

Лабораторная работа № 6

Адресация IPv4 и IPv6. Двойной стек

Шубина София Антоновна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Задания	7
3.1	6.3.1.1. Разбиение IPv4-сети на подсети	7
3.2	6.3.1.2. Разбиение IPv6-сети на подсети	7
4	Решение:	9
4.1	Разбиение IPv4-сети на подсети	9
4.1.1	Задача 1: Сеть 172.16.20.0/24	9
4.1.2	Задача 2: Сеть 10.10.1.64/26	10
4.1.3	Задача 3: Сеть 10.10.1.0/26	10
4.2	Разбиение IPv6-сети на подсети	10
4.2.1	Задача 1: Сеть 2001:db8:c0de::/48	10
4.2.2	Задача 2: Сеть 2a02:6b8::/64	11
5	Настройка двойного стека адресации IPv4 и IPv6 в локальной сети	12
5.1	Постановка задачи	12
6	Порядок выполнения работы	13
7	Характеристика подсетей	35
7.1	Настроим IP-адресацию на маршрутизаторе VyOS и оконечных устройствах, причём на интерфейсах маршрутизатора установить наименьший адрес в подсети.	37
7.2	Проверка корректности реализации	40
8	Вывод	45

Список иллюстраций

6.1	новый проект	13
6.2	захват трафика	14
6.3	PC1	14
6.4	PC2	14
6.5	server	15
6.6	PC1	15
6.7	PC2	16
6.8	FRR	16
6.9	PC2	17
6.10	PC1	18
6.11	PC2	18
6.12	PC1	19
6.13	PC2	19
6.14	PC3	19
6.15	PC4	20
6.16	server	20
6.17	PC4	21
6.18	изменение имени устройства	22
6.19	назначение IPv6	23
6.20	compare, commit, save	24
6.21	просмотр интерфейсов	24
6.22	PC3	25
6.23	PC4	26
6.24	недоступность сетей	26
6.25	недоступность сетей	27
6.26	недоступность сетей	27
6.27	недоступность сетей	28
6.28	сервер	28
6.29	ARP	29
6.30	ARP	29
6.31	ARP	29
6.32	ARP	30
6.33	ICMP	30
6.34	ICMP	31
6.35	ICMP	31
6.36	ICMP	31
6.37	ICMPv6	32

6.38 ICMPv6	33
6.39 ICMPv6	33
6.40 ICMPv6	33
7.1 создание сети	35
7.2 Vyos	37
7.3 просмотр интерфейсов	38
7.4 PC1	38
7.5 PC2	39
7.6 проверка PC1	39
7.7 проверка PC2	40
7.8 PC1	41
7.9 PC2	42
7.10 маршрутизатор	43
7.11 PC1	43
7.12 PC2	44

1 Цель работы

Изучение принципов распределения и настройки адресного пространства на устройствах сети.

2 Выполнение лабораторной работы

3 Задания

3.1 6.3.1.1. Разбиение IPv4-сети на подсети

1. Задана IPv4-сеть 172.16.20.0/24. Для заданной сети определите префикс, маску, broadcast-адрес, число возможных подсетей, диапазона адресов узлов. Разбейте сеть на 3 подсети с максимально возможным числом адресов узлов: 126, 62, 62 соответственно.
2. Задана сеть 10.10.1.64/26. Для заданной сети определите префикс, маску, broadcast-адрес, число возможных подсетей, диапазона адресов узлов. Выделите в этой сети подсеть на 30 узлов. Запишите характеристики для выделенной подсети.
3. Задана сеть 10.10.1.0/26. Для этой сети определите префикс, маску, broadcast-адрес, число возможных подсетей, диапазона адресов узлов. Выделите в этой сети подсеть на 14 узлов. Запишите характеристики для выделенной подсети.

3.2 6.3.1.2. Разбиение IPv6-сети на подсети

1. Задана сеть 2001:db8:c0de::/48. Охарактеризуйте адрес, определите маску, префикс, диапазона адресов для узлов сети (краевые значения). Разбейте сеть на 2 подсети двумя способами — с использованием идентификатора подсети и с использованием идентификатора интерфейса. Поясните

предложенные вами варианты разбиения.

2. Задана сеть 2a02:6b8::/64. Охарактеризуйте адрес, определите маску, префикс, диапазона адресов для узлов сети (краевые значения). Разбейте сеть на 2 подсети двумя способами — с использованием идентификатора подсети и с использованием идентификатора интерфейса. Поясните предложенные вами варианты разбиения.

4 Решение:

4.1 Разбиение IPv4-сети на подсети

4.1.1 Задача 1: Сеть 172.16.20.0/24

Для сети 172.16.20.0/24 определены следующие характеристики: префикс /24 соответствует маске подсети 255.255.255.0. Broadcast-адрес рассчитывается путем инвертирования битов сетевой части маски: исходная маска 11111111.11111111.11111111.00000000 (255.255.255.0) после инверсии превращается в 00000000.00000000.00000000.11111111 (0.0.0.255), и применение операции OR к адресу сети 172.16.20.0 дает broadcast-адрес 172.16.20.255. В данной сети возможно 256 подсетей, а диапазон адресов узлов составляет 172.16.20.1-172.16.20.254.

Разбиение на три подсети с требуемым количеством узлов (126, 62, 62) выполняется следующим образом:

Для первой подсети на 126 узлов требуется 128 адресов (126 + 2 служебных). Префикс маски /25 ($32-7=25$) соответствует маске 255.255.255.128. Адрес подсети 172.16.20.0. Broadcast-адрес рассчитывается через инвертированную маску 0.0.0.127: $172.16.20.0 \text{ OR } 0.0.0.127 = 172.16.20.127$. Диапазон адресов узлов: 172.16.20.1-172.16.20.126.

Для второй подсети на 62 узла требуется 64 адреса. Префикс /26 ($32-6=26$) соответствует маске 255.255.255.192. Адрес подсети 172.16.20.128. Broadcast: $172.16.20.128 \text{ OR } 0.0.0.63 = 172.16.20.191$. Диапазон узлов: 172.16.20.129-

172.16.20.190.

Для третьей подсети на 62 узла также требуется 64 адреса с префиксом /26 и маской 255.255.255.192. Адрес подсети 172.16.20.192. Broadcast: 172.16.20.192 OR 0.0.0.63 = 172.16.20.255. Диапазон узлов: 172.16.20.193-172.16.20.254.

4.1.2 Задача 2: Сеть 10.10.1.64/26

Для сети 10.10.1.64/26 определены характеристики: префикс /26 соответствует маске 255.255.255.192. Broadcast-адрес рассчитывается через инвертированную маску 0.0.0.63: 10.10.1.64 OR 0.0.0.63 = 10.10.1.127. В сети возможно 64 подсети, диапазон адресов узлов: 10.10.1.65-10.10.1.126.

Для подсети на 30 узлов требуется 32 адреса. Префикс /27 (32-5=27) соответствует маске 255.255.255.224. Адрес подсети 10.10.1.64. Broadcast: 10.10.1.64 OR 0.0.0.31 = 10.10.1.95. Диапазон узлов: 10.10.1.65-10.10.1.94.

4.1.3 Задача 3: Сеть 10.10.1.0/26

Для сети 10.10.1.0/26 характеристики: префикс /26, маска 255.255.255.192. Broadcast: 10.10.1.0 OR 0.0.0.63 = 10.10.1.63. Возможно 64 подсети, диапазон узлов: 10.10.1.1-10.10.1.62.

Для подсети на 14 узлов требуется 16 адресов. Префикс /28 (32-4=28) соответствует маске 255.255.255.240. Адрес подсети 10.10.1.0. Broadcast: 10.10.1.0 OR 0.0.0.15 = 10.10.1.15. Диапазон узлов: 10.10.1.1-10.10.1.14.

4.2 Разбиение IPv6-сети на подсети

4.2.1 Задача 1: Сеть 2001:db8:c0de::/48

Адрес сети 2001:db8:c0de::/48 имеет маску ffff:ffff:ffff:0000:0000:0000:0000:0000. Префикс составляет первые 48 бит (2001:db8:c0de). Диапазон адресов для узлов се-

ти: от 2001:db8:c0de:0000:0000:0000:0000:0000 до 2001:db8:c0de:ffff:ffff:ffff:ffff:ffff. Первые 48 бит фиксированы, следующие 16 бит идентифицируют подсеть, последние 64 бита содержат идентификатор интерфейса.

Разбиение на 2 подсети с использованием идентификатора подсети: 2001:db8:c0de:0002::/64 и 2001:db8:c0de:0003::/64. С использованием идентификатора интерфейса (на границе полубайта): 2001:db8:c0de:0000:1000::/68 и 2001:db8:c0de:0000:2000::/68.

4.2.2 Задача 2: Сеть 2a02:6b8::/64

Адрес сети 2a02:6b8::/64 имеет маску ffff:ffff:ffff:ffff:0000:0000:0000:0000. Префикс составляет первые 64 бита (2a02:6b8:0:0). Диапазон адресов узлов: от 2a02:6b8:0000:0000:0000:0000:0000:0000 до 2a02:6b8:0000:0000:ffff:ffff:ffff:ffff. Все 64 бита после префикса принадлежат идентификатору интерфейса.

