

Администрирование сетевых подсистем

Лабораторная работа №3

Тойчубекова Асель Нурлановна

Содержание

1	Цель работы	6
2	Теоретическое введение	7
3	Задание	9
4	Выполнение лабораторной работы	10
5	Выводы	27
6	Список литературы	28

Список иллюстраций

4.1	Запуск server	10
4.2	Установка dhcp	10
4.3	Сохранение конф файла	10
4.4	Редактирование kea-dhcp4.conf	11
4.5	Редактирование kea-dhcp4.conf	11
4.6	Редактирование kea-dhcp4.conf	11
4.7	Редактирование kea-dhcp4.conf	12
4.8	Настройка привязки dhcp	12
4.9	Проверка работы конф файла	12
4.10	Перезагрузка конф dhcpd	13
4.11	Настройка DHCP-сервера	13
4.12	Настройка DHCP-сервера	13
4.13	Перезапуск named и проверка обращения к DHCP	14
4.14	Настройка межсетевого узла server	14
4.15	Настройка межсетевого узла server	14
4.16	Восстановление контекста безопасности	15
4.17	Мониторинг процессов в системе	15
4.18	Запуск DHCP	15
4.19	Файл 01-routing.sh.	16
4.20	Редактирование Vagrantfile	16
4.21	Запуск clien	17
4.22	Информация о работе DHCP	17
4.23	Информация об имеющихся интерфейсах	18
4.24	Информация о работе DHCP	19
4.25	Создание ключа на сервере с Bind9	19
4.26	Ключ на сервере с Bind9	20
4.27	Изменение прав доступа	20
4.28	Подключение ключ	20
4.29	Разрешение обновление зоны	21
4.30	Проверка конф файла и перезапуск DNS	21
4.31	Формирование ключа для Kea	21
4.32	Редактирование владельца и прав доступа	21
4.33	Настройка DHCP,DNS	22
4.34	Проверка на наличие синтаксических ошибок	22
4.35	Запуск службы ddns и проверка статуса работы	22
4.36	Редактирование конф файла kea-dhcp4.conf	23

4.37	Проверка файла на ошибки	23
4.38	Запуск DHCP и проверка статуса	23
4.39	Получение адреса клиентом	23
4.40	Файл antoychubekova.net.jnl	24
4.41	DNS-записи	24
4.42	Создание каталога и подкаталогов	25
4.43	Редактирование конфигурационных файлов DNS-сервера	25
4.44	Создание исполняемого файла	25
4.45	Редактирование исполняемого файла	26
4.46	Редактирование Vagrantfile	26

Список таблиц

1 Цель работы

Целью данной лабораторной работы является приобретение практических навыков по установке и конфигурированию DHCP-сервера.

2 Теоретическое введение

В современных компьютерных сетях важнейшим элементом является автоматизация процессов настройки сетевых параметров. Одним из ключевых инструментов, обеспечивающих данную задачу, выступает протокол DHCP (Dynamic Host Configuration Protocol). Он реализует модель взаимодействия «клиент–сервер» и позволяет автоматизировать процесс присвоения IP-адресов и сопутствующих параметров рабочим станциям и другим узлам сети.

Использование DHCP избавляет администратора от необходимости ручного конфигурирования сетевых устройств, что значительно снижает вероятность ошибок и облегчает сопровождение сети. Сервер DHCP динамически распределяет IP-адреса из заданного диапазона, контролируемого администратором, и обеспечивает передачу клиентам дополнительных параметров, таких как маска подсети, адрес шлюза по умолчанию, DNS-серверы и имя домена. Взаимодействие клиентов и сервера осуществляется по протоколу UDP: сервер принимает запросы на порту 67 и отправляет ответы клиентам на порт 68.

Для корректного функционирования сети также важен регламент распределения адресов, который позволяет чётко разделить диапазоны IP-адресов по назначению: для шлюзов, сетевого оборудования, серверов, клиентских машин с динамической и статической конфигурацией, а также для принтеров и резервных узлов. Такая схема обеспечивает удобство управления и предотвращает конфликты адресов.

Важной частью работы администратора является диагностика и контроль работы DHCP. Для этого применяются сетевые утилиты:

- `ifconfig` — инструмент для настройки и просмотра параметров сетевых интерфейсов, позволяющий назначать IP-адреса и проверять их текущее состояние.
- `ping` — средство проверки доступности сетевых узлов и оценки качества соединения. Оно позволяет измерять время отклика (RTT) и выявлять потери пакетов, что помогает в диагностике перегрузок каналов связи или неисправностей маршрутизаторов.

3 Задание

1. Установите на виртуальной машине server DHCP-сервер.
2. Настройте виртуальную машину server в качестве DHCP-сервера для виртуальной внутренней сети .
3. Проверьте корректность работы DHCP-сервера в виртуальной внутренней сети путём запуска виртуальной машины client и применения соответствующих утилит диагностики.
4. Настройте обновление DNS-зоны при появлении в виртуальной внутренней сети новых узлов.
5. Проверьте корректность работы DHCP-сервера и обновления DNS-зоны в виртуальной внутренней сети путём запуска виртуальной машины client и применения соответствующих утилит диагностики.
6. Напишите скрипт для Vagrant, фиксирующий действия по установке и настройке DHCP-сервера во внутреннем окружении виртуальной машины server. Соответствующим образом внести изменения в Vagrantfile.

