

Отчёт по лабораторной работе №7

Сетевые технологии

Ищенко Ирина НПИбд-02-22

Содержание

1 Цель работы	5
2 Выполнение лабораторной работы	6
3 Выводы	29

Список иллюстраций

2.1 Топология моделируемой сети	6
2.2 Настройка gw-01: изменение имени, домена, пользователя	7
2.3 Настройка gw-01: удаление пользователя по умолчанию	7
2.4 Настройка gw-01: IPv4-адресация и DHCP-сервер	8
2.5 Настройка PC1 и проверка конфигурации	9
2.6 Журнал работы	9
2.7 Просмотр статистики DHCP, выданных адресов и журнала работы	10
2.8 Проверка захваченных анализатором трафика пакетов	11
2.9 Проверка захваченных анализатором трафика пакетов	12
2.10 Топология дополненной сети	14
2.11 Настройка gw-01: IPv6-адресация	14
2.12 Настройка gw-01: DHCPv6 без отслеживания состояния	15
2.13 Проверка настроек сети на PC2, пинг маршрутизатора, проверка DNS	16
2.14 Пинг маршрутизатора, проверка DNS	17
2.15 Получение адреса на PC2, пинг маршрутизатора, проверка DNS . .	18
2.16 Просмотр статистики DHCP, выданных адресов, пинг PC2	18
2.17 Проверка захваченных анализатором трафика пакетов	19
2.18 Проверка захваченных анализатором трафика пакетов	20
2.19 Настройка gw-01: DHCPv6 с отслеживанием состояния	21
2.20 Проверка настроек сети и DNS на PC3	22
2.21 Проверка настроек сети и DNS на PC3	23
2.22 Получение адреса на PC3	24
2.23 Проверка настроек сети на PC3	25
2.24 Проверка настроек сети на PC3	26
2.25 Пинг маршрутизатора	26
2.26 Просмотр статистики DHCP и выданных адресов	27
2.27 Проверка захваченных анализатором трафика пакетов	27

Список таблиц

1 Цель работы

Получение навыков настройки службы DHCP на сетевом оборудовании для распределения адресов IPv4 и IPv6.

2 Выполнение лабораторной работы

Запускаю GNS3 VM и GNS3. Создаю новый проект. В рабочей области GNS3 размещаю коммутатор Ethernet, маршрутизатор VyOS и хост VPCS. Формирую топологию сети согласно инструкции и таблице адресации. Включаю захват трафика между коммутатором и маршрутизатором. Включаю все устройства сети (рис. 2.1).

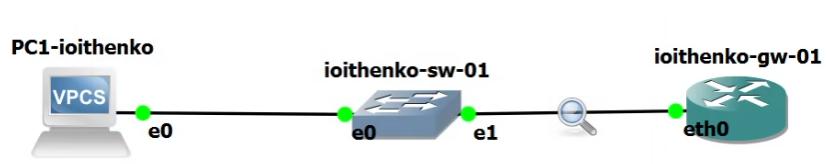


Рис. 2.1: Топология моделируемой сети

Перехожу к настройке маршрутизатора. Изменяю имя устройства, доменное имя, системного пользователя (рис. 2.2).

```
Welcome to VyOS - vyos ttyS0
vyos login: vyos
Password:
Welcome to VyOS!

Check out project news at https://blog.vyos.io
and feel free to report bugs at https://vyos.dev

You can change this banner using "set system login banner post-login" command.

VyOS is a free software distribution that includes multiple components,
you can check individual component licenses under /usr/share/doc/*copyright
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 ioithenko-gw-01
[edit]
vyos@vyos# set system domain-name ioithenko.net
[edit]
vyos@vyos# set system login user ioithenko
[edit]
456
[edit]
vyos@vyos# compare
[edit system]
+domain-name ioithenko.net
>host-name ioithenko-gw-01
[edit system login]
+user ioithenko {
+  authentication {
+    plaintext-password 123456
+  }
+
[edit]
vyos@vyos# commit
      Invalid command: [commit]

[edit]
vyos@vyos# commit
[edit]
vyos@vyos# save
Saving configuration to '/config/config.boot'...
Done
[edit]
vyos@vyos# exit
exit
vyos@vyos:~$ exit
logout

Welcome to VyOS - ioithenko-gw-01 ttyS0
```

Рис. 2.2: Настройка gw-01: изменение имени, домена, пользователя

Удаляю пользователя по умолчанию (рис. 2.3).

```
Welcome to VyOS - ioithenko-gw-01 ttyS0
ioithenko-gw-01 login: ioithenko
Password:
Welcome to VyOS!

Check out project news at https://blog.vyos.io
and feel free to report bugs at https://vyos.dev

You can change this banner using "set system login banner post-login" command.

VyOS is a free software distribution that includes multiple components,
you can check individual component licenses under /usr/share/doc/*copyright
ioithenko@ioithenko-gw-01:~$ configure
[edit]
ioithenko@ioithenko-gw-01# delete system login user vyos
[edit]
ioithenko@ioithenko-gw-01# commit
[edit]
ioithenko@ioithenko-gw-01# save
Saving configuration to '/config/config.boot'...
Done
[edit]
```

Рис. 2.3: Настройка gw-01: удаление пользователя по умолчанию

Настраиваю IPv4-адресацию и конфигурацию DHCP-сервера на `msk-ioithenko-gw-01` (рис. 2.4).

```
ioithenko@ioithenko-gw-01:~$ configure
[edit]
/24thenko@ioithenko-gw-01# set service interfaces ethernet eth0 address 10.0.0.1
    Configuration path: servi
[edit]
ioithenko@ioithenko-gw-01# exit discard          net eth0 address 10.0.0.1/24.0.1/
exit0.1/240.1/240.1/24                           net eth0 address 10.0.0.1/24
ioithenko@ioithenko-gw-01:~$ t service interfaces ethernet eth0 address 10.0.0.1/
ioithenko@ioithenko-gw-01:~$ configure
[edit]
ioithenko@ioithenko-gw-01# set interfaces ethernet eth0 address 10.0.0.1/24
[edit]
domain-name ioithenko.net set service dhcp-server shared-network-name ioithenko
[edit]
name-server 10.0.0.1
[edit]
subnet 10.0.0.0/24 default-router 10.0.0.1-server shared-network-name ioithenko
[edit]
subnet 10.0.0.0/24 range hosts start 10.0.0.2
[edit]
subnet 10.0.0.0/24 range hosts stop 10.0.0.253
[edit]
ioithenko@ioithenko-gw-01# compare
[edit interfaces ethernet eth0]
+address 10.0.0.1/24
[edit service]
+dhcp-server {
+    shared-network-name ioithenko {
+        domain-name ioithenko.net
+        name-server 10.0.0.1
+        subnet 10.0.0.0/24 {
+            default-router 10.0.0.1
+            range hosts {
+                start 10.0.0.2
+                stop 10.0.0.253
+            }
+        }
+    }
+}
[edit]
```

Рис. 2.4: Настройка gw-01: IPv4-адресация и DHCP-сервер

Здесь при помощи указанных выше команд была создана разделяемая сеть (`shared-network-name`) с названием `ioithenko`, подсеть (`subnet`) с адресом `10.0.0.0/24`, задан диапазон адресов (`range`) с именем `hosts`, содержащий адреса `10.0.0.2 – 10.0.0.253`.