Разбиение с использованием идентификатора подсети невозможно для сети /64, так как отсутствует отдельное поле для идентификатора подсети. Этот способ применим только для сетей с префиксом /48 или короче. С использованием идентификатора интерфейса (на границе полубайта): 2a02:6b8:0000:0000:1000::/68 и 2a02:6b8:0000:0000:2000::/68.

5 Настройка двойного стека адресации IPv4 и IPv6 в локальной сети

5.1 Постановка задачи

Задана топология сети с двумя локальными подсетями. Адресное пространство распределено следующим образом: - для первой подсети выделены адреса IPv4 - для второй подсети выделены адреса IPv6

Требуется выполнить:

1. Реализовать заданную топологию сети в GNS3
2. Настроить IPv4-адресацию на устройствах первой подсети и проверить подключение между устройствами этой подсети
3. Настроить IPv6-адресацию на устройствах второй подсети и проверить подключение между устройствами этой подсети
4. Проанализировать захваченный на соединении сервера двойного стека адресации с коммутатором трафик:
 - ARP
 - ICMP
 - ICMPv6

6 Порядок выполнения работы

1. Я запускаю GNS3 VM и GNS3, затем создаю новый проект.
2. В рабочем пространстве я размещаю и соединяю устройства в соответствии с топологией. Для подсети IPv4 я использую маршрутизатор FRR, а для подсети с IPv6 — маршрутизатор VyOS.
3. Я изменяю отображаемые названия устройств по следующим принципам:
 - Коммутаторам присваиваю названия: `msk-sashubinar-sw-0x`
 - Маршрутизаторам присваиваю названия: `msk-sashubina-gw-0x`
 - VPCS присваиваю названия: `PCx-sashubina` Где вместо user я указываю имя своей учётной записи- `sashubina`, вместо `x` — порядковый номер устройства.

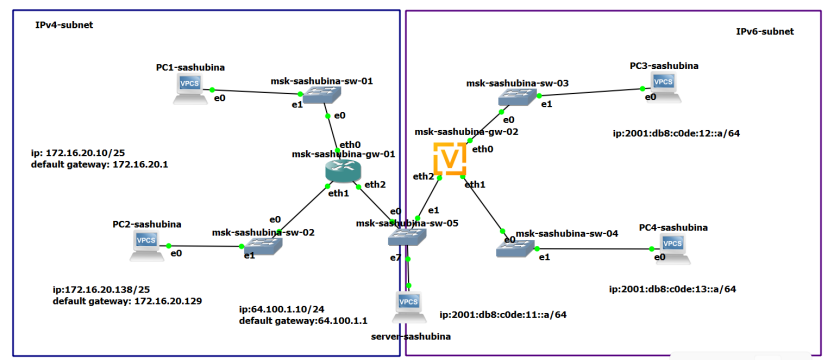


Рис. 6.1: новый проект

4. Я включаю захват трафика на соединении между сервером двойного стека адресации и ближайшим к нему коммутатором.

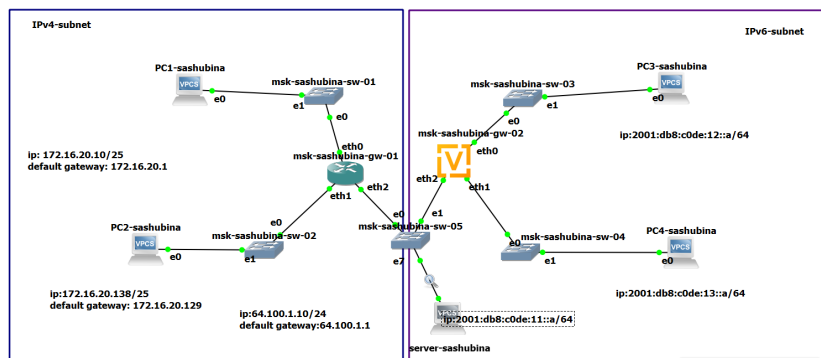


Рис. 6.2: захват трафика

5. Руководствуясь таблицей, я настраиваю IPv4-адресацию для интерфейсов узлов PC1, PC2, Server: PC1:

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

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

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

VPCS> █
```

Рис. 6.3: PC1

PC2: ip 172.16.20.138/25 172.16.20.129 save

```
VPCS> ip 172.16.20.138/25 172.16.20.129
Checking for duplicate address...
PC1 : 172.16.20.138 255.255.255.128 gateway 172.16.20.129

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

VPCS> █
```

Рис. 6.4: PC2

Server: ip 64.100.1.10/24 64.100.1.1 save

```

VPCS> ip 64.100.1.10/24 64.100.1.1
Checking for duplicate address...
PC1 : 64.100.1.10 255.255.255.0 gateway 64.100.1.1

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

VPCS> 

```

Рис. 6.5: server

6. Я просматриваю на PC1 и PC2 конфигурацию IPv4 и IPv6:

show ip

show ipv6

```

VPCS> show ip

NAME           : VPCS[1]
IP/MASK        : 172.16.20.10/25
GATEWAY        : 172.16.20.1
DNS            :
MAC            : 00:50:79:66:68:00
LPORT         : 10016
RHOST:PORT     : 127.0.0.1:10017
MTU            : 1500

VPCS> show ipv6

NAME           : VPCS[1]
LINK-LOCAL SCOPE : fe80::250:79ff:fe66:6800/64
GLOBAL SCOPE    :
ROUTER LINK-LAYER :
MAC            : 00:50:79:66:68:00
LPORT         : 10016
RHOST:PORT     : 127.0.0.1:10017
MTU            : 1500

VPCS> 

```

Рис. 6.6: PC1

```

VPCS> show ip
NAME       : VPCS[1]
IP/MASK    : 172.16.20.138/25
GATEWAY    : 172.16.20.129
DNS        :
MAC        : 00:50:79:66:68:01
LPORT      : 10018
RHOST:PORT : 127.0.0.1:10019
MTU        : 1500

VPCS> show ipv6
NAME       : VPCS[1]
LINK-LOCAL SCOPE : fe80::250:79ff:fe66:6801/64
GLOBAL SCOPE    :
ROUTER LINK-LAYER :
MAC            : 00:50:79:66:68:01
LPORT          : 10018
RHOST:PORT     : 127.0.0.1:10019
MTU            : 1500

VPCS>

```

Рис. 6.7: PC2

7. Руководствуясь таблицей, я настраиваю IPv4-адресацию для интерфейсов локальной сети маршрутизатора FRR msk-sashubina-gw-01.

```

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

```

Рис. 6.8: FRR

8. Я проверяю конфигурацию маршрутизатора и настройки IPv4-адресации:


```
msk-sashubina-gw-01# show running-config
msk-sashubina-gw-01# show interface brief
```

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

Current configuration:
!
frr version 8.2.2
frr defaults traditional
hostname frr
hostname msk-sashubina-gw-01
service integrated-vtysh-config
!
interface eth0
 ip address 172.16.20.1/25
exit
!
interface eth1
 ip address 172.16.20.129/25
exit
!
interface eth2
 ip address 64.100.1.1/24
exit
!
end
msk-sashubina-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
```

Рис. 6.9: PC2

Маршрутизатор msk-sashubina-gw-01 настроен с тремя рабочими интерфейсами. На интерфейсе eth0 установлен адрес 172.16.20.1/25 - это первая половина сети 172.16.20.0/24, покрывающая диапазон 172.16.20.1-126. Интерфейс eth1 имеет адрес 172.16.20.129/25 - вторая половина сети с диапазоном 172.16.20.129-254. Интерфейс eth2 настроен с адресом 64.100.1.1/24 для подключения к внешней сети.

Все интерфейсы используют наименьшие доступные адреса в своих подсетях, что соответствует требованиям. Маршрутизатор работает на базе FRR, интерфейсы eth0, eth1 и eth2 активны, остальные порты отключены. Конфигурация корректна и готова к работе.

