

Отчет по лабораторной работе №6

Дисциплина: Сетевые технологии

Иванов Сергей Владимирович

Содержание

1 Цель работы	4
2 Выполнение лабораторной работы	5
2.1 Разбиение IPv4-сети на подсети	5
2.2 Разбиение IPv6-сети на подсети	7
2.3 Настройка двойного стека адресации IPv4 и IPv6 в локальной сети	9
2.4 Задание для самостоятельного выполнения	24
3 Выводы	28

Список иллюстраций

2.1 Создание сети	10
2.2 Запуск Wireshark	10
2.3 Настройка PC1	11
2.4 Настройка PC2	11
2.5 Настройка Server	11
2.6 Просмотр конфигураций	12
2.7 Просмотр конфигураций	12
2.8 Настройка FRR	14
2.9 Конфигурация FRR	14
2.10 Проверка подключения с PC1	15
2.11 Проверка подключения с PC2	15
2.12 Настройка PC3	16
2.13 Настройка PC4	16
2.14 Настройка Server	17
2.15 Просмотр конфигураций	17
2.16 Просмотр конфигураций	18
2.17 Установка системы на VyOS	18
2.18 Конфигурация VyOS	19
2.19 Конфигурация VyOS	20
2.20 Проверка подключения с PC3	20
2.21 Проверка подключения с PC4	21
2.22 Проверка недоступности устройств IPv4 и IPv6 между собой	21
2.23 Обращение сервера к устройствам обеих подсетей	22
2.24 Трафик ARP	22
2.25 Трафик ICMP	23
2.26 Трафик ICMPv6	23
2.27 Характеристика и вариант таблицы адресации	24
2.28 Построение сети	24
2.29 Маршрутизация на PC1	25
2.30 Маршрутизация на PC2	25
2.31 Маршрутизация на VyOS	25
2.32 Настройки VyOS	26
2.33 Проверка подключения с PC1	26
2.34 Проверка подключения с PC2	27

1 Цель работы

Изучение принципов распределения и настройки адресного пространства на устройствах сети.

2 Выполнение лабораторной работы

2.1 Разбиение IPv4-сети на подсети

- 1) Задана IPv4-сеть 172.16.20.0/24. Для заданной сети определим префикс, маску, broadcast-адрес, число возможных подсетей, диапазон адресов узлов. Разобьем сеть на 3 подсети с максимально возможным числом адресов узлов 126, 62, 62 соответственно.

Префикс: /24

Маска: 11111111.11111111.11111111.00000000 в двоичной записи или
255.255.255.0

Диапазон адресов узлов: $2^{(32 - 24)} = 2^8 = 256 \Rightarrow 172.16.20.1 - 172.16.20.254$

broadcast: 172.16.20.255

Разделение на сети:

1. 126 узлов: ближайшая степень 2: $2^7 = 128$. $128 - 2$ (broadcast и адрес сети) как раз 126.

$32 - 7 = 25$, значит маска сети 11111111.11111111.11111111.10000000 или
255.255.255.128

Адрес сети: 172.16.20.0/25

Диапазон адресов узлов: $2^7 = 128 \Rightarrow 172.16.20.1 - 172.16.20.126$

broadcast: 172.16.20.127

2. 62 узла: ближайшая степень 2: $2^6 = 64$. $64 - 2$ (broadcast и адрес сети) как раз 62.

$32 - 6 = 26$, значит маска сети 11111111.11111111.11111111.11000000 или
255.255.255.192

После 172.16.20.127/25 следующая свободная подсеть 172.16.20.128/25.

Поделим ее на две /26: 172.16.20.128/26 и 172.16.20.192/26. Берем сеть 172.16.20.128/26.

Диапазон адресов узлов: 172.16.20.129 – 172.16.20.190
broadcast: 172.16.20.191

3. 62 узла: аналогично, только берем адрес сети 172.16.20.192/26 .

Диапазон адресов узлов: 172.16.20.193 – 172.16.20.254
broadcast: 172.16.20.255

2) Задана сеть 10.10.1.64/26. Для заданной сети определим префикс, маску, broadcast-адрес, число возможных подсетей, диапазон адресов узлов. Выделим в этой сети подсеть на 30 узлов. Запишем характеристики для выделенной подсети.

Префикс: /26

Маска: 11111111.11111111.11111111.11000000 в двоичной записи или
255.255.255.192

Диапазон адресов узлов: $2^{(32 - 26)} = 2^6 = 64 \Rightarrow 10.10.1.65 - 10.10.1.126$
broadcast: 10.10.1.127

Подсеть на 30 узлов:

ближайшая степень 2: $2^5 = 32$. $32 - 2$ (broadcast и адрес сети) как раз 30.
 $32 - 5 = 27$, значит маска сети 11111111.11111111.11111111.11100000 или
255.255.255.224

Делим 10.10.1.64/26 на две подсети 10.10.1.64/27 и 10.10.1.96/27

Берем адрес сети: 10.10.1.64/27

Диапазон адресов узлов: 10.10.1.65 – 10.10.1.94
broadcast: 10.10.1.95

3) Задана сеть 10.10.1.0/26. Для этой сети определим префикс, маску, broadcast-адрес, число возможных подсетей, диапазон адресов узлов. Выделим в этой сети подсеть на 14 узлов. Запишем характеристики для выделенной подсети.

Префикс: /26

Маска: 11111111.11111111.11111111.11000000 в двоичной записи или
255.255.255.192

Диапазон адресов узлов: $2^{(32 - 26)} = 2^6 = 64 \Rightarrow 10.10.1.1 - 10.10.1.62$

broadcast: 10.10.1.63

Подсеть на 14 узлов:

ближайшая степень 2: $2^4 = 16$. 16 – 2 (broadcast и адрес сети) как раз 14.

$32 - 4 = 28$, значит маска сети 11111111.11111111.11111111.11110000 или
255.255.255.240

Делим 10.10.1.0/26 на 4 подсети

- 10.10.1.0/28
- 10.10.1.14/28
- 10.10.1.28/28
- 10.10.1.42/28

Берем адрес сети: 10.10.1.0/28

Диапазон адресов узлов: 10.10.1.1 – 10.10.1.12

broadcast: 10.10.1.13

2.2 Разбиение IPv6-сети на подсети

1) Задана сеть 2001:db8:c0de::/48. Охарактеризуем адрес, определим маску, префикс, диапазон адресов для узлов сети (краевые значения). Разобьем сеть на 2 подсети двумя способами — с использованием идентификатора подсети и с использованием идентификатора интерфейса.

2001:db8:c0de::/48 – Зарезервированы для документации и примеров (2001:DB8::/32).

Префикс: /48

Маска: 48 единиц, остальные нули. ffff:ffff:ffff::

Диапазон адресов сети: 2001:db8:c0de:0000:0000:0000:0000 - 2001:db8:c0de:ffff:ffff:ffff:ffff

Разбивка на 2 подсети

1. С использованием идентификатора подсети:

Исходная сеть: 2001:db8:c0de:0000:0000:0000:0000. Добавляем к префиксу +1 ($48 + 1 = 49$).