Настраиваю PC1 и проверяю конфигурацию (рис. 2.5). Для устройства был выдан адрес `10.0.0.2/24`.

```

Opcode: 1 (REQUEST)
Client IP Address: 10.0.0.2
Your IP Address: 0.0.0.0
Server IP Address: 0.0.0.0
Gateway IP Address: 0.0.0.0
Client MAC Address: 00:50:79:66:68:00
Option 53: Message Type = Request
Option 54: DHCP Server = 10.0.0.1
Option 50: Requested IP Address = 10.0.0.2
Option 61: Client Identifier = Hardware Type=Ethernet MAC Address = 00:50:79:66:68:00
Option 12: Host Name = VPCS

Opcode: 2 (REPLY)
Client IP Address: 10.0.0.2
Your IP Address: 10.0.0.2
Server IP Address: 0.0.0.0
Gateway IP Address: 0.0.0.0
Client MAC Address: 00:50:79:66:68:00
Option 53: Message Type = Ack
Option 54: DHCP Server = 10.0.0.1
Option 51: Lease Time = 86400
Option 1: Subnet Mask = 255.255.255.0
Option 3: Router = 10.0.0.1
Option 6: DNS Server = 10.0.0.1
Option 15: Domain = ioithenko.net

IP 10.0.0.2/24 GW 10.0.0.1

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

VPCS> show ip

NAME      : VPCS[1]
IP/MASK   : 10.0.0.2/24
GATEWAY   : 10.0.0.1
DNS        : 10.0.0.1
DHCP SERVER : 10.0.0.1
DHCP LEASE  : 86343, 86400/43200/75600
DOMAIN NAME : ioithenko.net
MAC        : 00:50:79:66:68:00
LPORT      : 20004
RHOST:PORT : 127.0.0.1:20005
MTU        : 1500

VPCS> ping 10.0.0.1 -c 2
84 bytes from 10.0.0.1 icmp_seq=1 ttl=64 time=0.911 ms
84 bytes from 10.0.0.1 icmp_seq=2 ttl=64 time=1.018 ms
VPCS> I

```

Рис. 2.5: Настройка PC1 и проверка конфигурации

Использую команды для просмотра статистики DHCP-сервера и выданных адресов и вижу в списке выданный адрес для PC1 (рис. 2.6). Также просматриваю журнал работы DHCP (рис. 2.7).

```

ioithenko@ioithenko-gw-01:~$ show dhcp server statistics
Pool      Size    Leases   Available Usage
-----  -----
ioithenko  252       1      251  0%
ioithenko@ioithenko-gw-01:~$ show dhcp server leases
IP address  Hardware address  State  Lease start      Lease expiration  Re
maining     Pool      Hostname
-----  -----
10.0.0.2    00:50:79:66:68:00  active  2024/12/04 12:34:31  2024/12/05 12:34:31  23
:56:53      ioithenko  VPCS
ioithenko@ioithenko-gw-01:~$ I

```

Рис. 2.6: Журнал работы

```
, data : [ dhcp-eth0 ]}
Dec 04 12:33:51 sudo[16287]: ioithenko : TTY=ttyS0 ; PWD=/home/ioithenko ; USER=root ;
COMMAND=/usr/bin/sh -c /usr/sbin/vyshim /usr/libexec/vyos/conf_mode/dhcp_server.py
Dec 04 12:33:51 vyos-configd[611]: Received message: {"type": "node", "data": "/usr/lib
exec/vyos/conf_mode/dhcp_server.py"}
Dec 04 12:33:53 dhcpd[16309]: Wrote 0 leases to leases file.
Dec 04 12:33:53 dhcpd[16309]: Lease file test successful, removing temp lease file: /co
nfig/dhcpd.leases.1733315633
Dec 04 12:33:53 dhcpd[16311]: Wrote 0 leases to leases file.
Dec 04 12:33:53 dhcpd[16311]:
Dec 04 12:33:53 dhcpd[16311]: No subnet declaration for eth2 (no IPv4 addresses).
Dec 04 12:33:53 dhcpd[16311]: ** Ignoring requests on eth2. If this is not what
Dec 04 12:33:53 dhcpd[16311]: you want, please write a subnet declaration
Dec 04 12:33:53 dhcpd[16311]: in your dhcpd.conf file for the network segment
Dec 04 12:33:53 dhcpd[16311]: to which interface eth2 is attached. **
Dec 04 12:33:53 dhcpd[16311]:
Dec 04 12:33:53 dhcpd[16311]: No subnet declaration for eth1 (no IPv4 addresses).
Dec 04 12:33:53 dhcpd[16311]: ** Ignoring requests on eth1. If this is not what
Dec 04 12:33:53 dhcpd[16311]: you want, please write a subnet declaration
Dec 04 12:33:53 dhcpd[16311]: in your dhcpd.conf file for the network segment
Dec 04 12:33:53 dhcpd[16311]: to which interface eth1 is attached. **
Dec 04 12:33:53 dhcpd[16311]:
Dec 04 12:33:53 dhcpd[16311]: Server starting service.
Dec 04 12:34:27 dhcpd[16311]: DHCPDISCOVER from 00:50:79:66:68:00 via eth0
Dec 04 12:34:28 dhcpd[16311]: DHCPOFFER on 10.0.0.2 to 00:50:79:66:68:00 (VPCS) via eth
0
Dec 04 12:34:31 dhcpd[16311]: DHCPREQUEST for 10.0.0.2 (10.0.0.1) from 00:50:79:66:68:0
0 (VPCS) via eth0
Dec 04 12:34:31 dhcpd[16311]: DHCPACK on 10.0.0.2 to 00:50:79:66:68:00 (VPCS) via eth0
Dec 04 12:37:14 sudo[16387]: ioithenko : TTY=ttyS0 ; PWD=/home/ioithenko ; USER=root ;
COMMAND=/usr/libexec/vyos/op_mode/show_dhcp.py --statistics
Dec 04 12:37:37 sudo[16413]: ioithenko : TTY=ttyS0 ; PWD=/home/ioithenko ; USER=root ;
COMMAND=/usr/libexec/vyos/op_mode/show_dhcp.py --leases
ioithenko@ioithenko-gw-01:~$
```

Рис. 2.7: Просмотр статистики DHCP, выданных адресов и журнала работы

Просматриваю захваченные анализатором трафика пакеты (рис. 2.8 и 2.9).