9. Я проверяю подключение с помощью команд ping и trase. Убеждаюсь, что узлы PC1 и PC2 успешно отправляют эхо-запросы друг другу и на сервер с двойным стеком (Dual Stack Server), также я проверила отправку на свой шлюз.

```
VPCS> ping 172.16.20.138
84 bytes from 172.16.20.138 icmp_seq=1 ttl=63 time=5.443 ms
84 bytes from 172.16.20.138 icmp_seq=2 ttl=63 time=2.305 ms
84 bytes from 172.16.20.138 icmp_seq=3 ttl=63 time=2.661 ms
84 bytes from 172.16.20.138 icmp_seq=4 ttl=63 time=2.859 ms
84 bytes from 172.16.20.138 icmp_seq=5 ttl=63 time=2.601 ms

VPCS>
VPCS> ping 172.16.20.1
84 bytes from 172.16.20.1 icmp_seq=1 ttl=64 time=4.248 ms
84 bytes from 172.16.20.1 icmp_seq=2 ttl=64 time=1.647 ms
84 bytes from 172.16.20.1 icmp_seq=3 ttl=64 time=1.483 ms
84 bytes from 172.16.20.1 icmp_seq=4 ttl=64 time=1.335 ms
84 bytes from 172.16.20.1 icmp_seq=5 ttl=64 time=1.744 ms

VPCS> ping 64.100.1.10
84 bytes from 64.100.1.10 icmp_seq=1 ttl=63 time=7.732 ms
84 bytes from 64.100.1.10 icmp_seq=2 ttl=63 time=3.555 ms
84 bytes from 64.100.1.10 icmp_seq=3 ttl=63 time=2.717 ms
84 bytes from 64.100.1.10 icmp_seq=4 ttl=63 time=2.563 ms
84 bytes from 64.100.1.10 icmp_seq=5 ttl=63 time=2.672 ms

VPCS> █
```

Рис. 6.10: PC1

```
VPCS> ping 172.16.20.129
84 bytes from 172.16.20.129 icmp_seq=1 ttl=64 time=0.931 ms
84 bytes from 172.16.20.129 icmp_seq=2 ttl=64 time=1.379 ms
84 bytes from 172.16.20.129 icmp_seq=3 ttl=64 time=1.384 ms
84 bytes from 172.16.20.129 icmp_seq=4 ttl=64 time=1.934 ms
84 bytes from 172.16.20.129 icmp_seq=5 ttl=64 time=1.462 ms

VPCS> ping 172.16.20.10
84 bytes from 172.16.20.10 icmp_seq=1 ttl=63 time=3.809 ms
84 bytes from 172.16.20.10 icmp_seq=2 ttl=63 time=2.168 ms
84 bytes from 172.16.20.10 icmp_seq=3 ttl=63 time=2.456 ms
84 bytes from 172.16.20.10 icmp_seq=4 ttl=63 time=2.134 ms
84 bytes from 172.16.20.10 icmp_seq=5 ttl=63 time=3.103 ms

VPCS> ping 64.100.1.10
84 bytes from 64.100.1.10 icmp_seq=1 ttl=63 time=2.899 ms
84 bytes from 64.100.1.10 icmp_seq=2 ttl=63 time=3.800 ms
84 bytes from 64.100.1.10 icmp_seq=3 ttl=63 time=1.921 ms
84 bytes from 64.100.1.10 icmp_seq=4 ttl=63 time=1.893 ms
84 bytes from 64.100.1.10 icmp_seq=5 ttl=63 time=2.679 ms

VPCS> █
```

Рис. 6.11: PC2

```

VPCS> trace 172.16.20.138
trace to 172.16.20.138, 8 hops max, press Ctrl+C to stop
 1 172.16.20.1 3.206 ms 1.009 ms 1.667 ms
 2 *172.16.20.138 2.635 ms (ICMP type:3, code:3, Destination port unreachable)

VPCS> trace 64.100.1.10
trace to 64.100.1.10, 8 hops max, press Ctrl+C to stop
 1 172.16.20.1 1.383 ms 2.922 ms 0.830 ms
 2 * * *
 3 *64.100.1.10 3.195 ms (ICMP type:3, code:3, Destination port unreachable)

VPCS> █

```

Рис. 6.12: PC1

```

VPCS> trace 172.16.20.10
trace to 172.16.20.10, 8 hops max, press Ctrl+C to stop
 1 172.16.20.129 1.458 ms 1.279 ms 1.149 ms
 2 *172.16.20.10 2.491 ms (ICMP type:3, code:3, Destination port unreachable)
)

VPCS> trace 64.100.1.10
trace to 64.100.1.10, 8 hops max, press Ctrl+C to stop
 1 172.16.20.129 1.769 ms 0.879 ms 0.862 ms
 2 *64.100.1.10 2.589 ms (ICMP type:3, code:3, Destination port unreachable)

VPCS> █

```

Рис. 6.13: PC2

Все пакеты успешно доставляются

10. Руководствуясь таблицей, я настраиваю IPv6-адресацию для интерфейсов узлов PC3, PC4, Server:

PC3:

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

save

```

VPCS> ip 2001:db8:c0de:12::a/64
PC1 : 2001:db8:c0de:12::a/64

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

VPCS> █

```

Рис. 6.14: PC3

PC4:

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

save

```

VPCS> ip 2001:db8:c0de:13::a/64
PC1 : 2001:db8:c0de:13::a/64

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

VPCS> 

```

Рис. 6.15: PC4

Server:

```

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

```

```

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. 6.16: server

11. Я просматриваю на PC3 и PC4 конфигурацию IPv4 и IPv6: show ip show

```
VPCS> show ip
NAME          : VPCS[1]
IP/MASK       : 0.0.0.0/0
GATEWAY       : 0.0.0.0
DNS           :
MAC           : 00:50:79:66:68:01
LPORT        : 10022
RHOST:PORT    : 127.0.0.1:10023
MTU           : 1500

VPCS> show ipv6
NAME          : VPCS[1]
LINK-LOCAL SCOPE : fe80::250:79ff:fe66:6801/64
GLOBAL SCOPE    : 2001:db8:c0de:12::a/64
ROUTER LINK-LAYER :
MAC            : 00:50:79:66:68:01
LPORT         : 10022
RHOST:PORT     : 127.0.0.1:10023
MTU           : 1500

ipv6 VPCS> 
```

```
VPCS> show ip
NAME          : VPCS[1]
IP/MASK       : 0.0.0.0/0
GATEWAY       : 0.0.0.0
DNS           :
MAC           : 00:50:79:66:68:03
LPORT        : 10024
RHOST:PORT    : 127.0.0.1:10025
MTU           : 1500

VPCS> show ipv6
NAME          : VPCS[1]
LINK-LOCAL SCOPE : fe80::250:79ff:fe66:6803/
GLOBAL SCOPE    : 2001:db8:c0de:13::a/64
ROUTER LINK-LAYER :
MAC            : 00:50:79:66:68:03
LPORT         : 10024
RHOST:PORT     : 127.0.0.1:10025
MTU           : 1500

VPCS> 
```

Рис. 6.17: PC4

Все корректно настроено.

12. Руководствуясь таблицей, я настраиваю IPv6-адресацию для интерфейсов локальной сети маршрутизатора VyOS msk-sashubina-gw-02:

- Сначала я устанавливаю систему на маршрутизатор VyOS:

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

Но у меня она уже была установлена

- Затем перехожу в режим конфигурирования и изменяю имя устройства:

```
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:~$ install image
You are trying to install from an already installed system. An ISO
image file to install or URL must be specified.
Exiting...
vyos@vyos:~$ configure
[edit]
vyos@vyos# set system host-name msk-sashubina-gw-02
[edit]
vyos@vyos# compare
[edit system]
>host-name msk-sashubina-gw-02
[edit]
vyos@vyos# commit
[edit]
vyos@vyos# save
Saving configuration to '/config/config.boot'...
Done
[edit]
vyos@vyos# exit
exit
vyos@vyos:~$ reboot
Are you sure you want to reboot this system? [y/N] y
[ 594.605501] systemd-shutdown[1]: Failed to parse (null): No such file or directory
[ 594.624909] systemd-shutdown[1]: Failed to deactivate swaps: No such file or directory
[ 594.673589] [2888]: Failed to unmount /usr/lib/live/mount/persistence: Device or resource busy
[ 594.983301] reboot: Restarting system
```

Рис. 6.18: изменение имени устройства

- Назначаю IPv6-адреса маршрутизатору msk-sashubina-gw-02:

```
vyos@msk-sashubina-gw-02:~$ configure
vyos@msk-sashubina-gw-02# set interfaces ethernet eth0 address 2001:db8:
vyos@msk-sashubina-gw-02# set service router-advert interface eth0 prefix
vyos@msk-sashubina-gw-02# set interfaces ethernet eth1 address 2001:db8:
vyos@msk-sashubina-gw-02# set service router-advert interface eth1 prefix
vyos@msk-sashubina-gw-02# set interfaces ethernet eth2 address 2001:db8:
vyos@msk-sashubina-gw-02# set service router-advert interface eth2 prefix
vyos@msk-sashubina-gw-02# compare
vyos@msk-sashubina-gw-02# commit
vyos@msk-sashubina-gw-02# save
vyos@msk-sashubina-gw-02# show interfaces
```

```
vyos@msk-sashubina-gw-02:~$ configure
[edit]
vyos@msk-sashubina-gw-02# set interfaces ethernet eth0 address 2001:db8:c0de:12:
:1/64
[edit]
vyos@msk-sashubina-gw-02# set service router-advert interface eth0 prefix 2001:d
b8:c0de:12::/64
[edit]
vyos@msk-sashubina-gw-02# set interfaces ethernet eth1 address 2001:db8:c0de:13:
:1/64
[edit]
vyos@msk-sashubina-gw-02# set service router-advert interface eth1 prefix 2001:
db8:c0de:13::/64
[edit]
vyos@msk-sashubina-gw-02# set interfaces ethernet eth2 address 2001:db8:c0de:11:
:1/64
[edit]
vyos@msk-sashubina-gw-02# set service router-advert interface eth2 prefix 2001:d
b8:c0de:11::/64
[edit]
```

Рис. 6.19: назначение IPv6