Подсеть 1: 2001:db8:c0de::/49. Диапазон: 2001:db8:c0de:: - 2001:db8:c0de:7fff:ffff:ffff:ffff:ffff

Подсеть 2: 2001:db8:c0de:8::/49 т.к ($000016 = 0000\ 0000\ 0000\ 0002 \Rightarrow 1000\ 0000\ 0000\ 0002 = 800016$). Диапазон: 2001:db8:c0de:8000:0000:0000:0000 - 2001:db8:c0de:ffff:ffff:ffff:ffff

2. С использованием идентификатора интерфейса:

Исходная сеть: 2001:db8:c0de:0000:0000:0000:0000

Разбиваем на 2 подсети /64: 2001:db8:c0de:0000::/64. Добавляем к префиксу +1: $64 + 1 = 65$.

Подсеть 1: 2001:db8:c0de::/65. Диапазон: 2001:db8:c0de:: - 2001:db8:c0de:0000:7fff:ffff:ffff:ffff

Подсеть 2: 2001:db8:c0de:0000:8::/65 т.к ($000016 = 0000\ 0000\ 0000\ 0002 \Rightarrow 1000\ 0000\ 0000\ 0002 = 800016$). Диапазон: 2001:db8:c0de:0000:8:: - 2001:db8:c0de:0000:ffff:ffff:ffff:ffff

2) Задана сеть 2a02:6b8::/64. Охарактеризуем адрес, определим маску, префикс, диапазон адресов для узлов сети (краевые значения). Разобьем сеть на 2 подсети двумя способами — с использованием идентификатора подсети и с использованием идентификатора интерфейса.

2a02:6b8::/64 - Глобальный Unicast-адрес (префикс 2000::/3)

Префикс: /64

Маска: 64 единицы, остальные нули. ffff:ffff:ffff:ffff::

Диапазон адресов сети: 2a02:6b8:: - 2a02:6b8:0000:0000:ffff:ffff:ffff:ffff

Разбивка на 2 подсети:

- 1) С использованием идентификатора подсети:

Исходная сеть: 2a02:6b8::/64

Добавляем к префиксу +1 ($64 + 1 = 65$).

Подсеть 1: 2a02:6b8::/65. Диапазон: 2a02:6b8:: - 2a02:6b8:0000:0000:7fff:ffff:ffff:ffff

Подсеть 2: 2a02:6b8:0000:0000:8::/65 т.к ($000016 = 0000\ 0000\ 0000\ 0002$

=> $1000\ 0000\ 0000\ 0002 = 800016$). Диапазон: 2a02:6b8:0000:0000:8:: -
2a02:6b8:0000:0000:ffff:ffff:ffff:ffff

- 2) С использованием идентификатора интерфейса:

Исходная сеть: 2a02:6b8::/64

Разбиваем на 2 подсети /64

Добавляем к префиксу +1: $64 + 1 = 65$.

Подсеть 1: 2a02:6b8::/65. Диапазон: 2a02:6b8:: - 2a02:6b8:0000:0000:7fff:ffff:ffff:ffff

Подсеть 2: 2a02:6b8:0000:0000:8::/65 т.к ($000016 = 0000\ 0000\ 0000\ 0002$

=> $1000\ 0000\ 0000\ 0002 = 800016$). Диапазон: 2a02:6b8:0000:0000:8:: -
2a02:6b8:0000:0000:ffff:ffff:ffff:ffff

2.3 Настройка двойного стека адресации IPv4 и IPv6 в локальной сети

Запустим GNS3 VM и GNS3. Создадим новый проект. В рабочем пространстве разместим и соединим устройства в соответствии с топологией. Для подсети IPv4 используем маршрутизатор FRR, а для подсети с IPv6 — маршрутизатор VyOS. Изменим отображаемые названия устройств. Коммутаторам присвоим

названия по принципу msk-user-sw-0x, маршрутизаторам — по принципу msk-user-gw-0x, VPCS — по принципу PCx-user. (рис. 1)

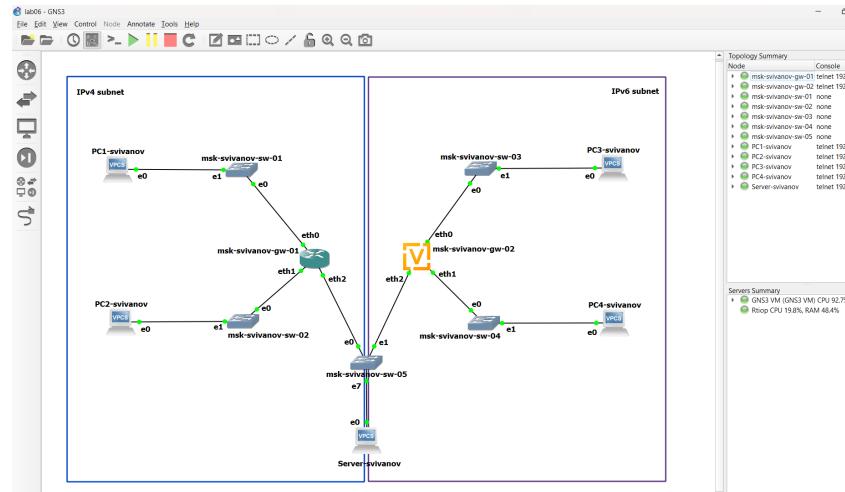


Рис. 2.1: Создание сети

Включим захват трафика на соединении между сервером двойного стека адресации и ближайшим к нему коммутатором. (рис. 2)

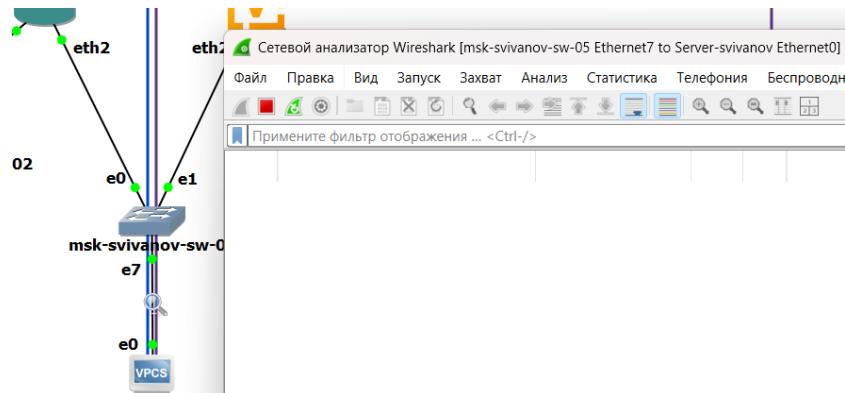


Рис. 2.2: Запуск Wireshark

Настроим IPv4-адресацию для интерфейсов узлов PC1, PC2, Server:
PC1: (рис. 3)

```
ip 172.16.20.10/25 172.16.20.1
save
```

```
PC1-svivanov> ip 172.16.20.10/25 172.16.20.1
Checking for duplicate address...
PC1-svivanov : 172.16.20.10 255.255.255.128 gateway 172.16.20.1

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

PC1-svivanov> █
```