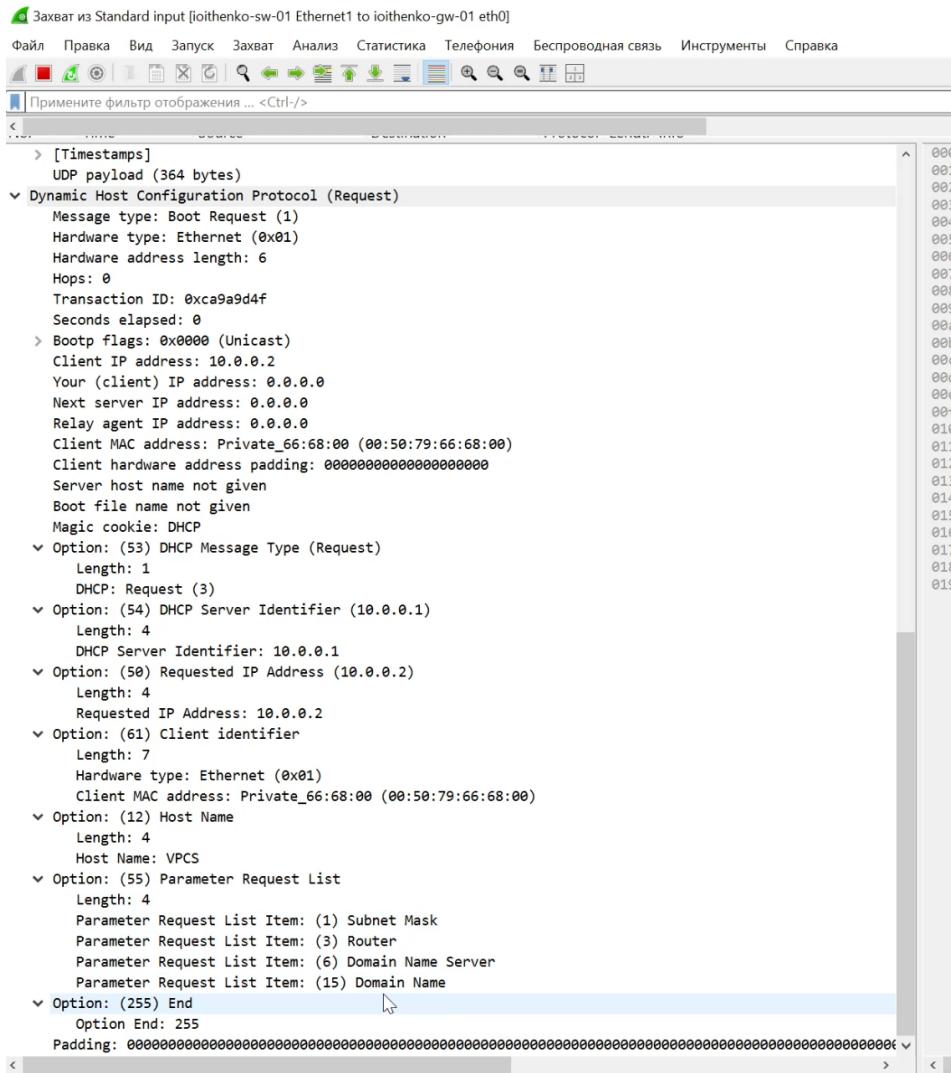


Рис. 2.8: Проверка захваченных анализатором трафика пакетов

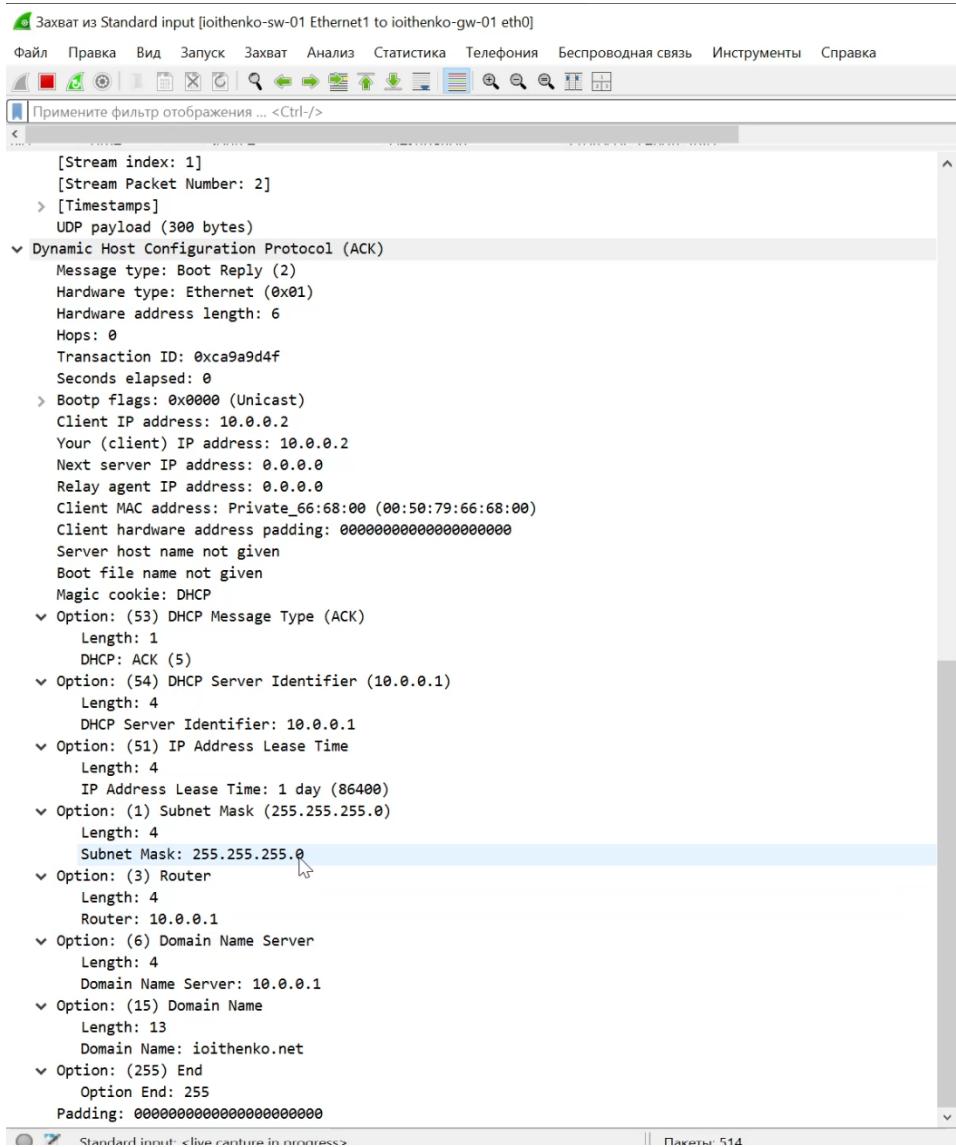


Рис. 2.9: Проверка захваченных анализатором трафика пакетов

Процесс получения устройством адреса по протоколу DHCP происходит в 4 шага:

- «DHCP DISCOVER»: устройство отправляет широковещательный запрос, в котором во фрейме (PDU канального уровня) в поле адреса отправителя указывается MAC-адрес устройства, а в поле адреса получателя — широковещательный адрес ffff.ffff.ffff; в пакете (PDU сетевого уровня) в поле адреса отправителя указан адрес 0.0.0.0, а в поле адреса получателя — адрес

255.255.255.255;

- «DHCP OFFER»: DHCP-сервер после получения широковещательного сообщения выделяет (но не резервирует) в своём пуле адресов некоторый адрес DHCP-клиенту на заданное время (lease time), назначает другие настройки (опции) и пересыпает всю информацию DHCP клиенту; при этом в соответствующих полях получателя в сообщении указываются выделенный клиенту IP-адрес и его MAC-адрес. В нашем случае адрес отправителя – 10.0.0.1 – адрес DHCP-сервера, а адрес получателя 10.0.0.2.
- «DHCP REQUEST»: клиент отправляет DHCP-серверу согласие с полученными параметрами;
- «DHCP ACKNOWLEDGE»: DHCP-сервер резервирует за DHCP-клиентом выделенный адрес на какое-то время (lease time), вносит информацию в свою ARP-таблицу и высыпает DHCP-клиенту сообщение об успешной регистрации адреса.

Дополняю предыдущую сеть в соответствии с топологией из инструкции. Использую хост Kali Linux 2019 (использовать Kali Linux CLI не получилось, так как воспринимались не все необходимые команды) (рис. 2.10). Включаю захват трафика на соединениях между маршрутизатором gw-01 и коммутаторами sw-02 и sw-03.

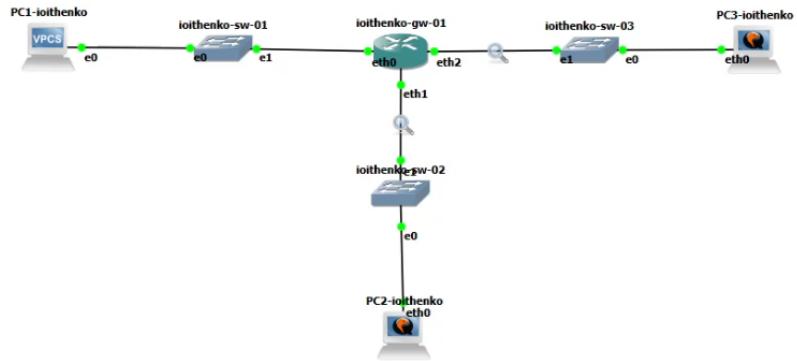


Рис. 2.10: Топология дополненной сети

Настраиваю IPv6-адресацию на маршрутизаторе (рис. 2.11).

```
Welcome to VyOS - ioithenko-gw-01 ttyS0
ioithenko-gw-01 login: ioithenko
Password:
Welcome to VyOS!

Check out project news at https://blog.vyos.io
and feel free to report bugs at https://vyos.dev

You can change this banner using "set system login banner post-login" command.

VyOS is a free software distribution that includes multiple components,
you can check individual component licenses under /usr/share/doc/*/copyright
ioithenko@ioithenko-gw-01:~$ configure
[edit]
ioithenko@ioithenko-gw-01# set interfaces ethernet eth1 address 2000::1/64
[edit]
ioithenko@ioithenko-gw-01# set interfaces ethernet eth2 address 2001::1/64
[edit]
ioithenko@ioithenko-gw-01# show interfaces
  ethernet eth0 {
    address 10.0.0.1/24
    hw-id 0c:a9:fa:46:00:00
  }
  ethernet eth1 {
+    address 2000::1/64
    hw-id 0c:a9:fa:46:00:01
  }
  ethernet eth2 {
+    address 2001::1/64
    hw-id 0c:a9:fa:46:00:02
  }
  loopback lo {
  }
[edit]
ioithenko@ioithenko-gw-01# commit
[edit]
ioithenko@ioithenko-gw-01# save
Saving configuration to '/config/config.boot'...
Done
[edit]
```

Рис. 2.11: Настройка gw-01: IPv6-адресация

На маршрутизаторе настраиваю DHCPv6 без отслеживания состояния (DHCPv6 Stateless configuration) (рис. 2.12).

```

[edit]
ko-stateless thenko-gw-01# set service dhcipv6-server shared-network-name ioithen
[edit]
ko-stateless subnet 2000::0/64 service dhcipv6-server shared-network-name ioithen
[edit]
ko-stateless common-options name-server 2000::1server shared-network-name ioithen
[edit]
ko-stateless common-options domain-search ioithenko.netared-network-name ioithen
[edit]
ioithenko@ioithenko-gw-01# compare
[edit service]
+dhcipv6-server {
+    shared-network-name ioithenko-stateless {
+        common-options {
+            domain-search ioithenko.net
+            name-server 2000::1
+        }
+        subnet 2000::0/64 {
+        }
+    }
+}
+router-advert {
+    interface eth1 {
+        other-config-flag
+        prefix 2000::/64 {
+        }
+    }
+}
[edit]
ioithenko@ioithenko-gw-01# commit
[edit]
ioithenko@ioithenko-gw-01# save
Saving configuration to '/config/config.boot'...
Done
[edit]
ioithenko@ioithenko-gw-01# run show configuration
interfaces {
    ethernet eth0 {
        address 10.0.0.1/24
        hw-id 0c:a9:fa:46:00:00
    }
    ethernet eth1 {
        address 2000::1/64
        hw-id 0c:a9:fa:46:00:01
    }
    ethernet eth2 {
        address 2001::1/64
        hw-id 0c:a9:fa:46:00:02
    }
    loopback lo {
    }
}
service {
    dhcp-server {
        shared-network-name ioithenko {
            domain-name ioithenko.net
            name-server 10.0.0.1
            subnet 10.0.0.0/24 {
                default-router 10.0.0.1
            }
        }
    }
}

```

Рис. 2.12: Настройка gw-01: DHCIPv6 без отслеживания состояния

Здесь с помощью указанных выше команд создана разделяемая сеть (*shared-network-name*) с названием *ioithenko-stateless*, задана информация общих опций (*common-options*) для разделяемой сети. При этом подсеть (*subnet*) $2000::/64$ не требуется настраивать, поскольку она не будет содержать полезной информации.