```

vyos@msk-sashubina-gw-02# compare
[edit interfaces ethernet eth0]
+address 2001:db8:c0de:12::1/64
[edit interfaces ethernet eth1]
+address 2001:db8:c0de:13::1/64
[edit interfaces ethernet eth2]
+address 2001:db8:c0de:11::1/64
[edit service]
+router-advert {
+  interface eth0 {
+    prefix 2001:db8:c0de:12::/64 {
+    }
+  }
+  interface eth1 {
+    prefix 2001:db8:c0de:13::/64 {
+    }
+  }
+  interface eth2 {
+    prefix 2001:db8:c0de:11::/64 {
+    }
+  }
+}
[edit]
vyos@msk-sashubina-gw-02# commit
[edit]
vyos@msk-sashubina-gw-02# save
Saving configuration to '/config/config.boot'...
Done
[edit]

```

Рис. 6.20: compare, commit, save

```

vyos@msk-sashubina-gw-02# show interfaces
ethernet eth0 {
  address dhcp
  address 2001:db8:c0de:12::1/64
  hw-id 0c:57:e3:cd:00:00
}
ethernet eth1 {
  address 2001:db8:c0de:13::1/64
  hw-id 0c:57:e3:cd:00:01
}
ethernet eth2 {
  address 2001:db8:c0de:11::1/64
  hw-id 0c:57:e3:cd:00:02
}
ethernet eth3 {
  hw-id 0c:57:e3:cd:00:03
}
ethernet eth4 {
  hw-id 0c:57:e3:cd:00:04
}
ethernet eth5 {
  hw-id 0c:57:e3:cd:00:05
}
ethernet eth6 {
  hw-id 0c:57:e3:cd:00:06
}
ethernet eth7 {
  hw-id 0c:57:e3:cd:00:07
}
ethernet eth8 {
  hw-id 0c:57:e3:cd:00:08
}
ethernet eth9 {
  hw-id 0c:57:e3:cd:00:09
}
loopback lo {
}
[edit]
vyos@msk-sashubina-gw-02#
[edit]
vyos@msk-sashubina-gw-02# █

```

Рис. 6.21: просмотр интерфейсов

Маршрутизатор настроен с тремя активными интерфейсами, работающими только с IPv6. На интерфейсе eth0 включен DHCP-клиент для автоматического получения IPv4-адреса, а также статически назначен IPv6-адрес 2001:db8:c0de:12::1/64. Интерфейсы eth1 и eth2 имеют статические IPv6-адреса 2001:db8:c0de:13::1/64 и 2001:db8:c0de:11::1/64 соответственно.

Все три интерфейса используют первые адреса в своих IPv6-подсетях (::1), что соответствует требованию выбора наименьших адресов. Остальные интерфейсы (eth3-eth9) физически присутствуют, но не настроены и отключены. Конфигурация демонстрирует переход на IPv6-адресацию с сохранением обратной совместимости через DHCP на eth0.

13. Я проверяю подключение с помощью команд ping и trace. Убеждаюсь, что узлы PC3 и PC4 успешно отправляют эхо-запросы друг другу и на сервер с двойным стеком (Dual Stack Server).

```
VPCS> ping 2001:db8:c0de:13::a

2001:db8:c0de:13::a icmp6_seq=1 ttl=62 time=4.810 ms
2001:db8:c0de:13::a icmp6_seq=2 ttl=62 time=0.785 ms
2001:db8:c0de:13::a icmp6_seq=3 ttl=62 time=0.858 ms
2001:db8:c0de:13::a icmp6_seq=4 ttl=62 time=0.920 ms
2001:db8:c0de:13::a icmp6_seq=5 ttl=62 time=0.840 ms

VPCS> ping 2001:db8:c0de:11::a

2001:db8:c0de:11::a icmp6_seq=1 ttl=62 time=2.679 ms
2001:db8:c0de:11::a icmp6_seq=2 ttl=62 time=1.615 ms
2001:db8:c0de:11::a icmp6_seq=3 ttl=62 time=1.660 ms
2001:db8:c0de:11::a icmp6_seq=4 ttl=62 time=1.010 ms
2001:db8:c0de:11::a icmp6_seq=5 ttl=62 time=3.680 ms

VPCS> trace 2001:db8:c0de:13::a

trace to 2001:db8:c0de:13::a, 64 hops max
 1 2001:db8:c0de:12::1 16.877 ms 6.997 ms 2.540 ms
 2 2001:db8:c0de:13::a 1.768 ms 0.956 ms 0.763 ms

VPCS> trace 2001:db8:c0de:11::a

trace to 2001:db8:c0de:11::a, 64 hops max
 1 2001:db8:c0de:12::1 0.466 ms 0.378 ms 0.485 ms
 2 2001:db8:c0de:11::a 3.456 ms 2.929 ms 1.276 ms

VPCS> █
```

Рис. 6.22: PC3

```

VPCS> ping 2001:db8:c0de:13::a

2001:db8:c0de:13::a icmp6_seq=1 ttl=62 time=4.810 ms
2001:db8:c0de:13::a icmp6_seq=2 ttl=62 time=0.785 ms
2001:db8:c0de:13::a icmp6_seq=3 ttl=62 time=0.858 ms
2001:db8:c0de:13::a icmp6_seq=4 ttl=62 time=0.920 ms
2001:db8:c0de:13::a icmp6_seq=5 ttl=62 time=0.840 ms

VPCS> ping 2001:db8:c0de:11::a

2001:db8:c0de:11::a icmp6_seq=1 ttl=62 time=2.679 ms
2001:db8:c0de:11::a icmp6_seq=2 ttl=62 time=1.615 ms
2001:db8:c0de:11::a icmp6_seq=3 ttl=62 time=1.660 ms
2001:db8:c0de:11::a icmp6_seq=4 ttl=62 time=1.010 ms
2001:db8:c0de:11::a icmp6_seq=5 ttl=62 time=3.680 ms

VPCS> trace 2001:db8:c0de:13::a

trace to 2001:db8:c0de:13::a, 64 hops max
 1 2001:db8:c0de:12::1 16.877 ms 6.997 ms 2.540 ms
 2 2001:db8:c0de:13::a 1.768 ms 0.956 ms 0.763 ms

VPCS> trace 2001:db8:c0de:11::a

trace to 2001:db8:c0de:11::a, 64 hops max
 1 2001:db8:c0de:12::1 0.466 ms 0.378 ms 0.485 ms
 2 2001:db8:c0de:11::a 3.456 ms 2.929 ms 1.276 ms

VPCS> 

```

Рис. 6.23: PC4

Подключение успешно. Пакеты отправляются

14. Я проверяю, что устройства из подсети IPv4 недоступны для устройств из подсети IPv6 и наоборот. Убеждаюсь, что только сервер двойного стека может обращаться к устройствам обеих подсетей.

```

VPCS> ping 172.16.20.10

host (172.16.20.10) not reachable

VPCS> ping 172.16.20.138

host (172.16.20.138) not reachable

VPCS> ping 64.100.1.10

host (64.100.1.10) not reachable

VPCS> 

```

Рис. 6.24: недоступность сетей

```
VPCS> ping 172.16.20.10  
host (172.16.20.10) not reachable  
VPCS> ping 172.16.20.138  
host (172.16.20.138) not reachable  
VPCS> ping 64.100.1.10  
host (64.100.1.10) not reachable  
VPCS> 
```

Рис. 6.25: недоступность сетей

```
VPCS> ping 2001:db8:c0de:12::a  
host (2001:db8:c0de:12::a) not reachable  
VPCS> ping 2001:db8:c0de:13::a  
host (2001:db8:c0de:13::a) not reachable  
VPCS> ping 2001:db8:c0de:11::a  
host (2001:db8:c0de:11::a) not reachable  
VPCS> 
```

Рис. 6.26: недоступность сетей

```

VPCS> ping 2001:db8:c0de:12::a

host (2001:db8:c0de:12::a) not reachable

VPCS> ping 2001:db8:c0de:13::a

host (2001:db8:c0de:13::a) not reachable

VPCS> ping 2001:db8:c0de:11::a

host (2001:db8:c0de:11::a) not reachable

VPCS> █

```

Рис. 6.27: недоступность сетей

```

VPCS> ping 172.16.20.10

84 bytes from 172.16.20.10 icmp_seq=1 ttl=63 time=6.834 ms
84 bytes from 172.16.20.10 icmp_seq=2 ttl=63 time=2.149 ms
84 bytes from 172.16.20.10 icmp_seq=3 ttl=63 time=1.450 ms
84 bytes from 172.16.20.10 icmp_seq=4 ttl=63 time=1.055 ms
84 bytes from 172.16.20.10 icmp_seq=5 ttl=63 time=1.347 ms