4 Выполнение лабораторной работы

Для начала лабораторной работы запустим vm server. (рис. 4.1).

```
C:\work1\atoychubekova\vagrant>vagrant up server
Vagrant failed to initialize at a very early stage:

There was an error loading a Vagrantfile. The file being lo
and the error message are shown below. This is usually caus
an invalid or undefined variable.

Path: C:/work1/atoychubekova/vagrant/Vagrantfile
Line number: 0
Message: undefined local variable or method `server'

C:\work1\atoychubekova\vagrant>
1Help 2UserMn 3View 4Edit 5Copy 6RenMov 7MkFo
```

Рисунок 4.1: Запуск server

На vm перейдем в режим суперпользователя и установим dhcp. (рис. 4.2).

```
[antoychubekova@server.antoychubekova.net ~]$ sudo -i
[sudo] password for antoychubekova:
[root@server.antoychubekova.net ~]# dnf -y install kea
Extra Packages for Enterprise [====] 100% 0 B/s | 0 B | ETA
```

Рисунок 4.2: Установка dhcp

Сохраним на всякий случай конфигурационный файл. (рис. 4.3).

```
[root@server.antoychubekova.net ~]# cp /etc/kea/kea-dhcp4.conf /etc/kea/kea-dhcp4.conf_$(date +%Y%m%d%H%M%S)
[root@server.antoychubekova.net ~]#
```

Рисунок 4.3: Сохранение конф файла

Откроем файл /etc/kea/kea-dhcp4.conf на редактирование. В этом файле заменим шаблон для domain-name. (рис. 4.4 и рис. 4.5 и рис. 4.6).

```
// "name": "domain-name" or "code": 15.
{
    "code": 15,
    "data": "antoychubekova.net"
},

// Domain search for domain entries.
```

Рисунок 4.4: Редактирование kea-dhcp4.conf

```
// example, name foo would be attempted to
// foo.mydomain.example.com and if it fails,
{
    "name": "domain-search",
    "data": "antoychubekova.net"
},

// String options that have a comma in their
```

Рисунок 4.5: Редактирование kea-dhcp4.conf

```
// but it's a lot of writing, so it's easier
{
    "name": "domain-name-servers",
    "data": "192.168.1.1"
},
```

Рисунок 4.6: Редактирование kea-dhcp4.conf

На базе одного из приведённых в файле примеров конфигурирования подсети зададим собственную конфигурацию dhcp-сети, задав адрес подсети, диапазон адресов для распределения клиентам, адрес маршрутизатора и broadcast-адреса. (рис. 4.7).

```

"subnet4": [
  {
    "id": 1,
    // specify subnet that DHCP is used
    "subnet": "192.168.1.0/24",
    // specify the range of IP addresses to be leased
    "pools": [ { "pool": "192.168.1.30 - 192.168.1.199" } ],
    "option-data": [
      {
        // specify your gateway
        "name": "routers",
        "data": "192.168.1.1"
      }
    ]
  }
]
},

```

Рисунок 4.7: Редактирование kea-dhcp4.conf

Настроим привязку dhcpd к интерфейсу eth1 виртуальной машины server. (рис. 4.8).

```

"Dhcp4": {
  // Add names of your network interfaces to listen on.
  "interfaces-config": {
    "interfaces": [ "eth1" ]
  }
  // Kea DHCPv4 server by default listens using raw sockets. This ensures
  // all sockets, including those sent by directly connected clients

```

Рисунок 4.8: Настройка привязки dhcp

Проверим правильность конфигурационного файла. Мы видим, что все корректно отрабатывается. (рис. 4.9).

```

[root@server.antoychubekova.net ~]# nano /etc/kea/kea-dhcp4.conf
[root@server.antoychubekova.net ~]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-09-18 14:29:48.522 INFO [kea-dhcp4.hosts/84401.140587266328768] HOSTS_BA
CKENDS_REGISTERED the following host backend types are available: mysql postgr
esql
2025-09-18 14:29:48.544 WARN [kea-dhcp4.dhcpsrv/84401.140587266328768] DHCP
SRV_V_MT_DISABLED_QUEUE_CONTROL disabling dhcp queue control when multi-thre
ading is enabled.
2025-09-18 14:29:48.546 WARN [kea-dhcp4.dhcp4/84401.140587266328768] DHCP4_R
ESERVATIONS_LOOKUP_FIRST_ENABLED Multi-threading is enabled and host reservat
ions lookup is always performed first.
2025-09-18 14:29:48.549 INFO [kea-dhcp4.dhcpsrv/84401.140587266328768] DHCP
SRV_V_CFGMGR_NEW_SUBNET4 a new subnet has been added to configuration: 192.168.1.0
/24 with params: t1=900, t2=1800, valid-lifetime=3600
2025-09-18 14:29:48.551 INFO [kea-dhcp4.dhcpsrv/84401.140587266328768] DHCP
SRV_V_CFGMGR_SOCKET_TYPE_SELECT using socket type raw
2025-09-18 14:29:48.553 INFO [kea-dhcp4.dhcpsrv/84401.140587266328768] DHCP
SRV_V_CFGMGR_ADD_IFACE listening on interface eth1
2025-09-18 14:29:48.556 INFO [kea-dhcp4.dhcpsrv/84401.140587266328768] DHCP
SRV_V_CFGMGR_SOCKET_TYPE_DEFAULT "dhcp-socket-type" not specified , using default
socket type raw
[root@server.antoychubekova.net ~]#