Проверяю настройки сети на PC2, пробую пропинговать маршрутизатор (успешно), проверяю настройки DNS (рис. 2.13 и 2.14).

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet6 2000::4ce7:1c37:fee7:bba9  prefixlen 64  scopeid 0x0
<global>
        inet6 fe80::bc7e:bd1d:3fd3:44d3  prefixlen 64  scopeid 0x2
  <link>
        ether 0c:65:9f:c1:00:00  txqueuelen 1000  (Ethernet)
          RX packets 12  bytes 1376 (1.3 KiB)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 40  bytes 4863 (4.7 KiB)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth1: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
        ether 0c:65:9f:c1:00:01  txqueuelen 1000  (Ethernet)
          RX packets 0  bytes 0 (0.0 B)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 0  bytes 0 (0.0 B)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth2: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
        ether 0c:65:9f:c1:00:02  txqueuelen 1000  (Ethernet)
          RX packets 0  bytes 0 (0.0 B)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 0  bytes 0 (0.0 B)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth3: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
        ether 0c:65:9f:c1:00:03  txqueuelen 1000  (Ethernet)
          RX packets 0  bytes 0 (0.0 B)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 0  bytes 0 (0.0 B)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth4: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
        ether 0c:65:9f:c1:00:04  txqueuelen 1000  (Ethernet)
```

Рис. 2.13: Проверка настроек сети на PC2, пинг маршрутизатора, проверка DNS

```
root@kali:~# route -n -A inet6
Kernel IPv6 routing table
Destination          Next Hop          Flag Met
Ref Use If
::1/128              ::                U     256
2        0 lo
2000::/64             ::                U     100
2        0 eth0
fe80::/64             ::                U     100
1        0 eth0
::/0                  fe80::ea9:faff:fe46:1   UG    100
1        0 eth0
::1/128              ::                UAn   0
4        0 lo
2000::4ce7:1c37:fee7:bba9/128  ::                UAn   0
3        0 eth0
fe80::bc7e:bd1d:3fd3:44d3/128  ::                UAn   0
3        0 eth0
ff00::/8               ::                U     256
3        0 eth0
::/0                  ::                !n    -1
1        0 lo
root@kali:~# ping 2000::1 -c 2
PING 2000::1(2000::1) 56 data bytes
64 bytes from 2000::1: icmp_seq=1 ttl=64 time=7.61 ms
64 bytes from 2000::1: icmp_seq=2 ttl=64 time=2.66 ms

--- 2000::1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 2.656/5.131/7.606/2.475 ms
root@kali:~# cat /etc/resolv.conf
# Generated by NetworkManager
search ioithenko.net
nameserver 2000::1
root@kali:~#
```

Рис. 2.14: Пинг маршрутизатора, проверка DNS

Получаю адрес по DHCPv6 с помощью dhclient -6 -S -v eth0. Вновь пингую маршрутизатор (успешно), проверяю настройки DNS (рис. 2.15).

```
nameserver 2000::1
root@kali:~# dhclient -6 -S -v eth0
Internet Systems Consortium DHCP Client 4.4.1
Copyright 2004-2018 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on Socket/eth0
Sending on  Socket/eth0
Created duid "\000\003\000\001\014e\237\301\000\000".
PRC: Requesting information (INIT).
XMT: Forming Info-Request, 0 ms elapsed.
XMT: Info-Request on eth0, interval 950ms.
RCV: Reply message on eth0 from fe80::ea9:faff:fe46:1.
PRC: Done.
root@kali:~# ping 2000::1 -c 2
PING 2000::1(2000::1) 56 data bytes
64 bytes from 2000::1: icmp_seq=1 ttl=64 time=3.32 ms
64 bytes from 2000::1: icmp_seq=2 ttl=64 time=4.16 ms

--- 2000::1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 3.323/3.741/4.160/0.418 ms
root@kali:~# cat /etc/resolv.conf
search ioithenko.net.
nameserver 2000::1
root@kali:~#
```

Рис. 2.15: Получение адреса на PC2, пинг маршрутизатора, проверка DNS

Использую команды для просмотра статистики DHCP-сервера и выданных адресов, но список пока пуст, хотя адрес был выдан (рис. 2.16). С маршрутизатора PC1 пингуется успешно.