VPCS> ping 172.16.20.138

84 bytes from 172.16.20.138 icmp_seq=1 ttl=63 time=2.603 ms
84 bytes from 172.16.20.138 icmp_seq=2 ttl=63 time=1.675 ms
84 bytes from 172.16.20.138 icmp_seq=3 ttl=63 time=1.257 ms
84 bytes from 172.16.20.138 icmp_seq=4 ttl=63 time=1.502 ms
84 bytes from 172.16.20.138 icmp_seq=5 ttl=63 time=1.164 ms

VPCS> ping 2001:db8:c0de:12::a

2001:db8:c0de:12::a icmp6_seq=1 ttl=62 time=4.886 ms
2001:db8:c0de:12::a icmp6_seq=2 ttl=62 time=1.050 ms
2001:db8:c0de:12::a icmp6_seq=3 ttl=62 time=0.919 ms
2001:db8:c0de:12::a icmp6_seq=4 ttl=62 time=1.262 ms
2001:db8:c0de:12::a icmp6_seq=5 ttl=62 time=1.146 ms

VPCS> ping 2001:db8:c0de:13::a

2001:db8:c0de:13::a icmp6_seq=1 ttl=62 time=2.648 ms
2001:db8:c0de:13::a icmp6_seq=2 ttl=62 time=1.494 ms
2001:db8:c0de:13::a icmp6_seq=3 ttl=62 time=0.814 ms
2001:db8:c0de:13::a icmp6_seq=4 ttl=62 time=0.977 ms
2001:db8:c0de:13::a icmp6_seq=5 ttl=62 time=1.798 ms