```

Рисунок 4.9: Проверка работы конф файла

Перезагрузим конфигурацию `dhcpcd` и разрешим загрузку DHCP-сервера при запуске виртуальной машины `server`. (рис. 4.10).

```
[root@server.antoychubekova.net ~]# systemctl --system daemon-reload
[root@server.antoychubekova.net ~]# systemctl enable kea-dhcp4.service
Created symlink '/etc/systemd/system/multi-user.target.wants/kea-dhcp4.service'
→ '/usr/lib/systemd/system/kea-dhcp4.service'.
[root@server.antoychubekova.net ~]#
```

Рисунок 4.10: Перезагрузка конф `dhcpcd`

Добавим запись для DHCP-сервера в конце файла прямой DNS-зоны `/var/named/master/fz/antoychubekova.net`. (рис. 4.11).

```
GNU nano 8.1 /var/named/master/fz/antoychubekova.net Modified
$TTL 1D
@      IN SOA  @ server.antoychubekova.net. (
                                2025091800    ; serial
                                1D      ; refresh
                                1H      ; retry
                                1W      ; expire
                                3H )    ; minimum

      NS      @
      A      192.168.1.1
$ORIGIN antoychubekova.net.
server      A      192.168.1.1
ns          A      192.168.1.1
dhcp        A      192.168.1.1
```

Рисунок 4.11: Настройка DHCP-сервера

И в конце файла обратной зоны `/var/named/master/rz/192.168.1`. При этом не забудем в обоих файлах изменить серийный номер файла зоны, указав текущую дату в нотации ГГГГММДДВВ. (рис. 4.12).

```
GNU nano 8.1 /var/named/master/rz/192.168.1 Modified
$TTL 1D
@      IN SOA  @ server.antoychubekova.net. (
                                2025091800    ; serial
                                1D      ; refresh
                                1H      ; retry
                                1W      ; expire
                                3H )    ; minimum

      NS      @
      A      192.168.1.1
      PTR     server.antoychubekova.net.
$ORIGIN 1.168.192.in-addr.arpa.
1      PTR     server.antoychubekova.net.
1      PTR     ns.antoychubekova.net.
1      PTR     dhcp.antoychubekova.net.
```

Рисунок 4.12: Настройка DHCP-сервера

Перезапустим named и проверим, что можно обратиться к DHCP-серверу по имени. Мы видим, что все правильно работает и мы можем обратиться к DHCP-серверу. (рис. 4.13).

```
[root@server.antoychubekova.net ~]# systemctl restart named
[root@server.antoychubekova.net ~]# ping dhcp.antoychubekova.net
PING dhcp.antoychubekova.net (192.168.1.1) 56(84) bytes of data.
64 bytes from server.antoychubekova.net (192.168.1.1): icmp_seq=1 ttl=64 time=
1.26 ms
64 bytes from server.antoychubekova.net (192.168.1.1): icmp_seq=2 ttl=64 time=
0.112 ms
64 bytes from server.antoychubekova.net (192.168.1.1): icmp_seq=3 ttl=64 time=
1.73 ms
64 bytes from server.antoychubekova.net (192.168.1.1): icmp_seq=4 ttl=64 time=
0.082 ms
64 bytes from server.antoychubekova.net (192.168.1.1): icmp_seq=5 ttl=64 time=
0.147 ms
64 bytes from server.antoychubekova.net (192.168.1.1): icmp_seq=6 ttl=64 time=
0.098 ms
```

Рисунок 4.13: Перезапуск named и проверка обращения к DHCP

Внесив изменения в настройки межсетевого экрана узла server, разрешив работу с DHCP. (рис. 4.14 и рис. 4.15).

```
[root@server.antoychubekova.net ~]# firewall-cmd --list-services
cockpit dhcpv6-client dns ssh
[root@server.antoychubekova.net ~]# firewall-cmd --get-services
0-AD RH-Satellite-6 RH-Satellite-6-capsule afp alvr amanda-client amanda-k5-cl
ient amqp amqps anno-1602 anno-1800 apcupsd aseqnet audit ausweisapp2 bacula b
acula-client bareos-director bareos-filedaemon bareos-storage bb bgp bitcoin b
itcoin-rpc bitcoin-testnet bitcoin-testnet-rpc bittorrent-lsd ceph ceph-export
er ceph-mon cfengine checkmk-agent civilization-iv civilization-v cockpit coll
ectd condor-collector cratedb ctdb dds dds-multicast dds-unicast dhcp dhcpv6 d
```