Рис. 2.3: Настройка PC1

PC2: (рис. 4)

```
ip 172.16.20.138/25 172.16.20.129
save
```

```
PC2-svivanov> ip 172.16.20.138/25 172.16.20.129
Checking for duplicate address...
PC2-svivanov : 172.16.20.138 255.255.255.128 gateway 172.16.20.129

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

PC2-svivanov> █
```

Рис. 2.4: Настройка PC2

Server: (рис. 5)

```
ip 64.100.1.10/24 64.100.1.1
save
```

```
Server-svivanov - PuTTY
Hostname is too long. (Maximum 12 characters)

VPCS> ip 64.100.1.10/24 64.100.1.1
Checking for duplicate address...
VPCS : 64.100.1.10 255.255.255.0 gateway 64.100.1.1

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

VPCS> █
```

Рис. 2.5: Настройка Server

Посмотрим на PC1 и PC2 конфигурацию IPv4 и IPv6: (рис. 6, 7)

```
show ip  
show ipv6
```

```
PC1-svivanov> show ip  
  
NAME : PC1-svivanov[1]  
IP/MASK : 172.16.20.10/25  
GATEWAY : 172.16.20.1  
DNS :  
MAC : 00:50:79:66:68:00  
LPORT : 20022  
RHOST:PORT : 127.0.0.1:20023  
MTU : 1500  
  
PC1-svivanov> show ipv6  
  
NAME : PC1-svivanov[1]  
LINK-LOCAL SCOPE : fe80::250:79ff:fe66:6800/64  
GLOBAL SCOPE :  
DNS :  
ROUTER LINK-LAYER :  
MAC : 00:50:79:66:68:00  
LPORT : 20022  
RHOST:PORT : 127.0.0.1:20023  
MTU : 1500  
  
PC1-svivanov> █
```

Рис. 2.6: Просмотр конфигураций

```
PC2-svivanov> show ip  
  
NAME : PC2-svivanov[1]  
IP/MASK : 172.16.20.138/25  
GATEWAY : 172.16.20.129  
DNS :  
MAC : 00:50:79:66:68:01  
LPORT : 20024  
RHOST:PORT : 127.0.0.1:20025  
MTU : 1500  
  
PC2-svivanov> show ipv6  
  
NAME : PC2-svivanov[1]  
LINK-LOCAL SCOPE : fe80::250:79ff:fe66:6801/64  
GLOBAL SCOPE :  
DNS :  
ROUTER LINK-LAYER :  
MAC : 00:50:79:66:68:01  
LPORT : 20024  
RHOST:PORT : 127.0.0.1:20025  
MTU : 1500  
  
PC2-svivanov> █
```

Рис. 2.7: Просмотр конфигураций

Настроим IPv4-адресацию для интерфейсов локальной сети маршрутизатора FRR msk-user-gw-01: (рис. 8)

```
frr# configure terminal
frr(config)# hostname msk-user-gw-01
msk-user-gw-01(config)# exit
msk-user-gw-01# write memory
msk-user-gw-01# configure terminal
msk-user-gw-01(config)# interface eth0
msk-user-gw-01(config-if)# ip address 172.16.20.1/25
msk-user-gw-01(config-if)# no shutdown
msk-user-gw-01(config-if)# exit
msk-user-gw-01(config)# interface eth1
msk-user-gw-01(config-if)# ip address 172.16.20.129/25
msk-user-gw-01(config-if)# no shutdown
msk-user-gw-01(config-if)# exit
msk-user-gw-01(config)# interface eth2
msk-user-gw-01(config-if)# ip address 64.100.1.1/24
msk-user-gw-01(config-if)# no shutdown
msk-user-gw-01(config-if)# exit
msk-user-gw-01(config)# exit
msk-user-gw-01# write memory
```

```

msk-svivanov-gw-01# configure terminal
msk-svivanov-gw-01(config)# interface eth0
msk-svivanov-gw-01(config-if)# ip address 172.16.20.1/25
msk-svivanov-gw-01(config-if)# no shutdown
msk-svivanov-gw-01(config-if)# exit
msk-svivanov-gw-01(config)# interface eth1
msk-svivanov-gw-01(config-if)# ip address 172.16.20.129/25
msk-svivanov-gw-01(config-if)# no shutdown
msk-svivanov-gw-01(config-if)# exit
msk-svivanov-gw-01(config)# interface eth2
msk-svivanov-gw-01(config-if)# ip address 64.100.1.1/24
msk-svivanov-gw-01(config-if)# no shutdown
msk-svivanov-gw-01(config-if)# exit
msk-svivanov-gw-01(config)# exit
msk-svivanov-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-svivanov-gw-01# 

```

Рис. 2.8: Настройка FRR

Проверим конфигурацию маршрутизатора и настройки IPv4-адресации: (рис. 9)

```

msk-user-gw-01# show running-config
msk-user-gw-01# show interface brief

```

```

interface eth1
  ip address 172.16.20.129/25
exit
!
interface eth2
  ip address 64.100.1.1/24
exit
!
end
msk-svivanov-gw-01# show interface brief
Interface      Status   VRF          Addresses
-----        -----   ---          -----
eth0          up       default      172.16.20.1/25
eth1          up       default      172.16.20.129/25
eth2          up       default      64.100.1.1/24
eth3          down     default
eth4          down     default
eth5          down     default
eth6          down     default
eth7          down     default
lo            up       default
pimreg        up       default
msk-svivanov-gw-01# 

```

Рис. 2.9: Конфигурация FRR

Проверим подключение с помощью команд ping и trace. Узлы PC1 и PC2 должны успешно отправлять эхо-запросы друг другу и на сервер с двойным стеком (Dual Stack Server). (рис. 10, 11)

```

PC1-svivanov> ping 172.16.20.138
84 bytes from 172.16.20.138 icmp_seq=1 ttl=63 time=4.284 ms
84 bytes from 172.16.20.138 icmp_seq=2 ttl=63 time=1.237 ms
84 bytes from 172.16.20.138 icmp_seq=3 ttl=63 time=3.476 ms
84 bytes from 172.16.20.138 icmp_seq=4 ttl=63 time=1.845 ms
84 bytes from 172.16.20.138 icmp_seq=5 ttl=63 time=3.439 ms

PC1-svivanov> ping 64.100.1.10
84 bytes from 64.100.1.10 icmp_seq=1 ttl=63 time=2.593 ms
84 bytes from 64.100.1.10 icmp_seq=2 ttl=63 time=3.082 ms
84 bytes from 64.100.1.10 icmp_seq=3 ttl=63 time=2.990 ms
84 bytes from 64.100.1.10 icmp_seq=4 ttl=63 time=2.161 ms
84 bytes from 64.100.1.10 icmp_seq=5 ttl=63 time=2.045 ms

PC1-svivanov> trace 172.16.20.138
trace to 172.16.20.138, 8 hops max, press Ctrl+C to stop
1 172.16.20.1    12.531 ms  0.339 ms  0.412 ms
2 *172.16.20.138   3.146 ms (ICMP type:3, code:3, Destination port unreachable)

PC1-svivanov> trace 64.100.1.10
trace to 64.100.1.10, 8 hops max, press Ctrl+C to stop
1 172.16.20.1    1.957 ms  1.203 ms  2.145 ms
2 *64.100.1.10    4.012 ms (ICMP type:3, code:3, Destination port unreachable)

PC1-svivanov>