VPCS> █

```

Рис. 6.28: сервер

Сервер подключается успешно.

15. Я анализирую захваченный на соединении сервера двойного стека адресации с коммутатором трафик ARP, ICMP, ICMPv6.

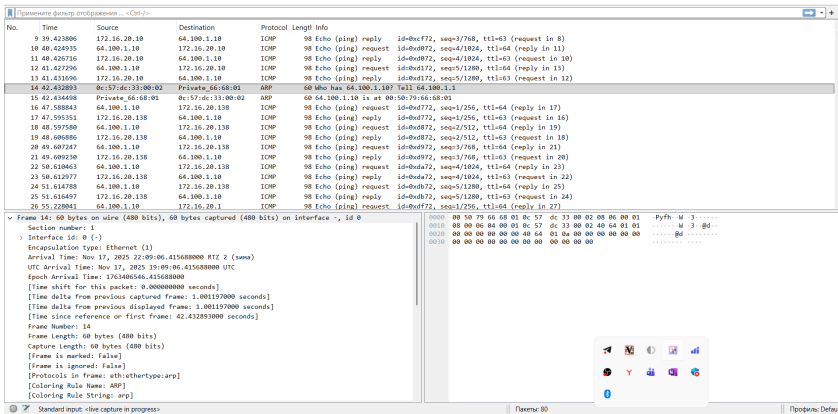


Рис. 6.29: ARP

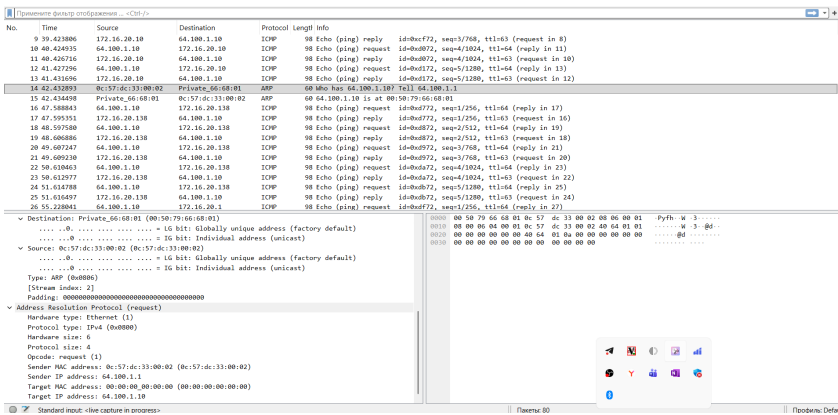


Рис. 6.30: ARP

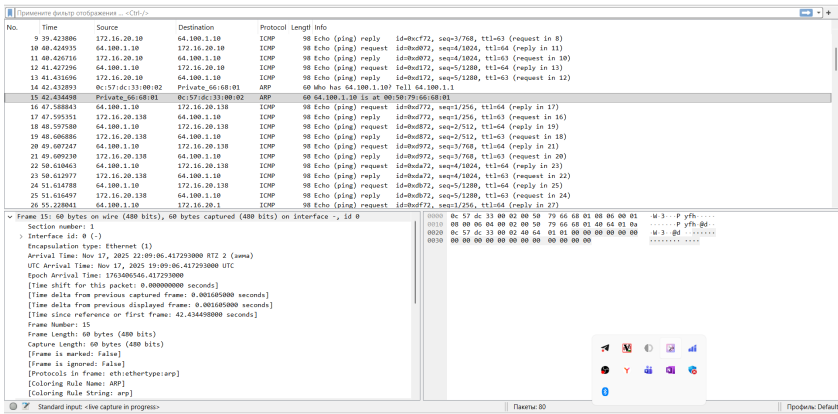


Рис. 6.31: ARP

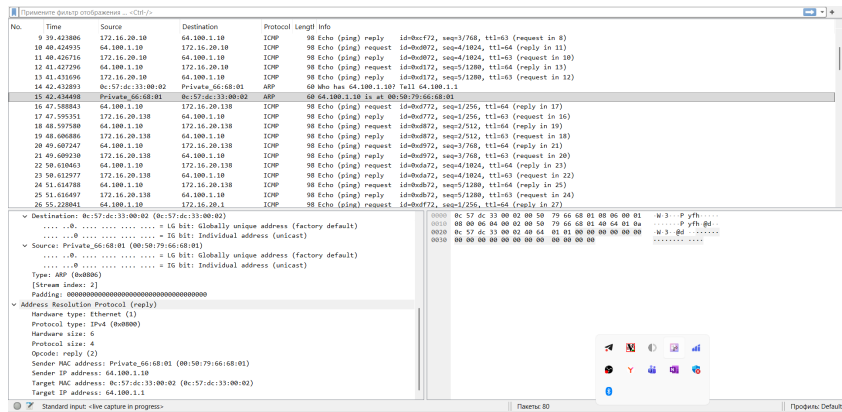


Рис. 6.32: ARP

Протокол ARP (Address Resolution Protocol) обладает следующими техническими характеристиками: длина ARP-кадра составляет 60 байт, протокол не использует порты, работая на канальном уровне (L2), а идентификатор протокола в Ethernet-фрейме — 0x0806. В ARP-запросе source (источник) включает MAC-адрес — физический адрес устройства-отправителя и IP-адрес — логический адрес отправителя в сети, в то время как destination (назначение) содержит MAC-адрес — физический адрес целевого устройства и IP-адрес — логический адрес получателя в сети. Анализ сетевого трафика демонстрирует, что механизм запрос-ответ протокола ARP происходит успешно: устройства корректно обмениваются ARP-запросами и ответами, что подтверждает правильное функционирование протокола и возможность установления сетевой связи между узлами.

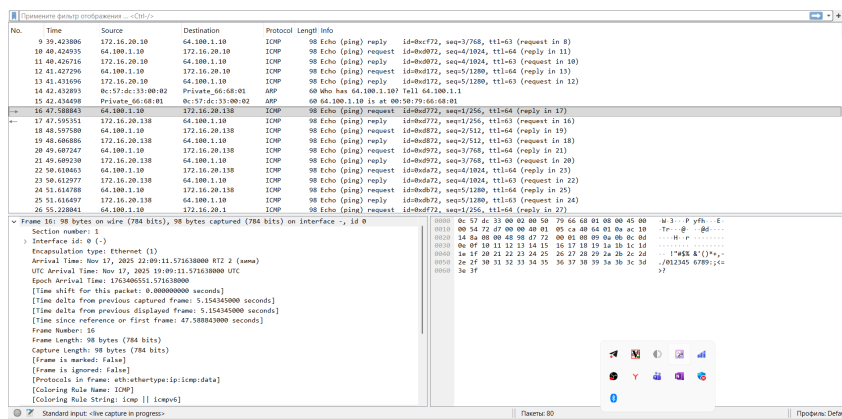


Рис. 6.33: ICMP

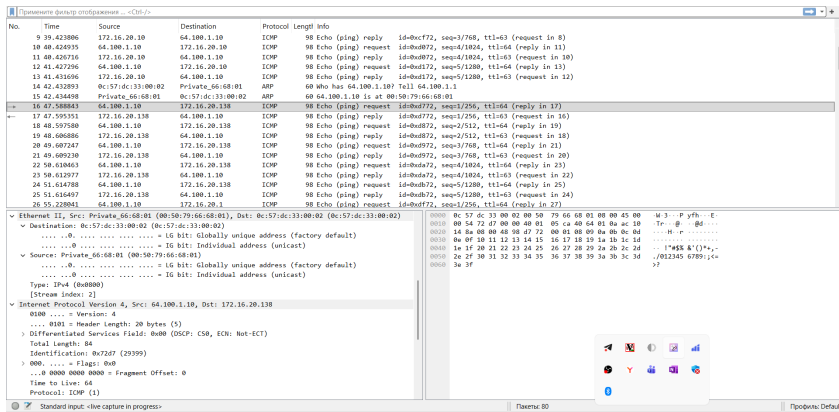


Рис. 6.34: ICMP

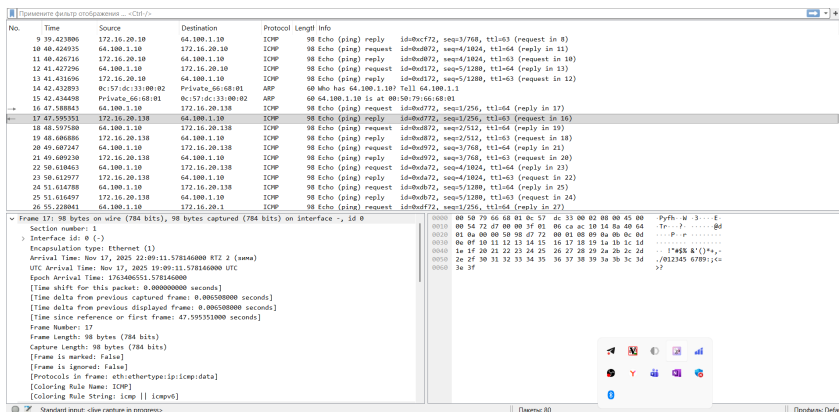


Рис. 6.35: ICMP

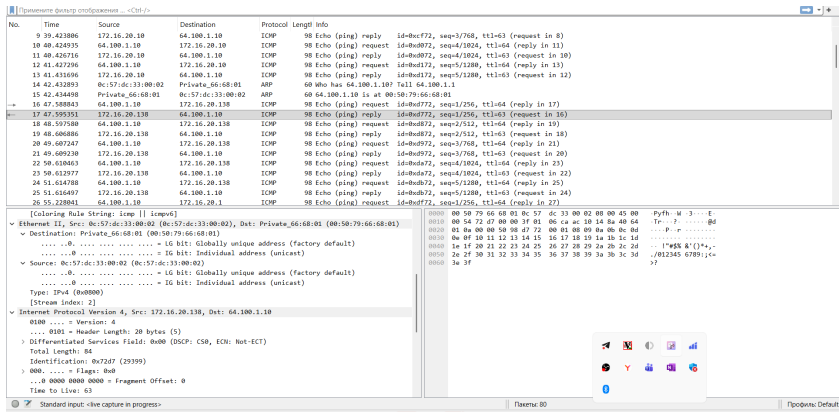


Рис. 6.36: ICMP

В анализируемом трафике наблюдается успешный обмен ICMP-пакетами: запросы (Echo Request) от 64.100.1.10 к 172.16.20.10 и ответы (Echo Reply) в обратном

направлении. Длина ICMP-пакетов составляет 98 байт, интервал между запросами - примерно 1 секунда. Разница в TTL (64 vs 63) подтверждает прохождение пакетов через маршрутизатор.

Протокол ARP работает на канальном уровне и обеспечивает преобразование IP-адресов в MAC-адреса. Длина ARP-кадра составляет 60 байт, идентификатор протокола в Ethernet-фрейме - 0x0806. В ARP-запросах source содержит MAC и IP адреса отправителя, а destination - широковещательный MAC-адрес и IP целевого устройства.

Оба протокола демонстрируют корректную работу: ICMP подтверждает сквозную связность между узлами, а ARP обеспечивает корректное разрешение адресов на канальном уровне. Сетевое соединение стабильно, потерь пакетов не наблюдается.

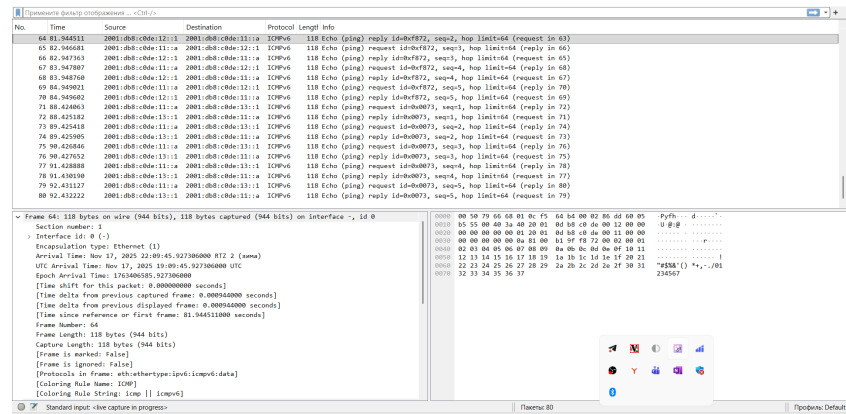


Рис. 6.37: ICMPv6

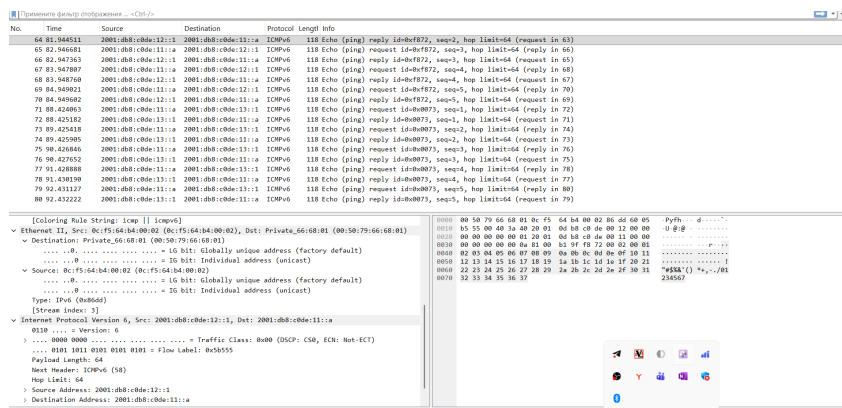


Рис. 6.38: ICMPv6

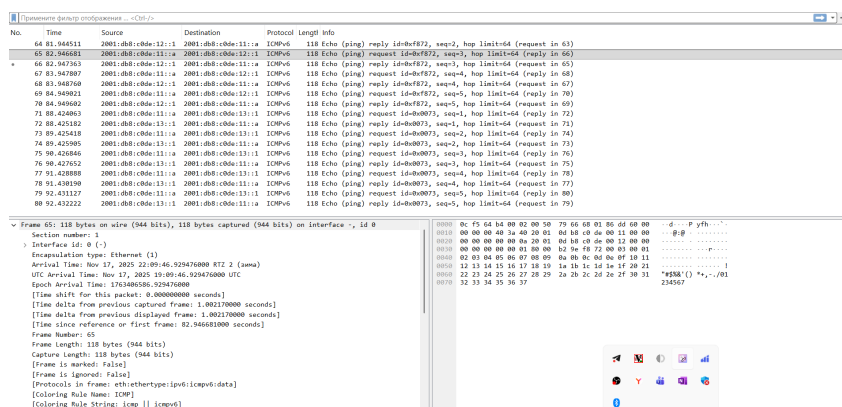


Рис. 6.39: ICMPv6

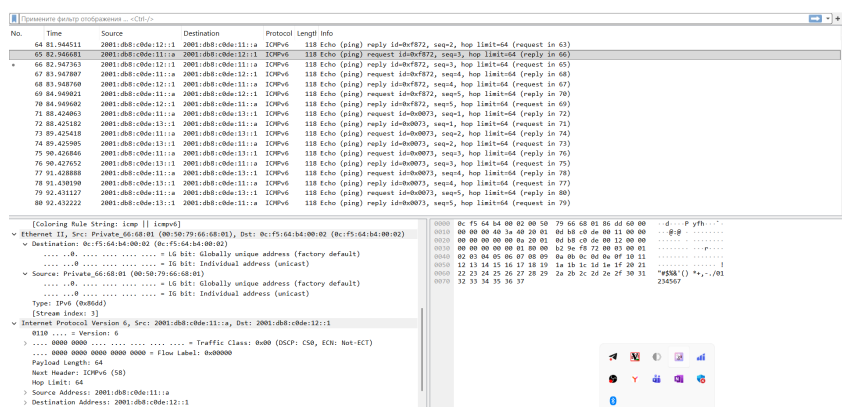


Рис. 6.40: ICMPv6

Протокол ICMPv6 используется для диагностики сетевых соединений в IPv6-сетях. В анализируемом трафике наблюдается успешный обмен ICMPv6-пакетами

между адресами 2001:db8:c0de:12::1 и 2001:db8:c0de:11::a. Длина ICMPv6-пакетов составляет 118 байт, что больше чем у IPv4-аналогов из-за увеличенного размера IPv6-заголовков.

Механизм запрос-ответ работает корректно: Echo Request отправляются от 2001:db8:c0de:11::a к 2001:db8:c0de:12::1, а Echo Reply следуют в обратном направлении. Параметры пакетов включают идентификатор (id=0x6782) и порядковые номера (seq), которые совпадают в соответствующих запросах и ответах. Норм Limit установлен на 64, что является стандартным значением для IPv6.

На канальном уровне видна корректная адресация: MAC-адреса отправителя и получателя правильно сопоставлены с IPv6-адресами. Протокол демонстрирует стабильную работу без потерь пакетов, подтверждая успешную IPv6-связность между узлами сети.

7 Характеристика подсетей

В рамках задания были созданы две независимые подсети с адресацией (IPv4 и IPv6).

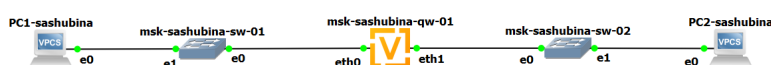


Рис. 7.1: создание сети

Первая подсеть (eth0 - соединение с PC1) использует IPv4 сеть 10.10.1.96/27 с маской 255.255.255.224. Данная подсеть охватывает диапазон адресов от 10.10.1.96 до 10.10.1.127, где 10.10.1.96 является сетевым адресом, а 10.10.1.127 - broadcast-адресом. Для назначения устройствам доступны адреса с 10.10.1.97 по 10.10.1.126, что предоставляет 30 usable IP-адресов. Параллельно для IPv6 назначена сеть 2001:db8:1:1::/64, обеспечивающая значительно большее адресное пространство.

Вторая подсеть (eth1 - соединение с PC2) построена на сети 10.10.1.16/28 с маской 255.255.255.240. Эта подсеть включает адреса от 10.10.1.16 до 10.10.1.31, где 10.10.1.16 - сетевой адрес, 10.10.1.31 - broadcast. Доступные для назначения адреса находятся в диапазоне 10.10.1.17-10.10.1.30, предоставляя 14 IP-адресов. Для IPv6 используется сеть 2001:db8:1:4::/64.

Таблица адресации и выбор наименьших адресов Была разработана комплексная таблица адресации, где особое внимание уделено выполнению требования о назначении наименьших адресов на интерфейсы маршрутизатора:

Для маршрутизатора на интерфейсе eth0 назначен адрес 10.10.1.97/27, который

является первым доступным адресом в подсети 10.10.1.96/27. Адрес 10.10.1.96 не может быть использован, так как является сетевым адресом. Соответственно, 10.10.1.97 действительно представляет собой наименьший возможный адрес для назначения устройству в данной подсети.

Для маршрутизатора на интерфейсе eth1 выбран адрес 10.10.1.17/28 - первый доступный адрес в подсети 10.10.1.16/28. Адрес 10.10.1.16 является сетевым, поэтому 10.10.1.17 становится минимальным адресом для использования в этой подсети.

Конечные устройства получили следующие адреса: PC1 - 10.10.1.98/27 с шлюзом 10.10.1.97, PC2 - 10.10.1.18/28 с шлюзом 10.10.1.17. Для IPv6 адресация построена по аналогичному принципу: маршрутизатор использует ::1 в каждой подсети, а конечные устройства - ::2.

7.1 Настроим IP-адресацию на маршрутизаторе VyOS и оконечных устройствах, причём на интерфейсах маршрутизатора установить наименьший адрес в подсети.