Рисунок 4.14: Настройка межсетевого узла server

```
[root@server.antoychubekova.net ~]# firewall-cmd --add-service=dhcp
success
[root@server.antoychubekova.net ~]# firewall-cmd --add-service=dhcp --permanent
bash: firewall-cmd: command not found...
[root@server.antoychubekova.net ~]# firewall-cmd --add-service=dhcp --permanen
t
success
[root@server.antoychubekova.net ~]#
```

Рисунок 4.15: Настройка межсетевого узла server

Восстановим контекст безопасности в SELinux. (рис. 4.16).

```
[root@server.antoychubekova.net ~]# restorecon -vR /etc
Relabeled /etc/NetworkManager/system-connections/eth1.nmconnection from unconf
ined_u:object_r:user_tmp_t:s0 to unconfined_u:object_r:NetworkManager_etc_rw_t
:s0
[root@server.antoychubekova.net ~]# restorecon -vR /var/named
[root@server.antoychubekova.net ~]# restorecon -vR /var/lib/kea/
[root@server.antoychubekova.net ~]#
```

Рисунок 4.16: Восстановление контекста безопасности

В дополнительном терминале запустим мониторинг происходящих в системе процессов в реальном времени. (рис. 4.17).

```
0)#012#3 0x0000000000416559 n/a (n/a + 0x0)#012#4 0x00000000004182da n/a (n/
a + 0x0)#012#5 0x0000000000417d6a n/a (n/a + 0x0)#012#6 0x0000000000404860 n
/a (n/a + 0x0)#012#7 0x0000000000450b9c n/a (n/a + 0x0)#012#8 0x000000000043
59a0 n/a (n/a + 0x0)#012#9 0x000007f9aa3001b68 start_thread (libc.so.6 + 0x94b
68)#012#10 0x000007f9aa30726bc __clone3 (libc.so.6 + 0x1056bc)#012#012Stack tra
ce of thread 87337:#012#0 0x000007f9aa30704bd syscall (libc.so.6 + 0x1034bd)#0
12#1 0x00000000004348b2 n/a (n/a + 0x0)#012#2 0x00000000004507e6 n/a (n/a +
0x0)#012#3 0x0000000000405123 n/a (n/a + 0x0)#012#4 0x000007f9aa2f9730e __lib
c_start_call_main (libc.so.6 + 0x2a30e)#012#5 0x000007f9aa2f973c9 __libc_start
_main@@GLIBC_2.34 (libc.so.6 + 0x2a3c9)#012#6 0x00000000004044aa n/a (n/a + 0
x0)#012ELF object binary architecture: AMD x86-64
Sep 18 14:46:24 server systemd[1]: systemd-coredump@6586-87341-0.service: Deac
tivated successfully.
Sep 18 14:46:26 server ptaxis[82867]: context mismatch in svga_surface_destroy
Sep 18 14:46:26 server ptaxis[82867]: context mismatch in svga_surface_destroy
Sep 18 14:46:26 server ptaxis[82867]: context mismatch in svga_surface_destroy
```

Рисунок 4.17: Мониторинг процессов в системе

В основном рабочем терминале запустим DHCP-сервер. (рис. 4.18).

```
[root@server.antoychubekova.net ~]# systemctl start kea-dhcp4.service
[root@server.antoychubekova.net ~]#
```

Рисунок 4.18: Запуск DHCP

Запуск DHCP-сервера прошёл успешно, далее не выключая виртуальной машины server и не прерывая на ней мониторинга происходящих в системе процессов, приступим к анализу работы DHCP-сервера на клиенте. Перед запуском виртуальной машины client в каталоге с проектом в вашей операционной системе в подкаталоге vagrant/provision/client создадим файл 01-routing.sh. (рис. 4.19).

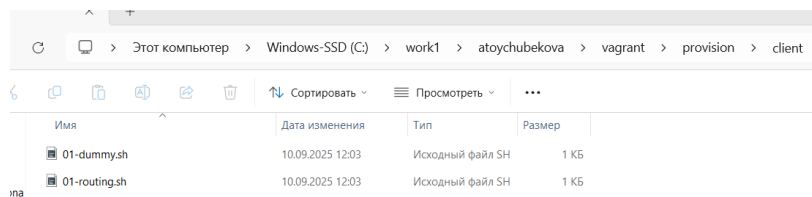


Рисунок 4.19: Файл 01-routing.sh.

Этот скрипт изменяет настройки NetworkManager так, чтобы весь трафик на виртуальной машине client шёл по умолчанию через интерфейс eth1.