```
ioithenko@ioithenko-gw-01# run show dhcpcv6 server leases
IPv6 address      State    Last communication      Lease expiration      Remaining      Type
Pool     IAID_DUID
-----  -----
[edit]
```

Рис. 2.16: Просмотр статистики DHCP, выданных адресов, пинг PC2

Просматриваю захваченные анализатором трафика пакеты (рис. 2.17 и 2.18).

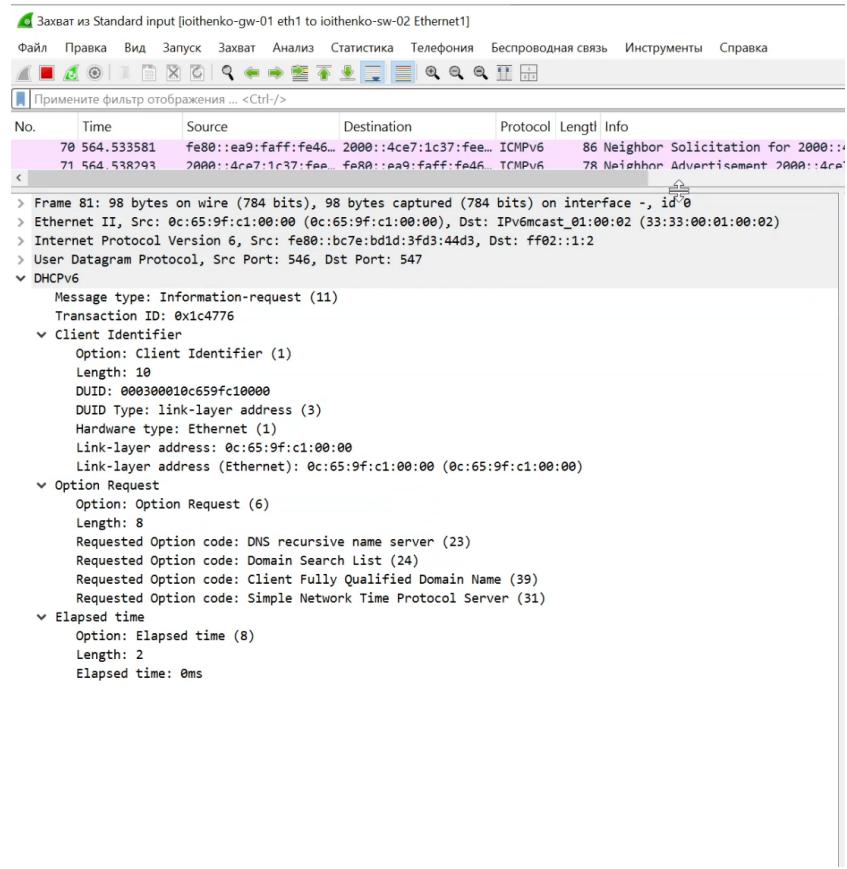


Рис. 2.17: Проверка захваченных анализатором трафика пакетов

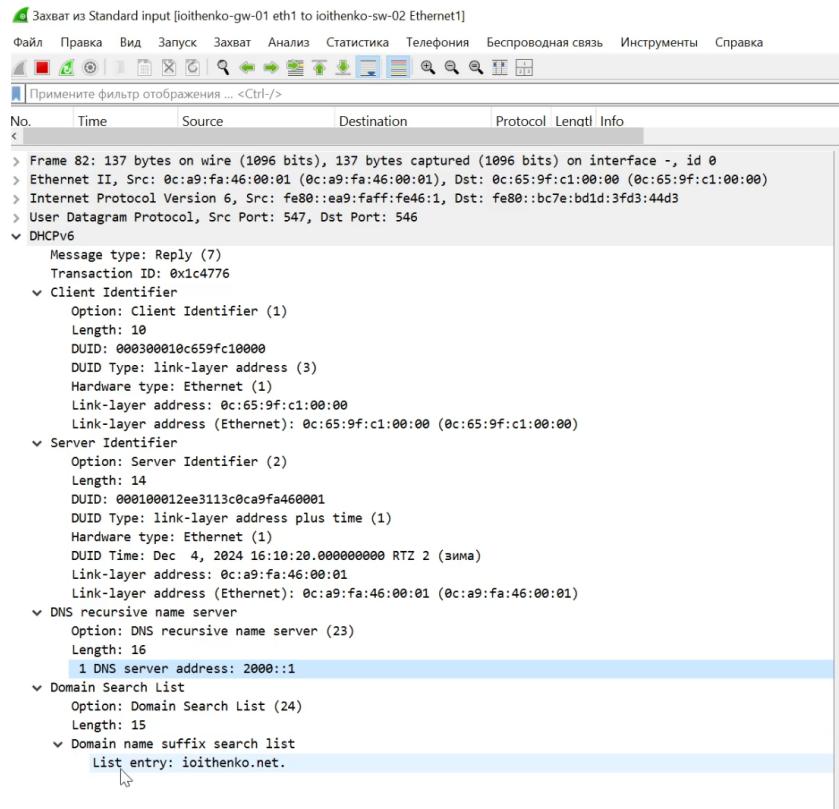


Рис. 2.18: Проверка захваченных анализатором трафика пакетов

Процесс получения адреса по DHCPv6 без отслеживания состояния происходит в 2 шага:

- INFORMATION-REQUEST: используется клиентом для запроса только параметров конфигурации (например, адреса DNS-сервера).
- REPLY: используется DHCPv6-сервером для отправки клиенту сетевых настроек и завершения обработки запроса.

На маршрутизаторе настраиваю DHCPv6 с отслеживанием состояния (DHCPv6 Stateful configuration) (рис. 2.19).

```
[edit]
[edit]nko@ioithenko-gw-01# set service router-advert interface eth2 managed-flag
[edit]nko@ioithenko-gw-01# set service dhcipv6-server shared-network-name ioithen
[edit]
[edit]nko@ioithenko-gw-01# set service dhcipv6-server shared-network-name ioithen
[edit]
[edit]nko@ioithenko-gw-01# set subnet 2001::0/64t service dhcipv6-server shared-network-name ioithen
[edit]
[edit]nko@ioithenko-gw-01# set subnet 2001::0/64 name-server 2001::1ver shared-network-name ioithen
[edit]
[edit]nko@ioithenko-gw-01# set subnet 2001::0/64 domain-search ioithenko.net
[edit]
[edit]nko@ioithenko-gw-01# set subnet 2001::0/64 address-range start 2001::100 stop 2001::199
[edit]
[edit]nko@ioithenko-gw-01# compare
[edit service dhcipv6-server]
+shared-network-name ioithenko-stateful {
+    subnet 2001::0/64 {
+        address-range {
+            start 2001::100 {
+                stop 2001::199
+            }
+        }
+        domain-search ioithenko.net
+        name-server 2001::1
+    }
+}
[edit service router-advert]
+interface eth2 {
+    managed-flag
+}
[edit]
[edit]nko@ioithenko-gw-01# commit
[edit]
[edit]nko@ioithenko-gw-01# save
Saving configuration to '/config/config.boot'...
Done
[edit]
[edit]nko@ioithenko-gw-01# run show dhcipv6 server leases
IPv6 address      State      Last communication      Lease expiration      Remaining      Type
Pool      IAID_DUID
-----  -----
----- [edit]
[edit]nko@ioithenko-gw-01#
```

Рис. 2.19: Настройка gw-01: DHCPv6 с отслеживанием состояния

Здесь при помощи указанных выше команд создана разделяемая сеть (shared-network-name) с названием `ioithenko-stateful`, подсеть (subnet) с адресом `2001::/64`, задан диапазон адресов (range) с именем `hosts`, содержащий адреса `2001::100 – 2001::199`.

На PC3 проверяю настройки сети и DNS (рис. 2.20 и 2.21).

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet6 2001::198  prefixlen 128  scopeid 0x0<global>
          inet6 fe80::badd:cfle:e376:3c2f  prefixlen 64  scopeid
          link>
              ether 0c:54:26:e1:00:00  txqueuelen 1000  (Ethernet)
              RX packets 13  bytes 1480 (1.4 KiB)
              RX errors 0  dropped 0  overruns 0  frame 0
              TX packets 231  bytes 35440 (34.6 KiB)
              TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth1: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
      ether 0c:54:26:e1:00:01  txqueuelen 1000  (Ethernet)
      RX packets 0  bytes 0 (0.0 B)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 0  bytes 0 (0.0 B)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth2: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
      ether 0c:54:26:e1:00:02  txqueuelen 1000  (Ethernet)
      RX packets 0  bytes 0 (0.0 B)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 0  bytes 0 (0.0 B)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

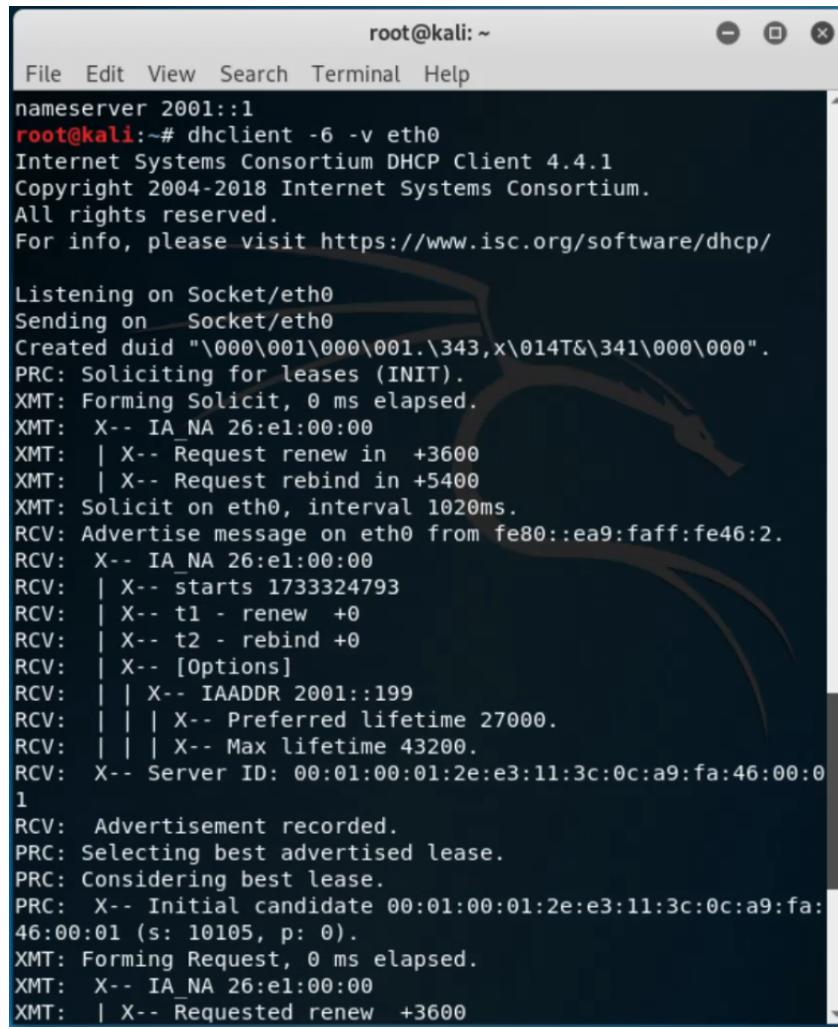
eth3: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
      ether 0c:54:26:e1:00:03  txqueuelen 1000  (Ethernet)
      RX packets 0  bytes 0 (0.0 B)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 0  bytes 0 (0.0 B)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

Рис. 2.20: Проверка настроек сети и DNS на PC3