```

Рис. 2.10: Проверка подключения с PC1

```

PC2-svivanov> ping 172.16.20.10
84 bytes from 172.16.20.10 icmp_seq=1 ttl=63 time=28.206 ms
84 bytes from 172.16.20.10 icmp_seq=2 ttl=63 time=3.354 ms
84 bytes from 172.16.20.10 icmp_seq=3 ttl=63 time=4.043 ms
84 bytes from 172.16.20.10 icmp_seq=4 ttl=63 time=2.513 ms
84 bytes from 172.16.20.10 icmp_seq=5 ttl=63 time=3.244 ms

PC2-svivanov> ping 64.100.1.10
84 bytes from 64.100.1.10 icmp_seq=1 ttl=63 time=7.170 ms
84 bytes from 64.100.1.10 icmp_seq=2 ttl=63 time=1.699 ms
84 bytes from 64.100.1.10 icmp_seq=3 ttl=63 time=1.412 ms
84 bytes from 64.100.1.10 icmp_seq=4 ttl=63 time=1.527 ms
84 bytes from 64.100.1.10 icmp_seq=5 ttl=63 time=2.825 ms

PC2-svivanov> trace 172.16.20.10
trace to 172.16.20.10, 8 hops max, press Ctrl+C to stop
1 172.16.20.129   4.014 ms  0.598 ms  0.593 ms
2 *172.16.20.10    0.846 ms (ICMP type:3, code:3, Destination port unreachable)

PC2-svivanov> trace 64.100.1.10
trace to 64.100.1.10, 8 hops max, press Ctrl+C to stop
1 172.16.20.129   1.541 ms  0.932 ms  0.651 ms
2 *64.100.1.10    2.623 ms (ICMP type:3, code:3, Destination port unreachable)

PC2-svivanov>

```

Рис. 2.11: Проверка подключения с PC2

Настроим IPv6-адресацию для интерфейсов узлов PC3, PC4, Server:
 PC3: (рис. 12)

```

ip 2001:db8:c0de:12::a/64
save

```

```
PC3-svivanov> ip 2001:db8:c0de:12::a/64
PC1 : 2001:db8:c0de:12::a/64

PC3-svivanov> save
Saving startup configuration to startup.vpc
. done

PC3-svivanov> █
```

Рис. 2.12: Настройка PC3

PC4: (рис. 13)

```
ip 2001:db8:c0de:13::a/64
```

```
save
```

```
PC4-svivanov> ip 2001:db8:c0de:13::a/64
PC1 : 2001:db8:c0de:13::a/64

PC4-svivanov> save
Saving startup configuration to startup.vpc
. done

PC4-svivanov> █
```

Рис. 2.13: Настройка PC4

Server: (рис. 14)

```
ip 2001:db8:c0de:11::a/64
```

```
save
```



```
VPCS>
VPCS>
VPCS> ip 2001:db8:c0de:11::a/64
PC1 : 2001:db8:c0de:11::a/64

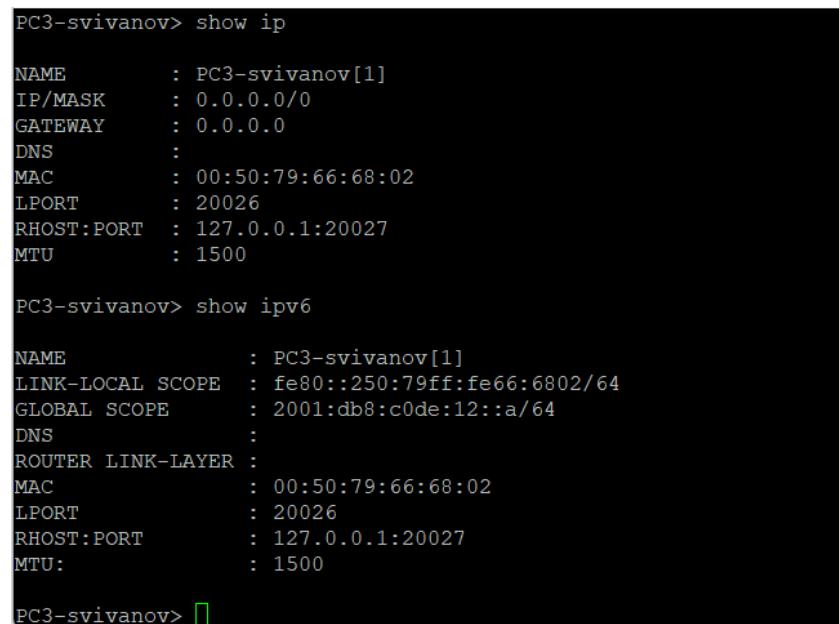
VPCS> save
Saving startup configuration to startup.vpc
. done

VPCS> █
```

Рис. 2.14: Настройка Server

Посмотрим на PC3 и PC4 конфигурацию IPv4 и IPv6: (рис. 15, 16)

```
show ip
show ipv6
```



```
PC3-svivanov> show ip

NAME      : PC3-svivanov[1]
IP/MASK   : 0.0.0.0/0
GATEWAY   : 0.0.0.0
DNS       :
MAC       : 00:50:79:66:68:02
LPORT     : 20026
RHOST:PORT: 127.0.0.1:20027
MTU       : 1500

PC3-svivanov> show ipv6

NAME      : PC3-svivanov[1]
LINK-LOCAL SCOPE : fe80::250:79ff:fe66:6802/64
GLOBAL SCOPE   : 2001:db8:c0de:12::a/64
DNS       :
ROUTER LINK-LAYER :
MAC       : 00:50:79:66:68:02
LPORT     : 20026
RHOST:PORT: 127.0.0.1:20027
MTU       : 1500

PC3-svivanov> █
```

Рис. 2.15: Просмотр конфигураций

```
PC4-svivanov> show ip

NAME      : PC4-svivanov[1]
IP/MASK   : 0.0.0.0/0
GATEWAY   : 0.0.0.0
DNS       :
MAC       : 00:50:79:66:68:03
LPORT     : 20028
RHOST:PORT: 127.0.0.1:20029
MTU       : 1500

PC4-svivanov> show ipv6

NAME      : PC4-svivanov[1]
LINK-LOCAL SCOPE : fe80::250:79ff:fe66:6803/64
GLOBAL SCOPE   : 2001:db8:c0de:13::a/64
DNS       :
ROUTER LINK-LAYER :
MAC       : 00:50:79:66:68:03
LPORT     : 20028
RHOST:PORT: 127.0.0.1:20029
MTU:      : 1500

PC4-svivanov> █
```