В Vagrantfile подключим этот скрипт в разделе конфигурации для клиента:

- client.vm.provision «client routing»,
- type: «shell»,
- preserve_order: true,
- run: «always»,
- path: «provision/client/01-routing.sh» (рис. 4.20).

```

client.vm.provision "client routing",
  type: "shell",
  preserve_order: true,
  run: "always",
  path: "provision/client/01-routing.sh"
end
end

```

Рисунок 4.20: Редактирование Vagrantfile

Зафиксируем внесённые изменения для внутренних настроек виртуальной машины client и запустите её. (рис. 4.21).


```

C:\work1\atoychubekova\vagrant>vagrant up client --provision
Bringing machine 'client' up with 'virtualbox' provider...
==> client: Clearing any previously set forwarded ports...
==> client: Fixed port collision for 22 => 2222. Now on port 2200.
==> client: Clearing any previously set network interfaces...
==> client: Preparing network interfaces based on configuration...
client: Adapter 1: nat
client: Adapter 2: intnet
==> client: Forwarding ports...
client: 22 (guest) => 2200 (host) (adapter 1)
==> client: Running 'pre-boot' VM customizations...
==> client: Booting VM...
==> client: Waiting for machine to boot. This may take a few minutes...
client: SSH address: 127.0.0.1:2200
client: SSH username: vagrant
client: SSH auth method: password
==> client: Machine booted and ready!
[client] GuestAdditions 7.2.0 running --- OK.
==> client: Checking for guest additions in VM...
==> client: Setting hostname...

```

Рисунок 4.21: Запуск client

После загрузки виртуальной машины client мы видим информацию о работе DHCP-сервера в файле /var/lib/kea/kea-leases4.csv. (рис. 4.22).

```

GNU nano 8.1 kea-leases4.csv
address,hwaddr,client_id,valid_lifetime,expire,subnet_id,fqdn_fwd,fqdn_rev,hostname,state,user_context,pool_id
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259100,1,1,1,client.eavarnikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259710,1,1,1,client.eavarnikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259761,1,1,1,client.eavarnikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259787,1,1,1,client.eavarnikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259787,1,0,0,2,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,0,1758256187,1,0,0,2,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758308274,1,1,1,client.eavarnikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758308302,1,1,1,client.eavarnikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758308311,1,1,1,client.eavarnikovskaya.net,0,,0

```

Рисунок 4.22: Информация о работе DHCP

Тут

address — выданный клиенту IP-адрес.

hwaddr — MAC-адрес клиента (аппаратный адрес сетевой карты).

client_id — идентификатор клиента DHCP (часто включает MAC).

valid_lifetime — время жизни аренды (в секундах).

expire — время истечения аренды (в формате Unix timestamp).

subnet_id — ID подсети, к которой относится клиент.

fqdn_fwd — был ли выполнен прямой DNS-апдейт (A-запись).

fqdn_rev — был ли выполнен обратный DNS-апдейт (PTR-запись).

hostname — имя хоста клиента.

state — состояние аренды (0 = свободна, 1 = активна и т.д.).

user_context — доп. данные (если настроено).

pool_id — идентификатор пула, из которого выделен адрес.

Пример:

192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259100,1,1,1,client.eavernikovskaya.net,0,0

Клиент с MAC 08:00:27:24:9e:2b получил IP 192.168.1.30.

DHCP Client ID совпадает с MAC.

Время жизни аренды — 3600 секунд (1 час).

Срок истекает в 1758259100.

Подсеть ID = 1.

DNS обновления: прямое (A) и обратное (PTR) выполнены (1,1).

Имя хоста: client.eavernikovskaya.net.

State = 0 → аренда не активна (возможно, освобождена).

Войдем в систему виртуальной машины client под пользователем и откроем терминал. Используя команду `ifconfig` выведем на экран информацию об имеющихся интерфейсах. (рис. 4.23).

```
[antoychubekova@client.antoychubekova.net ~]$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fd17:625c:f037:2:a00:27ff:feaa:ce23 prefixlen 64 scopeid 0x0<global>
    inet6 fe80::a00:27ff:feaa:ce23 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:aa:ce:23 txqueuelen 1000 (Ethernet)
    RX packets 1494 bytes 176367 (172.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1284 bytes 217726 (212.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.30 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::4700:5e3:4c0e:232d prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:24:9e:2b txqueuelen 1000 (Ethernet)
    RX packets 254 bytes 34036 (33.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 649 bytes 56537 (55.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 18 bytes 2118 (2.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 18 bytes 2118 (2.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[antoychubekova@client.antoychubekova.net ~]$
```

Рисунок 4.23: Информация об имеющихся интерфейсах

Интерфейс eth0 – это первая сетевая карта. У неё IPv4-адрес 10.0.2.15 с маской 255.255.255.0 и широковещательным адресом 10.0.2.255. Также есть два IPv6-адреса: один глобальный (fd17:625c:f037:2::...) и один локальный (fe80::...). MAC-адрес карты – 08:00:27:aa:ce:23. Интерфейс активен, принимает и отправляет пакеты без ошибок. Этот интерфейс обычно используется для выхода в интернет через NAT.