```
root@kali:~# route -n -A inet6
Kernel IPv6 routing table
Destination          Next Hop           Fla
g Met Ref Use If
::1/128              ::                U
    256 2      0 lo
2001::198/128        ::                U
    100 1      0 eth0
fe80::/64             ::                U
    100 1      0 eth0
::/0                  fe80::ea9:faff:fe46:2   UG
    100 2      0 eth0
::1/128              ::                UAn
    0  4      0 lo
2001::198/128        ::                UAn
    0  3      0 eth0
fe80::badd:cf1e:e376:3c2f/128  ::                UAn
    0  3      0 eth0
ff00::/8              ::                U
    256 3      0 eth0
::/0                  ::                !n
    -1  1      0 lo
root@kali:~# cat /etc/resolv.conf
# Generated by NetworkManager
search ioithenko.net
nameserver 2001::1
root@kali:~# dhclient -6
```

Рис. 2.21: Проверка настроек сети и DNS на PC3

Получаю адрес по DHCPv6 (рис. 2.22).



The screenshot shows a terminal window titled "root@kali: ~" running the command "dhclient -6 -v eth0". The output displays the DHCP client's interaction with a nameserver at 2001::1. It shows the client sending a solicitation, receiving an advertisement, selecting a lease, and then sending a request. The terminal window has a dark background with a dragon logo, and the text is white or light gray.

```
nameserver 2001::1
root@kali:~# dhclient -6 -v eth0
Internet Systems Consortium DHCP Client 4.4.1
Copyright 2004-2018 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on Socket/eth0
Sending on Socket/eth0
Created duid "\000\001\000\001.\343,x\014T&\341\000\000".
PRC: Soliciting for leases (INIT).
XMT: Forming Solicit, 0 ms elapsed.
XMT: X-- IA_NA 26:e1:00:00
XMT: | X-- Request renew in +3600
XMT: | X-- Request rebind in +5400
XMT: Solicit on eth0, interval 1020ms.
RCV: Advertise message on eth0 from fe80::ea9:faff:fe46:2.
RCV: X-- IA_NA 26:e1:00:00
RCV: | X-- starts 1733324793
RCV: | X-- t1 - renew +0
RCV: | X-- t2 - rebind +0
RCV: | X-- [Options]
RCV: | | X-- IAADDR 2001::199
RCV: | | | X-- Preferred lifetime 27000.
RCV: | | | X-- Max lifetime 43200.
RCV: X-- Server ID: 00:01:00:01:2e:e3:11:3c:0c:a9:fa:46:00:0
1
RCV: Advertisement recorded.
PRC: Selecting best advertised lease.
PRC: Considering best lease.
PRC: X-- Initial candidate 00:01:00:01:2e:e3:11:3c:0c:a9:fa:
46:00:01 (s: 10105, p: 0).
XMT: Forming Request, 0 ms elapsed.
XMT: X-- IA_NA 26:e1:00:00
XMT: | X-- Requested renew +3600
```

Рис. 2.22: Получение адреса на PC3

Вновь на PC3 проверяю настройки сети и DNS, пингую маршрутизатор (успешно) (рис. 2.23, 2.24 и 2.25).

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet6 2001::198  prefixlen 128  scopeid 0x0<global>
      inet6 2001::199  prefixlen 128  scopeid 0x0<global>
      inet6 fe80::badd:cf1e:e376:3c2f  prefixlen 64  scopeid
          0x20<link>
          ether 0c:54:26:e1:00:00  txqueuelen 1000  (Ethernet)
          RX packets 17  bytes 2014 (1.9 KiB)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 241  bytes 36474 (35.6 KiB)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth1: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
      ether 0c:54:26:e1:00:01  txqueuelen 1000  (Ethernet)
      RX packets 0  bytes 0 (0.0 B)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 0  bytes 0 (0.0 B)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth2: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
      ether 0c:54:26:e1:00:02  txqueuelen 1000  (Ethernet)
      RX packets 0  bytes 0 (0.0 B)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 0  bytes 0 (0.0 B)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth3: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
      ether 0c:54:26:e1:00:03  txqueuelen 1000  (Ethernet)
      RX packets 0  bytes 0 (0.0 B)
```