Рис. 2.16: Просмотр конфигураций

Настроим IPv6-адресацию для интерфейсов локальной сети маршрутизатора VyOS msk-user-gw-02:

Установим систему на маршрутизатор VyOS: (рис. 17)

```
vyos@vyos:~$ install image
```

```
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:~$ █
```

Рис. 2.17: Установка системы на VyOS

Перейдем в режим конфигурирования, изменим имя устройства: (рис. 18)

```
vyos@vyos$ configure
vyos@vyos# set system host-name msk-user-gw-02
vyos@vyos# compare
vyos@vyos# commit
```

```
vyos@vyos# save  
vyos@vyos# exit  
vyos@vyos$ reboot
```

```
vyos@vyos:~$ configure  
[edit]  
vyos@vyos# set system host-name msk-svivanov-gw-02  
[edit]  
vyos@vyos# compare  
[edit system]  
>host-name msk-svivanov-gw-02  
[edit]  
vyos@vyos# commit  
save'[edit]  
vyos@vyos# save  
Saving configuration to '/config/config.boot'...  
Done  
[edit]  
vyos@vyos# exit  
exit  
vyos@vyos:~$ reboot
```

Рис. 2.18: Конфигурация VyOS

Назначим IPv6-адреса маршрутизатору msk-user-gw-02: (рис. 19)

```
vyos@msk-user-gw-02:~$ configure  
vyos@msk-user-gw-02# set interfaces ethernet eth0 address 2001:db8:c0de:12::1/64  
vyos@msk-user-gw-02# set service router-advert interface eth0 prefix 2001:db8:c0de:  
vyos@msk-user-gw-02# set interfaces ethernet eth1 address 2001:db8:c0de:13::1/64  
vyos@msk-user-gw-02# set service router-advert interface eth1 prefix 2001:db8:c0de:  
vyos@msk-user-gw-02# set interfaces ethernet eth2 address 2001:db8:c0de:11::1/64  
vyos@msk-user-gw-02# set service router-advert interface eth2 prefix 2001:db8:c0de:  
vyos@msk-user-gw-02# compare  
vyos@msk-user-gw-02# commit  
vyos@msk-user-gw-02# save  
vyos@msk-user-gw-02# show interfaces
```

```

+      prefix 2001:db8:c0de:11::/64 {
+      }
+  }
[edit]
vyos@msk-svivanov-gw-02# commit
[edit]
vyos@msk-svivanov-gw-02# save
Saving configuration to '/config/config.boot'...
Done
[edit]
vyos@msk-svivanov-gw-02# show interfaces
  ethernet eth0 {
    address dhcp
    address 2001:db8:c0de:12::1/64
    hw-id 0c:f5:64:b4:00:00
  }
  ethernet eth1 {
    address 2001:db8:c0de:13::1/64
    hw-id 0c:f5:64:b4:00:01
  }
  ethernet eth2 {
    address 2001:db8:c0de:11::1/64
    hw-id 0c:f5:64:b4:00:02

```

Рис. 2.19: Конфигурация VyOS

Проверим подключение с помощью команд ping и trace. Узлы PC3 и PC4 должны успешно отправлять эхо-запросы друг другу и на сервер с двойным стеком (Dual Stack Server). (рис. 20, 21)

```

PC3-svivanov> ping 2001:db8:c0de:13::a
2001:db8:c0de:13::a icmp6_seq=1 ttl=62 time=7.775 ms
2001:db8:c0de:13::a icmp6_seq=2 ttl=62 time=1.353 ms
2001:db8:c0de:13::a icmp6_seq=3 ttl=62 time=2.612 ms
2001:db8:c0de:13::a icmp6_seq=4 ttl=62 time=2.207 ms
2001:db8:c0de:13::a icmp6_seq=5 ttl=62 time=2.727 ms

PC3-svivanov> trace 2001:db8:c0de:13::a
trace to 2001:db8:c0de:13::a, 64 hops max
 1 2001:db8:c0de:12::1    4.372 ms  1.608 ms  2.131 ms
 2 2001:db8:c0de:13::a    2.687 ms  1.888 ms  1.928 ms

PC3-svivanov> ping 2001:db8:c0de:11::a
2001:db8:c0de:11::a icmp6_seq=1 ttl=62 time=2.336 ms
2001:db8:c0de:11::a icmp6_seq=2 ttl=62 time=2.279 ms
2001:db8:c0de:11::a icmp6_seq=3 ttl=62 time=2.712 ms
2001:db8:c0de:11::a icmp6_seq=4 ttl=62 time=2.373 ms
2001:db8:c0de:11::a icmp6_seq=5 ttl=62 time=2.832 ms

PC3-svivanov> trace 2001:db8:c0de:11::a
trace to 2001:db8:c0de:11::a, 64 hops max
 1 2001:db8:c0de:12::1    0.489 ms  0.867 ms  1.843 ms
 2 2001:db8:c0de:11::a    2.773 ms  1.474 ms  0.601 ms

PC3-svivanov> □

```

Рис. 2.20: Проверка подключения с PC3

```
PC4-svivanov> ping 2001:db8:c0de:12::a
2001:db8:c0de:12::a icmp6_seq=1 ttl=62 time=5.765 ms
2001:db8:c0de:12::a icmp6_seq=2 ttl=62 time=1.517 ms
2001:db8:c0de:12::a icmp6_seq=3 ttl=62 time=0.971 ms
2001:db8:c0de:12::a icmp6_seq=4 ttl=62 time=1.127 ms
2001:db8:c0de:12::a icmp6_seq=5 ttl=62 time=2.603 ms

PC4-svivanov> trace 2001:db8:c0de:12::a
trace to 2001:db8:c0de:12::a, 64 hops max
 1 2001:db8:c0de:13::1    3.064 ms   1.278 ms   0.912 ms
 2 2001:db8:c0de:12::a    3.500 ms   1.840 ms   1.758 ms

PC4-svivanov> ping 2001:db8:c0de:11::a
2001:db8:c0de:11::a icmp6_seq=1 ttl=62 time=2.022 ms
2001:db8:c0de:11::a icmp6_seq=2 ttl=62 time=1.965 ms
2001:db8:c0de:11::a icmp6_seq=3 ttl=62 time=1.857 ms
2001:db8:c0de:11::a icmp6_seq=4 ttl=62 time=2.005 ms
2001:db8:c0de:11::a icmp6_seq=5 ttl=62 time=1.890 ms

PC4-svivanov> trace 2001:db8:c0de:11::a
trace to 2001:db8:c0de:11::a, 64 hops max
 1 2001:db8:c0de:13::1    1.203 ms   1.034 ms   0.855 ms
 2 2001:db8:c0de:11::a    3.014 ms   2.039 ms   1.228 ms

PC4-svivanov>
```