Интерфейс eth1 – это вторая сетевая карта. У неё IPv4-адрес 192.168.1.30 с маской 255.255.255.0 и широковещательным адресом 192.168.1.255. IPv6-адрес локальный (fe80::4700:5e3:4c0e:232d). MAC-адрес – 08:00:27:24:9e:2b. Интерфейс также активен, пакеты передаются и принимаются без ошибок. Этот интерфейс используется для связи внутри локальной сети.

Интерфейс lo (loopback) – это виртуальный интерфейс, который нужен для работы самой системы с собой. Его IPv4-адрес – 127.0.0.1 (localhost), IPv6-адрес – ::1. Через него приложения могут обращаться к серверу, минуя сеть.

На машине server посмотрим список выданных адресов. (рис. 4.24).

```
[root@server.antoichubekova.net kea]# cat /var/lib/kea/kea-leases4.csv
address,hwaddr,client_id,valid_lifetime,expire,subnet_id,fqdn_fwd,fqdn_rev,hostname,state,user_context,pool_id
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259100,1,1,1,client.eavernikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259710,1,1,1,client.eavernikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259761,1,1,1,client.eavernikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259787,1,1,1,client.eavernikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758259787,1,0,0,2,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,0,1758256187,1,0,0,2,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758308274,1,1,1,client.eavernikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758308302,1,1,1,client.eavernikovskaya.net,0,,0
192.168.1.30,08:00:27:24:9e:2b,01:08:00:27:24:9e:2b,3600,1758308311,1,1,1,client.eavernikovskaya.net,0,,0
[root@server.antoichubekova.net kea]#
```

Рисунок 4.24: Информация о работе DHCP

Требуется настроить обновление DNS-зоны при появлении в виртуальной внутренней сети новых узлов. Создадим ключ на сервере с Bind9. (рис. 4.25).

```
[root@server.antoichubekova.net kea]# mkdir -p /etc/named/keys
[root@server.antoichubekova.net kea]# tsig-keygen -a HMAC-SHA512 DHCP_UPDATER > /etc/named/keys/dhcp_updater.key
[root@server.antoichubekova.net kea]#
```

Рисунок 4.25: Создание ключа на сервере с Bind9

Файл /etc/named/keys/dhcp_updater.key выглядит следующим образом. (рис. 4.26).

```
GNU nano 8.1 /etc/named/keys/dhcp_updater.key
key "DHCP_UPDATER" {
    algorithm hmac-sha512;
    secret "Jndya+exf5H39ZjnDZTcugKoiRJBZQp0wu83rcjpcVNrphRJx/3HkGBADm9GF8D5u5cVmfgd7VmspJdk/hD6Mg==";
};
```

Рисунок 4.26: Ключ на сервере с Bind9

Поправим права доступа. (рис. 4.27).

```
[root@server.antoychubekova.net kea]# chown -R named:named /etc/named/keys
[root@server.antoychubekova.net kea]# "
```

Рисунок 4.27: Изменение прав доступа

Подключим ключ в файле /etc/named.conf. (рис. 4.28).

```
GNU nano 8.1 /etc/named.conf

    file "data/named.run";
    severity dynamic;
};

zone "." IN {
    type hint;
    file "named.ca";
};

include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
include "/etc/named/antoychubekova.net";
include "/etc/named/keys/dhcp_updater.key"
```

^G Help ^O Write Out ^F Where Is ^K Cut ^T Execute
 ^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify

Рисунок 4.28: Подключение ключ

На виртуальной машине server под пользователем с правами суперпользователя отредактируем файл /etc/named/antoychubekova.net, разрешив обновление зоны. (рис. 4.29).

```
GNU nano 8.1 /etc/named/antoychubekova.net

zone "antoychubekova.net" IN {
    type master;
    file "master/fz/antoychubekova.net";
    update-policy {
        grant DHCP_UPDATER wildcard *.antoychubekova.net A DHCID;
    };
};

zone "1.168.192.in-addr.arpa" IN {
    type master;
    file "master/rz/192.168.1";
    update-policy {
        grant DHCP_UPDATER wildcard *.1.168.192.in-addr.arpa PTR DHCID;
    };
};

^G Help      ^O Write Out ^F Where Is  ^K Cut       ^T Execute   ^C Location  ^H-U Undo
^X Exit      ^R Read File ^A Replace   ^U Paste     ^J Justify   ^_ Go To Line ^M-E Redo
```

Рисунок 4.29: Разрешение обновление зоны

Сделаем проверку конфигурационного файла и перезапустим DNS-сервер. (рис. 4.30).

```
[root@server.antoychubekova.net kea]# nano /etc/named/antoychubekova.net
[root@server.antoychubekova.net kea]# named-checkconf
[root@server.antoychubekova.net kea]# systemctl restart named
```