```
vyos@msk-sashubina-gw-01# set interfaces ethernet eth0 address '10.10.1.97/27'

Configuration path: [interfaces ethernet eth0 address 10.10.1.97/27] already exists

[edit]
vyos@msk-sashubina-gw-01# set interfaces ethernet eth0 address '2001:db8:1:1::1/64'

Configuration path: [interfaces ethernet eth0 address 2001:db8:1:1::1/64] already exists

[edit]
vyos@msk-sashubina-gw-01# set service router-advert interface eth0 prefix '2001:db8:1:1::/64'

Configuration path: [service router-advert interface eth0 prefix 2001:db8:1:1::/64] already exists

[edit]
vyos@msk-sashubina-gw-01# set interfaces ethernet eth1 address '10.10.1.17/28'

Configuration path: [interfaces ethernet eth1 address 10.10.1.17/28] already exists

[edit]
vyos@msk-sashubina-gw-01# set interfaces ethernet eth1 address '2001:db8:1:4::1/64'

Configuration path: [interfaces ethernet eth1 address 2001:db8:1:4::1/64] already exists

[edit]
vyos@msk-sashubina-gw-01# set service router-advert interface eth1 prefix '2001:db8:1:4::/64'

Configuration path: [service router-advert interface eth1 prefix 2001:db8:1:4::/64] already exists

[edit]
vyos@msk-sashubina-gw-01# commit
[edit]
vyos@msk-sashubina-gw-01# save
Saving configuration to '/config/config.boot'...
Done
[edit]
vyos@msk-sashubina-gw-01# █
```

Рис. 7.2: Vyos

```
vyos@msk-sashubina-gw-01# 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
}
ethernet eth2 {
    hw-id 0c:d0:c7:39:00:02
}
ethernet eth3 {
    hw-id 0c:d0:c7:39:00:03
}
ethernet eth4 {
    hw-id 0c:d0:c7:39:00:04
}
ethernet eth5 {
    hw-id 0c:d0:c7:39:00:05
}
ethernet eth6 {
    hw-id 0c:d0:c7:39:00:06
}
ethernet eth7 {
    hw-id 0c:d0:c7:39:00:07
}
ethernet eth8 {
    hw-id 0c:d0:c7:39:00:08
}
ethernet eth9 {
    hw-id 0c:d0:c7:39:00:09
}
loopback lo {
}
[edit]
vyos@msk-sashubina-gw-01#
[edit]
```

Рис. 7.3: просмотр интерфейсов

```
VPCS> ip 10.10.1.98/27 10.10.1.97
Checking for duplicate address...
VPCS : 10.10.1.98 255.255.255.224 gateway 10.10.1.97

VPCS> ip 2001:db8:1:1::2/64
PC1 : 2001:db8:1:1::2/64

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

VPCS> █
```

Рис. 7.4: PC1

```

VPCS> ip 10.10.1.18/28 10.10.1.17
Checking for duplicate address...
VPCS : 10.10.1.18 255.255.255.240 gateway 10.10.1.17

VPCS> ip 2001:db8:1:4::2/64
PC1 : 2001:db8:1:4::2/64

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

VPCS> █

```

Рис. 7.5: PC2

```

VPCS> show ip

NAME           : VPCS[1]
IP/MASK        : 10.10.1.98/27
GATEWAY        : 10.10.1.97
DNS            :
MAC            : 00:50:79:66:68:00
LPORT         : 20000
RHOST:PORT     : 127.0.0.1:20001
MTU            : 1500

VPCS> show ipv6

NAME           : VPCS[1]
LINK-LOCAL SCOPE : fe80::250:79ff:fe66:6800/64
GLOBAL SCOPE    : 2001:db8:1:1::2/64
DNS            :
ROUTER LINK-LAYER : 0c:d0:c7:39:00:00
MAC            : 00:50:79:66:68:00
LPORT         : 20000
RHOST:PORT     : 127.0.0.1:20001
MTU            : 1500

VPCS> █

```

Рис. 7.6: проверка PC1

```
VPCS> show ip
NAME      : VPCS[1]
IP/MASK    : 10.10.1.18/28
GATEWAY    : 10.10.1.17
DNS        :
MAC        : 00:50:79:66:68:01
LPORT     : 20002
RHOST:PORT : 127.0.0.1:20003
MTU        : 1500

VPCS> show ipv6
NAME      : VPCS[1]
LINK-LOCAL SCOPE : fe80::250:79ff:fe66:6801/64
GLOBAL SCOPE    : 2001:db8:1:4::2/64
DNS            :
ROUTER LINK-LAYER : 0c:d0:c7:39:00:01
MAC            : 00:50:79:66:68:01
LPORT         : 20002
RHOST:PORT     : 127.0.0.1:20003
MTU           : 1500

VPCS> █
```

Рис. 7.7: проверка PC2

7.2 Проверка корректности реализации

Все ping-запросы между устройствами успешно выполняются


```
VPCS> ping 10.10.1.97

84 bytes from 10.10.1.97 icmp_seq=1 ttl=64 time=1.005 ms
84 bytes from 10.10.1.97 icmp_seq=2 ttl=64 time=2.020 ms
84 bytes from 10.10.1.97 icmp_seq=3 ttl=64 time=0.928 ms
84 bytes from 10.10.1.97 icmp_seq=4 ttl=64 time=1.973 ms
84 bytes from 10.10.1.97 icmp_seq=5 ttl=64 time=0.809 ms

VPCS> ping 2001:db8:1:1::1

2001:db8:1:1::1 icmp6_seq=1 ttl=64 time=2.097 ms
2001:db8:1:1::1 icmp6_seq=2 ttl=64 time=0.905 ms
2001:db8:1:1::1 icmp6_seq=3 ttl=64 time=0.694 ms
2001:db8:1:1::1 icmp6_seq=4 ttl=64 time=0.951 ms
2001:db8:1:1::1 icmp6_seq=5 ttl=64 time=0.526 ms

