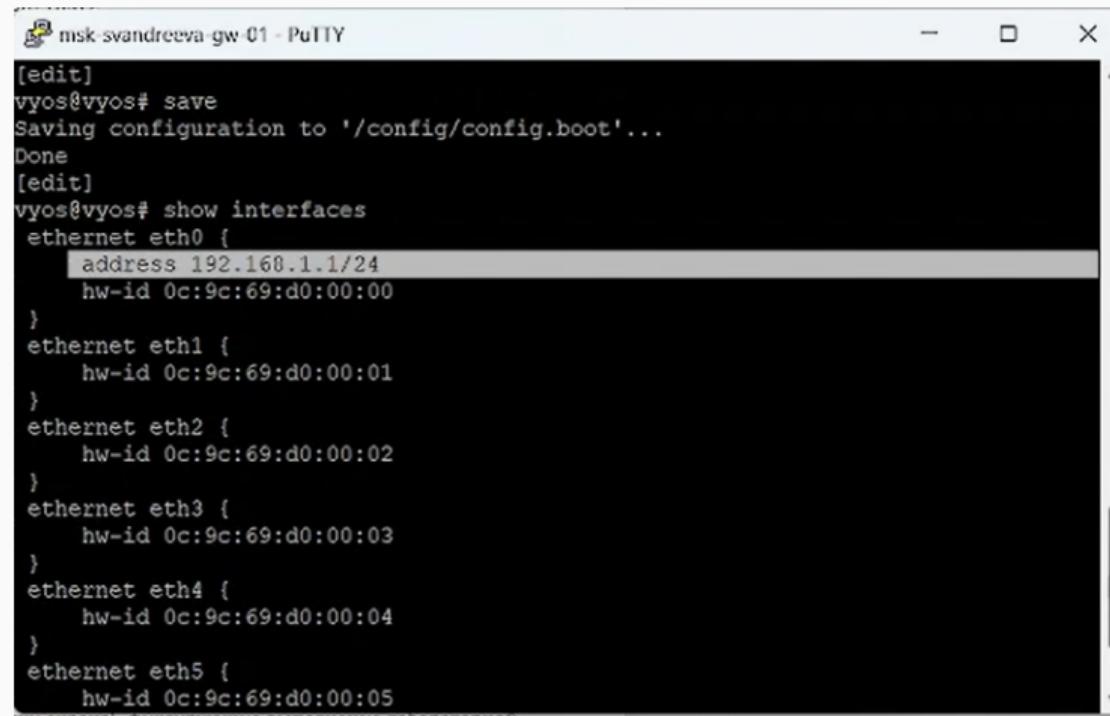


The screenshot shows a PuTTY terminal window titled "msk-svandreeva-gw-01 - PuTTY". The window displays the following command-line session:

```
hw-id 0c:9c:69:d0:00:02
}
ethernet eth3 {
    hw-id 0c:9c:69:d0:00:03
}
ethernet eth4 {
    hw-id 0c:9c:69:d0:00:04
}
ethernet eth5 {
    hw-id 0c:9c:69:d0:00:05
}
ethernet eth6 {
    hw-id 0c:9c:69:d0:00:06
}
[edit]
vyos@vyos# delete interfaces ethernet eth0 address dhcp
[edit]
vyos@vyos# commit
[edit]
vyos@vyos# save
Saving configuration to '/config/config.boot'...
Done
[edit]
vyos@vyos# show interfa
```

Рис. 24: Режим конфигурации маршрутизатора VyOS

Выполнение лабораторной работы



The screenshot shows a PuTTY terminal window titled "msk-svandreeva-gw-01 - PuTTY". The session icon is a yellow gear. The window contains the following text:

```
[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:9c:69:d0:00:00
}
ethernet eth1 {
    hw-id 0c:9c:69:d0:00:01
}
ethernet eth2 {
    hw-id 0c:9c:69:d0:00:02
}
ethernet eth3 {
    hw-id 0c:9c:69:d0:00:03
}
ethernet eth4 {
    hw-id 0c:9c:69:d0:00:04
}
ethernet eth5 {
    hw-id 0c:9c:69:d0:00:05
```

Рис. 25: Режим конфигурации маршрутизатора VyOS

Выполнение лабораторной работы

Проверим подключение. Узел PC1 успешно отправлять эхо-запросы на адрес маршрутизатора 192.168.1.1

```
PC1-svandreeva - PuTTY
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     : 20004
RHOST:PORT : 127.0.0.1:20005
MTU       : 1500

VPCS> ping 192.168.1.1
I
84 bytes from 192.168.1.1 icmp_seq=1 ttl=64 time=1.729 ms
84 bytes from 192.168.1.1 icmp_seq=2 ttl=64 time=2.294 ms
84 bytes from 192.168.1.1 icmp_seq=3 ttl=64 time=2.903 ms
84 bytes from 192.168.1.1 icmp_seq=4 ttl=64 time=2.179 ms
84 bytes from 192.168.1.1 icmp_seq=5 ttl=64 time=4.427 ms

VPCS>
```

Выполнение лабораторной работы

В окне Wireshark проанализируйте полученную информацию. Было отправлены пакеты 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 сообщающий этот MAC-адрес PC-1, а затем запрашивающие MAC-адрес PC-1 и сообщающие его шлюзу.

Выполнение лабораторной работы

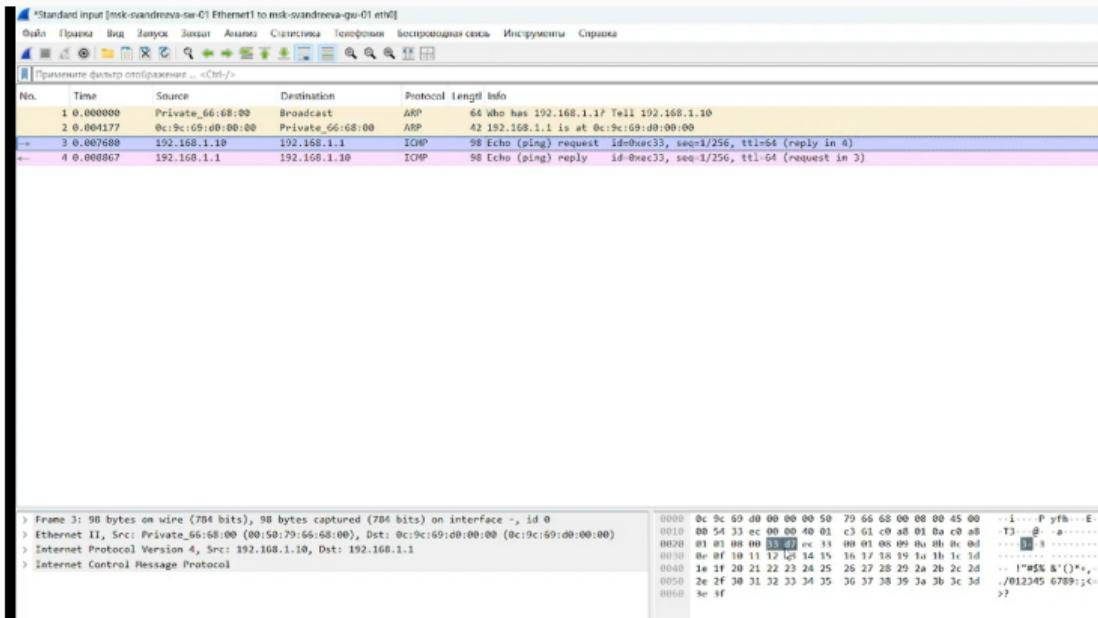


Рис. 27: Анализ трафика Wireshark

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.