Рис. 2.23: Проверка настроек сети на PC3

```
root@kali:~# route -n -A inet6
Kernel IPv6 routing table
Destination          Next Hop           Flagg
g Met Ref Use If
::1/128              ::                U
    256 2      0 lo
2001:::198/128       ::                U
        100 2      0 eth0
2001:::199/128       ::                U
        256 1      0 eth0
fe80:::/64            ::                U
        100 1      0 eth0
:::/0                 fe80:::ea9:faff:fe46:2   UG
        100 2      0 eth0
::1/128              ::                UAn
        0  4      0 lo
2001:::198/128       ::                UAn
        0  4      0 eth0
2001:::199/128       ::                UAn
        0  2      0 eth0
fe80:::badd:cfile:e376:3c2f/128  ::                UAn
        0  3      0 eth0
ff00:::/8             ::                U
        256 3      0 eth0
:::/0                 ::                !n
        -1  1      0 lo
```

Рис. 2.24: Проверка настроек сети на PC3

```
root@kali:~# ping 2001::1 -c 2
PING 2001::1(2001::1) 56 data bytes
64 bytes from 2001::1: icmp_seq=1 ttl=64 time=12.1 ms
64 bytes from 2001::1: icmp_seq=2 ttl=64 time=2.13 ms

--- 2001::1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 2.126/7.093/12.060/4.967 ms
root@kali:~# cat /etc/resolv.conf
search ioithenko.net.
nameserver 2001::1
root@kali:~#
```

Рис. 2.25: Пинг маршрутизатора

Использую команды для просмотра статистики DHCP-сервера и выданных адресов на маршрутизаторе и убеждаюсь, что адрес был выдан (рис. 2.26).

```

[edit]
ioithenko@ioithenko-gw-01# run show dhcpv6 server leases
IPv6 address      State      Last communication      Lease expiration      Remaining      Type
    Pool           IAID_DUID
-----
2001::198      active     2024/12/04 14:55:56   2024/12/04 17:00:56  1:52:36      non-temporary
mportary ioithenko-stateful 00:00:e1:26:00:04:f3:31:a8:fe:2f:23:6e:06:f8:53:14:35:b8:b
b:d5:61
2001::199      active     2024/12/04 15:06:33   2024/12/04 17:11:33  2:03:13      non-temporary
mportary ioithenko-stateful 00:00:e1:26:00:01:00:01:2e:e3:2c:78:0c:54:26:e1:00:00
[edit]
ioithenko@ioithenko-gw-01#

```

Рис. 2.26: Просмотр статистики DHCP и выданных адресов

Просматриваю захваченные анализатором трафика пакеты (рис. 2.27).

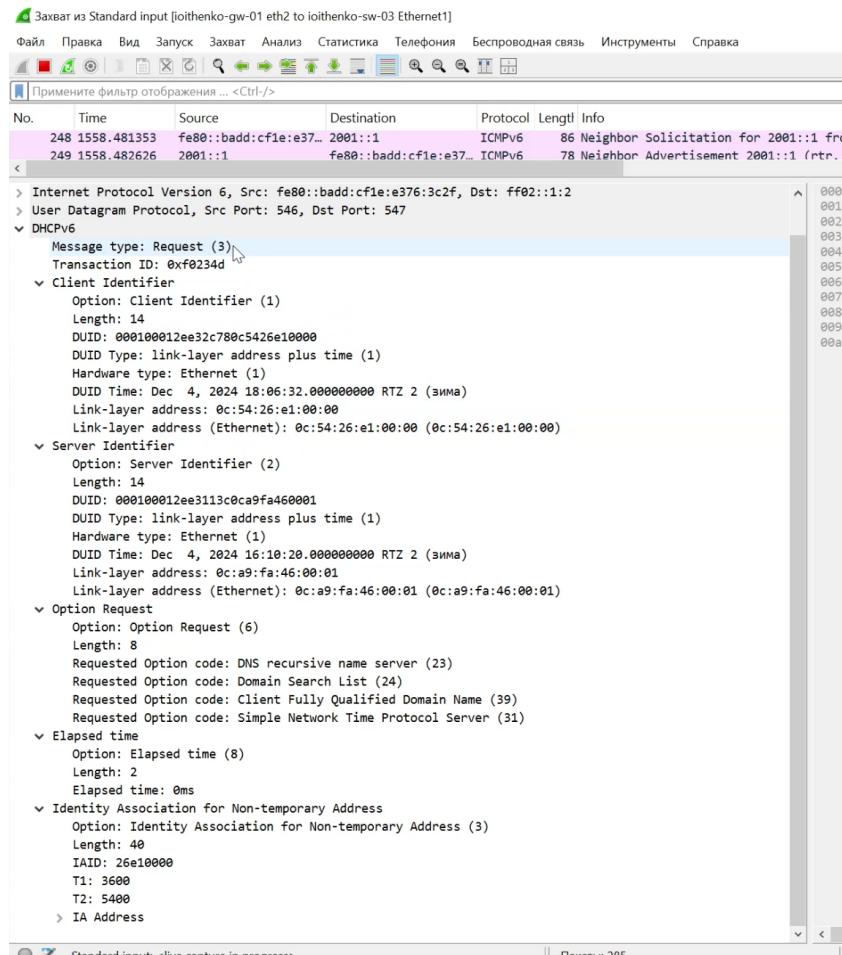


Рис. 2.27: Проверка захваченных анализатором трафика пакетов

Процесс получения устройством адреса по DHCPv6 с отслеживанием состояния происходит в 4 шага:

- SOLICIT: устройство направляет на зарезервированный IPv6-адрес много-

адресной рассылки FF02::1:2 широковещательный запрос;

- ADVERTISE: DHCPv6-сервер сообщает DHCPv6-клиенту, что сервер доступен для предоставления службы DHCPv6;
- REQUEST: используется клиентом для запроса IPv6-адреса и всех остальных параметров конфигурации от сервера в случае, когда DHCPv6-сервер работает с сохранением состояния;
- REPLY: используется DHCPv6-сервером для отправки клиенту сетевых настроек и завершения обработки запроса.

3 Выводы

В ходе выполнения лабораторной работы я получила навыки настройки службы DHCP на сетевом оборудовании для распределения адресов IPv4 и IPv6.