VPCS> ping 10.10.1.18

84 bytes from 10.10.1.18 icmp_seq=1 ttl=63 time=2.944 ms
84 bytes from 10.10.1.18 icmp_seq=2 ttl=63 time=1.941 ms
84 bytes from 10.10.1.18 icmp_seq=3 ttl=63 time=2.338 ms
84 bytes from 10.10.1.18 icmp_seq=4 ttl=63 time=1.104 ms
84 bytes from 10.10.1.18 icmp_seq=5 ttl=63 time=0.909 ms

VPCS> ping 2001:db8:1:4::2

2001:db8:1:4::2 icmp6_seq=1 ttl=62 time=2.749 ms
2001:db8:1:4::2 icmp6_seq=2 ttl=62 time=1.113 ms
2001:db8:1:4::2 icmp6_seq=3 ttl=62 time=1.110 ms
2001:db8:1:4::2 icmp6_seq=4 ttl=62 time=1.953 ms
2001:db8:1:4::2 icmp6_seq=5 ttl=62 time=1.592 ms

VPCS> █
```

Рис. 7.8: PC1

```
VPCS> ping 10.10.1.17

84 bytes from 10.10.1.17 icmp_seq=1 ttl=64 time=3.952 ms
84 bytes from 10.10.1.17 icmp_seq=2 ttl=64 time=0.882 ms
84 bytes from 10.10.1.17 icmp_seq=3 ttl=64 time=0.668 ms
84 bytes from 10.10.1.17 icmp_seq=4 ttl=64 time=0.777 ms
84 bytes from 10.10.1.17 icmp_seq=5 ttl=64 time=0.650 ms

VPCS> ping 2001:db8:1:4::1

2001:db8:1:4::1 icmp6_seq=1 ttl=64 time=1.026 ms
2001:db8:1:4::1 icmp6_seq=2 ttl=64 time=1.003 ms
2001:db8:1:4::1 icmp6_seq=3 ttl=64 time=0.646 ms
2001:db8:1:4::1 icmp6_seq=4 ttl=64 time=0.709 ms
2001:db8:1:4::1 icmp6_seq=5 ttl=64 time=1.746 ms

VPCS> ping 10.10.1.98

84 bytes from 10.10.1.98 icmp_seq=1 ttl=63 time=3.619 ms
84 bytes from 10.10.1.98 icmp_seq=2 ttl=63 time=1.357 ms
84 bytes from 10.10.1.98 icmp_seq=3 ttl=63 time=1.247 ms
84 bytes from 10.10.1.98 icmp_seq=4 ttl=63 time=1.646 ms
84 bytes from 10.10.1.98 icmp_seq=5 ttl=63 time=1.418 ms

VPCS> ping 2001:db8:1:1::2

2001:db8:1:1::2 icmp6_seq=1 ttl=62 time=7.218 ms
2001:db8:1:1::2 icmp6_seq=2 ttl=62 time=4.054 ms
2001:db8:1:1::2 icmp6_seq=3 ttl=62 time=4.716 ms
2001:db8:1:1::2 icmp6_seq=4 ttl=62 time=5.153 ms
2001:db8:1:1::2 icmp6_seq=5 ttl=62 time=7.705 ms

VPCS> █
```

Рис. 7.9: PC2

```

PING 10.10.1.98 (10.10.1.98) 56(84) bytes of data.
64 bytes from 10.10.1.98: icmp_seq=1 ttl=64 time=1.40 ms
64 bytes from 10.10.1.98: icmp_seq=2 ttl=64 time=0.786 ms
64 bytes from 10.10.1.98: icmp_seq=3 ttl=64 time=0.665 ms
64 bytes from 10.10.1.98: icmp_seq=4 ttl=64 time=0.523 ms
64 bytes from 10.10.1.98: icmp_seq=5 ttl=64 time=1.12 ms
64 bytes from 10.10.1.98: icmp_seq=6 ttl=64 time=0.743 ms
64 bytes from 10.10.1.98: icmp_seq=7 ttl=64 time=0.726 ms
64 bytes from 10.10.1.98: icmp_seq=8 ttl=64 time=0.599 ms
64 bytes from 10.10.1.98: icmp_seq=9 ttl=64 time=0.887 ms
64 bytes from 10.10.1.98: icmp_seq=10 ttl=64 time=0.794 ms
64 bytes from 10.10.1.98: icmp_seq=11 ttl=64 time=3.74 ms
^Z
[1]+  Stopped                  ping 10.10.1.98
[edit]
vyos@msk-sashubina-gw-01# ping 2001:db8:1:1::2
PING 2001:db8:1:1::2(2001:db8:1:1::2) 56 data bytes
64 bytes from 2001:db8:1:1::2: icmp_seq=1 ttl=64 time=6.51 ms
64 bytes from 2001:db8:1:1::2: icmp_seq=2 ttl=64 time=0.701 ms
64 bytes from 2001:db8:1:1::2: icmp_seq=3 ttl=64 time=0.716 ms
64 bytes from 2001:db8:1:1::2: icmp_seq=4 ttl=64 time=0.567 ms
64 bytes from 2001:db8:1:1::2: icmp_seq=5 ttl=64 time=6.90 ms
64 bytes from 2001:db8:1:1::2: icmp_seq=6 ttl=64 time=0.611 ms
64 bytes from 2001:db8:1:1::2: icmp_seq=7 ttl=64 time=2.66 ms
64 bytes from 2001:db8:1:1::2: icmp_seq=8 ttl=64 time=1.07 ms
^Z
[2]+  Stopped                  ping 2001:db8:1:1::2
[edit]
vyos@msk-sashubina-gw-01# ping 10.10.1.18
PING 10.10.1.18 (10.10.1.18) 56(84) bytes of data.
64 bytes from 10.10.1.18: icmp_seq=1 ttl=64 time=2.30 ms
64 bytes from 10.10.1.18: icmp_seq=2 ttl=64 time=0.599 ms
64 bytes from 10.10.1.18: icmp_seq=3 ttl=64 time=0.667 ms
64 bytes from 10.10.1.18: icmp_seq=4 ttl=64 time=0.871 ms
64 bytes from 10.10.1.18: icmp_seq=5 ttl=64 time=0.451 ms
64 bytes from 10.10.1.18: icmp_seq=6 ttl=64 time=0.505 ms
^Z
[3]+  Stopped                  ping 10.10.1.18
[edit]
vyos@msk-sashubina-gw-01# ping 2001:db8:1:4::2
PING 2001:db8:1:4::2(2001:db8:1:4::2) 56 data bytes
64 bytes from 2001:db8:1:4::2: icmp_seq=1 ttl=64 time=3.05 ms
64 bytes from 2001:db8:1:4::2: icmp_seq=2 ttl=64 time=0.903 ms
64 bytes from 2001:db8:1:4::2: icmp_seq=3 ttl=64 time=0.809 ms
64 bytes from 2001:db8:1:4::2: icmp_seq=4 ttl=64 time=7.04 ms
^Z
[4]+  Stopped                  ping 2001:db8:1:4::2
[edit]
vyos@msk-sashubina-gw-01# █

```

Рис. 7.10: маршрутизатор

Traceroute демонстрирует правильные пути прохождения трафик

```

VPCS> trace 10.10.1.18
trace to 10.10.1.18, 8 hops max, press Ctrl+C to stop
 1  10.10.1.97  6.928 ms  1.026 ms  0.454 ms
 2  *10.10.1.18  2.613 ms (ICMP type:3, code:3, Destination port unreachable)

VPCS> trace 2001:db8:1:4::2
trace to 2001:db8:1:4::2, 64 hops max
 1  2001:db8:1:1::1  3.787 ms  0.500 ms  0.536 ms
 2  2001:db8:1:4::2  2.276 ms  1.145 ms  0.845 ms

VPCS> █

```

Рис. 7.11: PC1

```
VPCS> trace 10.10.1.98
trace to 10.10.1.98, 8 hops max, press Ctrl+C to stop
 1  10.10.1.17  0.675 ms  0.564 ms  0.291 ms
 2  *10.10.1.98  0.994 ms (ICMP type:3, code:3, Destination port unreachable)

VPCS> trace 2001:db8:1:1::2
trace to 2001:db8:1:1::2, 64 hops max
 1 2001:db8:1:4::1  1.267 ms  0.556 ms  0.309 ms
 2 2001:db8:1:1::2  1.356 ms  1.190 ms  0.749 ms

VPCS> █
```

Рис. 7.12: PC2

Сообщения “port unreachable” в traceroute являются нормальным поведением и подтверждают достижение пакетами целевых узлов

IPv4 и IPv6 соединения работают стабильно и параллельно. Маршрутизатор корректно выполняет функции шлюза для обеих подсетей. Таким образом, задание выполнено полностью в соответствии с требованиями, включая корректный выбор наименьших адресов для интерфейсов маршрутизатора и обеспечение работоспособности всей сетевой инфраструктуры.

8 Вывод

Я изучила принципы распределения и настройки адресного пространства на устройствах сети.