Рис. 2.21: Проверка подключения с PC4

Убедимся, что устройства из подсети IPv4 не доступны для устройств из подсети IPv6 и наоборот. Только сервер двойного стека может обращаться к устройствам обеих подсетей. (рис. 22, 23)

```
PC3-svivanov> ping 172.16.20.10
host (172.16.20.10) not reachable

PC3-svivanov>
```

PC1-svivanov - PuTTY

```
PC1-svivanov>
PC1-svivanov> ping 2001:db8:c0de:12::a
host (2001:db8:c0de:12::a) not reachable
```

Рис. 2.22: Проверка недоступности устройств IPv4 и IPv6 между собой

```
VPCS> ping 172.16.20.10
84 bytes from 172.16.20.10 icmp_seq=1 ttl=63 time=3.412 ms
84 bytes from 172.16.20.10 icmp_seq=2 ttl=63 time=3.515 ms
84 bytes from 172.16.20.10 icmp_seq=3 ttl=63 time=1.009 ms
84 bytes from 172.16.20.10 icmp_seq=4 ttl=63 time=2.485 ms
84 bytes from 172.16.20.10 icmp_seq=5 ttl=63 time=1.109 ms

VPCS> ping 2001:db8:c0de:12::a
2001:db8:c0de:12::a icmp6_seq=1 ttl=62 time=5.610 ms
2001:db8:c0de:12::a icmp6_seq=2 ttl=62 time=1.021 ms
2001:db8:c0de:12::a icmp6_seq=3 ttl=62 time=1.096 ms
2001:db8:c0de:12::a icmp6_seq=4 ttl=62 time=0.795 ms
2001:db8:c0de:12::a icmp6_seq=5 ttl=62 time=2.308 ms
```

Рис. 2.23: Обращение сервера к устройствам обеих подсетей

Посмотрим захваченный на соединении сервера двойного стека адресации с коммутатором трафик ARP, ICMP, ICMPv6.

ARP-трафик (IPv4):

Запрос “Who has 64.100.1.10? Tell 64.100.1.1”: маршрутизатор gw-01 ищет MAC-адрес сервера с IP 64.100.1.10

Сервер отвечает своим MAC-адресом. (рис. 24)

19 1882.584300	0c:57:dc:33:00:02	Broadcast	ARP	60 who has 64.100.1.10? Tell 64.100.1.1
20 1882.584520	Private_66:68:04	0c:57:dc:33:00:02	ARP	60 64.100.1.10 is at 00:50:79:66:68:04
21 1882.584974	172.16.20.10	64.100.1.10	ICMP	98 Echo (ping) request id=0x3eee, seq=1/256, ttl=63 (r
22 1882.585173	64.100.1.10	172.16.20.10	ICMP	98 Echo (ping) reply id=0x3eee, seq=1/256, ttl=64 (r
23 1883.588630	172.16.20.10	64.100.1.10	ICMP	98 Echo (ping) request id=0x3fee, seq=2/512, ttl=63 (r
24 1883.588850	64.100.1.10	172.16.20.10	ICMP	98 Echo (ping) reply id=0x3fee, seq=2/512, ttl=64 (r
25 1884.594388	172.16.20.10	64.100.1.10	ICMP	98 Echo (ping) request id=0x40ee, seq=3/768, ttl=63 (r
26 1884.594551	64.100.1.10	172.16.20.10	ICMP	98 Echo (ping) reply id=0x40ee, seq=3/768, ttl=64 (r
27 1885.597585	172.16.20.10	64.100.1.10	ICMP	98 Echo (ping) request id=0x41ee, seq=4/1024, ttl=63 (
28 1885.597900	64.100.1.10	172.16.20.10	ICMP	98 Echo (ping) reply id=0x41ee, seq=4/1024, ttl=64 (
29 1886.600146	172.16.20.10	64.100.1.10	ICMP	98 Echo (ping) request id=0x42ee, seq=5/1280, ttl=63 (
30 1886.600337	64.100.1.10	172.16.20.10	ICMP	98 Echo (ping) reply id=0x42ee, seq=5/1280, ttl=64 (

> Frame 19: Packet, 60 bytes on wire (480 bits), 60 bytes captured (480 bits)	0000 ff ff ff ff ff ff 0c 57 dc 33 00 02 08 06 00 01 ...
> Ethernet II, Src: 0c:57:dc:33:00:02 (0c:57:dc:33:00:02), Dst: Broadcast	0010 08 00 06 04 00 01 0c 57 dc 33 00 02 40 64 01 01 ...
Address Resolution Protocol (request)	0020 00 00 00 00 00 00 40 64 01 00 00 00 00 00 00 00 ...
Hardware type: Ethernet (1)	0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...
Protocol type: IPv4 (0x0800)	
Hardware size: 6	
Protocol size: 4	
Opcode: request (1)	
Sender MAC address: 0c:57:dc:33:00:02 (0c:57:dc:33:00:02)	
Sender IP address: 64.100.1.1	
Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)	
Target IP address: 64.100.1.10	

Рис. 2.24: Трафик ARP

ICMP-трафик:

Устройство 172.16.20.10 (PC1) отправляет эхо-запросы на сервер.

Сервер отвечает эхо-ответами. Связь успешная, всего таких пар запрос-ответ 5, что соответствует тому, что мы в терминале отправили 5 ICMP запросов с PC1 на сервер. (рис. 25)

19 1882.584300 0c:57:dc:33:00:02	Broadcast	ARP	60 Who has 64.100.1.10? Tell 64.100.1.1
20 1882.584520 Private_66:68:04	0c:57:dc:33:00:02	ARP	60 64.100.1.10 is at 00:50:79:66:68:04
→ 21 1882.584974 172.16.20.10	64.100.1.10	ICMP	98 Echo (ping) request id=0x3eee seq=1/256, ttl=63 (r)
→ 22 1882.585173 64.100.1.10	172.16.20.10	ICMP	98 Echo (ping) reply id=0x3eee seq=1/256, ttl=64 (r)
23 1883.588630 172.16.20.10	64.100.1.10	ICMP	98 Echo (ping) request id=0x3fee, seq=2/512, ttl=63 (r)
24 1883.588850 64.100.1.10	172.16.20.10	ICMP	98 Echo (ping) reply id=0x3fee, seq=2/512, ttl=64 (r)
25 1884.594388 172.16.20.10	64.100.1.10	ICMP	98 Echo (ping) request id=0x40ee, seq=3/768, ttl=63 (r)
26 1884.594551 64.100.1.10	172.16.20.10	ICMP	98 Echo (ping) reply id=0x40ee, seq=3/768, ttl=64 (r)
27 1885.597585 172.16.20.10	64.100.1.10	ICMP	98 Echo (ping) request id=0x41ee, seq=4/1024, ttl=63 (r)
28 1885.597900 64.100.1.10	172.16.20.10	ICMP	98 Echo (ping) reply id=0x41ee, seq=4/1024, ttl=64 (r)
29 1886.600146 172.16.20.10	64.100.1.10	ICMP	98 Echo (ping) request id=0x42ee, seq=5/1280, ttl=63 (r)
30 1886.600332 64.100.1.10	172.16.20.10	ICMP	98 Echo (ping) reply id=0x42ee, seq=5/1280, ttl=64 (r)
31 1981.366425 172.16.20.10	64.100.1.10	UDP	106 9889 → 9890 Len=64
32 1981.366652 64.100.1.10	172.16.20.10	ICMP	86 Destination unreachable (Port unreachable)
33 1981.37078 172.16.20.10	64.100.1.10	UDP	106 9889 → 9890 Len=64