Рисунок 4.30: Проверка конф файла и перезапуск DNS

Сформируем ключ для Кеа. Файл ключа назовём /etc/kea/tsig-keys.json. Перенесём ключ на сервер Кеа DHCP(с файла /etc/named/keys/dhcp_updater.key) и перепишем его в формате json.(рис. 4.31).

```
"tsip-keys": [
{
  "name": "DHCP_UPDATER",
  "algorithm": "hmac-sha512",
  "secret": "unyN3RUxStZczhN5Tyqe6kIDa6t/HiFF8X2nN8QzuJRKATmBMxJG1T06QvHTLmP20Aq0vNa6vtLmXy1Enw6g=="
}
]
```

Рисунок 4.31: Формирование ключа для Кеа

Сменим владельца и поправим права доступа (рис. 4.32).

```
[root@server.antoychubekova.net kea]# chown kea:kea /etc/kea/tsip-keys.json
[root@server.antoychubekova.net kea]# chmod 640 /etc/kea/tsip-keys.json
[root@server.antoychubekova.net kea]#
```

Рисунок 4.32: Редактирование владельца и прав доступа

Настройка происходит в файле /etc/kea/kea-dhcp-ddns.conf. (рис. 4.33).

```

GNU nano 8.1 /etc/kea/kea-dhcp-ddns.conf
// See Section 11 for examples and details description.
"DhcpDdns":
{
  "ip-address": "127.0.0.1",
  "port": 53001,
  "control-socket": {
    "socket-type": "unix",
    "socket-name": "/run/kea/kea-ddns-ctrl-socket"
  },
  <?include "/etc/kea/tsig-keys.json"?>
  "forward-ddns": {
    "ddns-domains": [
      {
        "name": "antoychubekova.net.",
        "key-name": "DHCP_UPDATER",
        "dns-servers": [
          { "ip-address": "192.168.1.1" }
        ]
      }
    ]
  },
  "reverse-ddns": {
    "ddns-domains": [
      {
        "name": "1.168.192.in-addr.arpa.",
        "key-name": "DHCP_UPDATER",
        "dns-servers": [
          { "ip-address": "192.168.1.1" }
        ]
      }
    ]
  }
}

```

Рисунок 4.33: Настройка DHCP,DNS

Изменим владельца файла и проверим файл на наличие возможных синтаксических ошибок. (рис. 4.34).

```

[root@server.antoychubekova.net kea]# kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf
2025-09-19 21:31:27.285 INFO [kea-dhcp-ddns.dctl/39617.140354598075968] DCTL_CONFIG_CHECK_COMPLETE server has completed configuration check: listening on 127.0.0.1, port 53001, using UDP, result: success(0), text-Configuration check successful
[root@server.antoychubekova.net kea]#

```

Рисунок 4.34: Проверка на наличие синтаксических ошибок

Запустим службу ddns и проверим статус работы службы, она имеет статус running. (рис. 4.35).

```

[root@server.antoychubekova.net kea]# systemctl enable --now kea-dhcp-ddns.service
Created symlink /etc/systemd/system/multi-user.target.wants/kea-dhcp-ddns.service → /usr/lib/systemd/system/kea-dhcp-ddns.service.
[root@server.antoychubekova.net kea]# systemctl status kea-dhcp-ddns.service
● kea-dhcp-ddns.service - Kea DHCP-DDNS Server
   Loaded: loaded (/usr/lib/systemd/system/kea-dhcp-ddns.service; enabled; preset: disabled)
   Active: active (running) since Fri 2025-09-19 21:33:13 UTC; 13s ago
     Invocation: 3ff1f9652260425fbbc95b158ed99eca
       Docs: man:kea-dhcp-ddns(8)
    Main PID: 39638 (kea-dhcp-ddns)
       Tasks: 5 (limit: 18374)
      Memory: 1.7M (peak: 6.1M)
         CPU: 186ms
    OGroup: /system.slice/kea-dhcp-ddns.service
           └─39638 /usr/sbin/kea-dhcp-ddns -c /etc/kea/kea-dhcp-ddns.conf

Sep 19 21:33:13 server.antoychubekova.net systemd[1]: Started kea-dhcp-ddns.service - Kea DHCP-DDNS Server.
Sep 19 21:33:13 server.antoychubekova.net kea-dhcp-ddns[39638]: 2025-09-19 21:33:13.802 INFO [kea-dhcp-ddns.dctl/39638.140057476821312] DCTL_START
Sep 19 21:33:13 server.antoychubekova.net kea-dhcp-ddns[39638]: INFO COMMAND_ACCEPTOR_START Starting to accept connections via unix domain socket
Sep 19 21:33:13 server.antoychubekova.net kea-dhcp-ddns[39638]: INFO DCTL_CONFIG_COMPLETE server has completed configuration: listening on 127.0.0.1
Sep 19 21:33:13 server.antoychubekova.net kea-dhcp-ddns[39638]: INFO DHCP_DDNS_STARTED Kea DHCP-DDNS server version 2.6.3 started
lines 1-17/27 (END)