> Frame 21: Packet, 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
> Ethernet II, Src: 0c:57:dc:33:00:02 (0c:57:dc:33:00:02), Dst: Private
> Internet Protocol Version 4, Src: 172.16.20.10, Dst: 64.100.1.10
> Internet Control Message Protocol
Type: Echo (ping) request (8)
Code: 0
Checksum: 0xe11c [correct]
[Checksum Status: Good]
Identifier (BE): 16110 (0x3eee)
Identifier (LE): 60990 (0xee3e)
Sequence Number (BE): 1 (0x0001)
Sequence Number (LE): 256 (0x0100)
[Response frame: 22]
> Data (56 bytes)

0000 00 50 79 66 68 04 0c 57 dc 33 00 02 08 00 45 00 -Py
0010 00 54 ee 3e 00 00 3f 01 8b e2 ac 10 14 0a 40 64 .T.
0020 01 0a 08 00 e1 c3 ee 00 01 08 09 0a 0b 0c 0d ...
0030 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d ...
0040 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d ...
0050 2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d ..0
0060 3e 3f >?

Рис. 2.25: Трафик ICMP

ICMPv6-трафик:

Router Solicitation - узел запрашивает маршрутизаторы в сети.

Multicast Listener Report - узлы сообщают о подписке на multicast-группы.

Neighbour Solicitation - это аналог ARP для IPv6, используется для поиска MAC-адреса по IPv6-адресу.

Router Advertisement - маршрутизатор рассыпает информацию о себе и префиксах сети. (рис. 26)

37 1986.367635 0c:57:dc:33:00:02	Private_66:68:04	ARP	60 Who has 64.100.1.10? Tell 64.100.1.1
38 1986.367778 Private_66:68:04	0c:57:dc:33:00:02	ARP	60 64.100.1.10 is at 00:50:79:66:68:04
39 2265.532569 2001:db8:c0de:11::a	ff02::2	ICMPv6	62 Router Solicitation
40 2537.540194 ::	ff02::16	ICMPv6	130 Multicast Listener Report Message v2
41 2538.335532 ::	ff02::16	ICMPv6	130 Multicast Listener Report Message v2
42 2538.465388 ::	ff02::1:ffb4:2	ICMPv6	86 Neighbour Solicitation for fe80::ef5:64ff:feb4:2
43 2539.489514 fe80::ef5:64ff:feb4..	ff02::16	ICMPv6	150 Multicast Listener Report Message v2
44 2539.490431 fe80::ef5:64ff:feb4..	ff02::16	ICMPv6	150 Multicast Listener Report Message v2
45 2539.856636 fe80::ef5:64ff:feb4..	ff02::16	ICMPv6	90 Multicast Listener Report Message v2
46 2540.513119 fe80::ef5:64ff:feb4..	ff02::16	ICMPv6	150 Multicast Listener Report Message v2
47 2856.916666 fe80::ef5:64ff:feb4..	ff02::16	ICMPv6	170 Multicast Listener Report Message v2
48 2857.121500 ::	ff02::1:ff00:1	ICMPv6	86 Neighbor Solicitation for 2001:db8:c0de:11::1
49 2857.267298 fe80::ef5:64ff:feb4..	ff02::1	ICMPv6	118 Router Advertisement from 0c:f5:64:b4:00:02
50 2857.568626 fe80::ef5:64ff:feb4..	ff02::16	ICMPv6	170 Multicast Listener Report Message v2
51 2873.388077 fe80::ef5:64ff:feb4..	ff02::1	ICMPv6	118 Router Advertisement from 0c:f5:64:b4:00:02

> Frame 40: Packet, 130 bytes on wire (1040 bits), 130 bytes captured (1040 bits)
> Ethernet II, Src: 0c:f5:64:b4:00:02 (0c:f5:64:b4:00:02), Dst: IPv6mc
> Internet Protocol Version 6, Src: ::, Dst: ff02::16
> Internet Control Message Protocol v6
Type: Multicast Listener Report Message v2 (143)
Code: 0
Checksum: 0x689d [correct]
[Checksum Status: Good]
Reserved: 0000
Number of Multicast Address Records: 3
> Multicast Address Record Changed to exclude: ff02::1:ffb4:2
> Multicast Address Record Changed to exclude: ff05::2
> Multicast Address Record Changed to exclude: ff02::1

0000 33 33 00 00 00 16 0c f5 64 b4 00 02 86 dd 00 33...
0010 00 00 00 4c 00 01 00 00 00 00 00 00 00 00 00 00 ...
0020 00 00 00 00 00 ff 02 00 00 00 00 00 00 00 00 00 ...
0030 00 00 00 00 00 16 3a 00 05 02 00 00 01 00 8f 00 ...
0040 68 9d 00 00 03 04 00 00 ff 02 00 00 00 00 00 00 h...
0050 00 00 00 00 00 01 ff b4 00 02 04 00 00 00 ff 05 ...
0060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 02 04 ...
0070 00 00 ff 02 00 00 00 00 00 00 00 00 00 00 00 00 00 ...
0080 00 02 ...

Рис. 2.26: Трафик ICMPv6

2.4 Задание для самостоятельного выполнения

Задана топология сети. Предполагается, что маршрутизатор разбивает сеть на две подсети с адресами IPv4 и IPv6:

- подсеть 1: 10.10.1.96/27; 2001:DB8:1:1::/64;
- подсеть 2: 10.10.1.16/28; 2001:DB8:1:4::/64.

Для начала охарактеризуем подсети, укажем, какие адреса в них входят. Предложим вариант таблицы адресации для заданной топологии и адресного пространства, причём для интерфейсов маршрутизатора выбрать наименьший адрес в подсети. (рис. 27)

IPv4 подсеть		
	1 подсеть	2 подсеть
Адрес сети	10.10.1.96	10.10.1.16
Маска	255.255.255.224	255.255.255.240
Количество адресов	32 ($2^{(32 - 27)}$)	16 ($2^{(32 - 28)}$)
Диапазон адресов	10.10.1.96 - 10.10.1.128	10.10.1.16 - 10.10.1.32
Broadcast - адрес	10.10.1.127	10.10.1.31
Доступные адреса	10.10.1.97 - 10.10.1.126	10.10.1.17 - 10.10.1.30

IPv6 подсеть		
	1 подсеть	2 подсеть
Адрес сети	2001:DB8:1:1::/64	2001:DB8:1:4::/64
Предфикс	/64	/64
Доступные адреса	2001:DB8:1:1:0:0:0:0 - 2001:DB8:1:1:ffff:ffff:ffff:ffff -	2001:DB8:1:4:0:0:0:0 - 2001:DB8:1:4:ffff:ffff:ffff:ffff -

Устройство	Интерфейс	IPv4 - адрес	IPv6 - адрес	Шлюз по умолчанию
PC1	NIC	10.10.1.98/27	2001:DB8:1:1::2/64	10.10.1.97/27 2001:DB8:1:1::/64
PC2	NIC	10.10.1.18/28	2001:DB8:1:4::2/64	10.10.1.17/28 2001:DB8:1:4::/64
gw-01	eth0	10.10.1.97/27	2001:DB8:1:1::1/64	-
gw-01	eth1	10.10.1.17/28	2001:DB8:1:4::1/64	-

Рис. 2.27: Характеристика и вариант таблицы адресации

Построим нашу сеть в новом проекте GNS3. (рис. 28)



Рис. 2.28: Построение сети

Настроим маршрутизацию на PC1 в соответствии с составленной таблицей.
(рис. 29)

```
PC1-svivanov> ip 10.10.1.98/27 10.10.1.97
Checking for duplicate address...
PC1-svivanov : 10.10.1.98 255.255.255.224 gateway 10.10.1.97

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

PC1-svivanov> ip 2001:db8:1:1::2/64 2001:db8:1:1::1
PC1 : 2001:db8:1:1::2/64

PC1-svivanov> save
Saving startup configuration to startup.vpc
. done
```

Рис. 2.29: Маршрутизация на PC1

Настроим маршрутизацию на PC2 в соответствии с составленной таблицей.
(рис. 30)

```
PC2-svivanov> ip 10.10.1.18/28 10.10.1.17
Checking for duplicate address...
PC2-svivanov : 10.10.1.18 255.255.255.240 gateway 10.10.1.17

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

PC2-svivanov> ip 2001:db8:1:4::2/64
PC1 : 2001:db8:1:4::2/64

PC2-svivanov> save
Saving startup configuration to startup.vpc
. done
```

Рис. 2.30: Маршрутизация на PC2

Настроим маршрутизацию на VyOS-роутере. (рис. 31)

```
vyos@vyos# set service router-advert interface eth0 prefix 2001:db8:1:1::/64
[edit]
vyos@vyos# set interfaces ethernet eth1 address 10.10.1.17/28
[edit]
vyos@vyos# set interfaces ethernet eth1 address 2001:db8:1:4::1/64
[edit]
vyos@vyos# set service router-advert interface eth1 prefix 2001:db8:1:4::/64
[edit]
vyos@vyos# compare
[edit interfaces ethernet eth0]
+address 10.10.1.97/27
+address 2001:db8:1:1::1/64
[edit interfaces ethernet eth1]
```

Рис. 2.31: Маршрутизация на VyOS

Проверим настройку маршрутизации на VyOS. (рис. 32)

```
vyos@vyos# show interfaces
ethernet eth0 {
    address 10.10.1.97/27
    address 2001:db8:1:1::1/64
    hw-id 0c:d0:c7:39:00:00
}
ethernet eth1 {
    address 10.10.1.17/28
    address 2001:db8:1:4::1/64
    hw-id 0c:d0:c7:39:00:01
}
```

Рис. 2.32: Настройки VyOS

Проверим подключение между устройствами подсети с помощью команд ping и trace.

Сначала проверим подключение с PC1 к PC2. Видим, что все корректно. (рис. 33)

```
PC1-svivanov> ping 10.10.1.18
84 bytes from 10.10.1.18 icmp_seq=1 ttl=63 time=3.021 ms
84 bytes from 10.10.1.18 icmp_seq=2 ttl=63 time=3.264 ms
84 bytes from 10.10.1.18 icmp_seq=3 ttl=63 time=1.726 ms
84 bytes from 10.10.1.18 icmp_seq=4 ttl=63 time=1.097 ms
84 bytes from 10.10.1.18 icmp_seq=5 ttl=63 time=1.231 ms

PC1-svivanov> trace 10.10.1.18
trace to 10.10.1.18, 8 hops max, press Ctrl+C to stop
 1  10.10.1.97    3.714 ms  0.514 ms  0.755 ms
 2  *10.10.1.18   1.653 ms (ICMP type:3, code:3, Destination port unreachable)

PC1-svivanov> ping 2001:db8:1:4::2
2001:db8:1:4::2 icmp6_seq=1 ttl=62 time=2.923 ms
2001:db8:1:4::2 icmp6_seq=2 ttl=62 time=0.880 ms
2001:db8:1:4::2 icmp6_seq=3 ttl=62 time=3.391 ms
2001:db8:1:4::2 icmp6_seq=4 ttl=62 time=3.213 ms
2001:db8:1:4::2 icmp6_seq=5 ttl=62 time=2.455 ms

PC1-svivanov> trace 2001:db8:1:4::2
trace to 2001:db8:1:4::2, 64 hops max
 1 2001:db8:1:1::1    2.703 ms  1.413 ms  0.480 ms
 2 2001:db8:1:4::2   1.220 ms  1.639 ms  1.489 ms

PC1-svivanov>
```

Рис. 2.33: Проверка подключения с PC1

Проверим подключение с PC2 к PC1. Видим, что все корректно. (рис. 34)

```
PC2-svivanov> ping 10.10.1.98
84 bytes from 10.10.1.98 icmp_seq=1 ttl=63 time=3.938 ms
84 bytes from 10.10.1.98 icmp_seq=2 ttl=63 time=2.648 ms
84 bytes from 10.10.1.98 icmp_seq=3 ttl=63 time=3.131 ms
84 bytes from 10.10.1.98 icmp_seq=4 ttl=63 time=1.134 ms
84 bytes from 10.10.1.98 icmp_seq=5 ttl=63 time=1.878 ms

PC2-svivanov> trace 10.10.1.98
trace to 10.10.1.98, 8 hops max, press Ctrl+C to stop
 1  10.10.1.17    1.309 ms  0.425 ms  0.371 ms
 2  *10.10.1.98   1.118 ms (ICMP type:3, code:3, Destination port unreachable)

PC2-svivanov> ping 2001:db8:1:1::2

2001:db8:1:1::2 icmp6_seq=1 ttl=62 time=2.448 ms
2001:db8:1:1::2 icmp6_seq=2 ttl=62 time=3.053 ms
2001:db8:1:1::2 icmp6_seq=3 ttl=62 time=2.542 ms
2001:db8:1:1::2 icmp6_seq=4 ttl=62 time=2.703 ms
2001:db8:1:1::2 icmp6_seq=5 ttl=62 time=2.536 ms

PC2-svivanov> trace 2001:db8:1:1::2
trace to 2001:db8:1:1::2, 64 hops max
 1 2001:db8:1:4::1   3.092 ms  0.907 ms  1.275 ms
 2 2001:db8:1:1::2   2.269 ms  3.593 ms  4.988 ms

PC2-svivanov> █
```

Рис. 2.34: Проверка подключения с PC2

3 Выводы

В ходе выполнения лабораторной работы мы изучили принципы распределения и настройки адресного пространства на устройствах сети.