```

Рисунок 4.35: Запуск службы ddns и проверка статуса работы

Внесем изменения в конфигурационный файл /etc/kea/kea-dhcp4.conf, добавив в него разрешение на динамическое обновление DNS-записей с локального узла прямой и обратной зон. (рис. 4.36).

```

},

"dhcp-ddns": {
    "enable-updates": true
},

"ddns-qualifying-suffix": "antoychubekova.net",
"ddns-override-client-update": true,

```

Рисунок 4.36: Редактирование конф файла kea-dhcp4.conf

Проверим файл на наличие возможных синтаксических ошибок. Все корректно отрабатывается. (рис. 4.37).

```

[root@server antoychubekova.net kea]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-09-19 21:38:37.011 INFO [kea-dhcp4.hosts/40809.140234522495168] HOSTS_BACKENDS_REGISTERED the following host backend types are available: mysql postgresql
2025-09-19 21:38:37.033 WARN [kea-dhcp4.dhcp4/40809.140234522495168] DHCP4_MT_DISABLED_QUEUE_CONTROL disabling dhcp queue control when multi-threading is enabled.
2025-09-19 21:38:37.034 WARN [kea-dhcp4.dhcp4/40809.140234522495168] DHCP4_RESERVATIONS_LOOKUP_FIRST_ENABLED Multi-threading is enabled and host reservations lookup is always performed first.
2025-09-19 21:38:37.047 INFO [kea-dhcp4.dhcp4/40809.140234522495168] DHCP4_SRV_CFGMGR_NEW_SUBNET4 a new subnet has been added to configuration: 192.168.1.0/24 with params: tl=0000, tc=10000, valid-lifetime=3600
2025-09-19 21:38:37.050 INFO [kea-dhcp4.dhcp4/40809.140234522495168] DHCP4_SRV_CFGMGR_SOCKET_TYPE_SELECT using socket type raw
2025-09-19 21:38:37.052 INFO [kea-dhcp4.dhcp4/40809.140234522495168] DHCP4_SRV_CFGMGR_ADD_IFACE listening on interface eth1.
2025-09-19 21:38:37.053 INFO [kea-dhcp4.dhcp4/40809.140234522495168] DHCP4_SRV_CFGMGR_SOCKET_TYPE_DEFAULT "dhcp-socket-type" not specified , using default socket type raw

```

Рисунок 4.37: Проверка файла на ошибки

Перезапустим DHCP-сервер и проверим статус. (рис. 4.38).

```

[root@server antoychubekova.net kea]# systemctl restart kea-dhcp4.service
[root@server antoychubekova.net kea]# systemctl status kea-dhcp4.service
● kea-dhcp4.service - Kea DHCPv4 Server
   Loaded: loaded (/usr/lib/systemd/system/kea-dhcp4.service; enabled; preset: disabled)
   Active: active (running) since Fri 2025-09-19 21:40:47 UTC; 10s ago
  Invocation: 3d3f03c64ed0d449ab8911f167ca13854
     Docs: man:kea-dhcp4(8)
   Main PID: 40868 (kea-dhcp4)
      Tasks: 7 (limit: 10374)
    Memory: 3M (peak: 6M)
       CPU: 150ms
   CGroup: /system.slice/kea-dhcp4.service
           └─40868 /usr/sbin/kea-dhcp4 -c /etc/kea/kea-dhcp4.conf

Sep 19 21:40:47 server.antoychubekova.net systemd[1]: Started kea-dhcp4.service - Kea DHCPv4 Server.
Sep 19 21:40:47 server.antoychubekova.net kea-dhcp4[40868]: 2025-09-19 21:40:47.747 INFO [kea-dhcp4.dhcp4/40868.139964407802048] DHCP4_STARTING Kea DHCPv4 server
Sep 19 21:40:47 server.antoychubekova.net kea-dhcp4[40868]: 2025-09-19 21:40:47.752 INFO [kea-dhcp4.commands/40868.139964407802048] COMMAND_RECEIVED Received command
lines 1-15/15 (END)

```

Рисунок 4.38: Запуск DHCP и проверка статуса

На машине client переполучим адрес. (рис. 4.39).

```

[antoychubekova@client.antoychubekova.net ~]$ nmcli connection down eth1
Connection 'eth1' successfully deactivated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/4)
[antoychubekova@client.antoychubekova.net ~]$ nmcli connection up eth1
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/6)
[antoychubekova@client.antoychubekova.net ~]$

```

Рисунок 4.39: Получение адреса клиентом

В каталоге прямой DNS-зоны /var/named/master/fz появился файл antoychubekova.net.jnl, в котором в бинарном файле автоматически вносятся изменения записей зоны. (рис. 4.40).

```
[root@server.antoychubekova.net kea]# cd /var/named/master/fz
[root@server.antoychubekova.net fz]# ls
antoychubekova.net  antoychubekova.net.jnl
[root@server.antoychubekova.net fz]#
```

Рисунок 4.40: Файл antoychubekova.net.jnl

На виртуальной машине client под нашим пользователем откроем терминал и с помощью утилиты dig убедимся в наличии DNS-записи о клиенте в прямой DNS-зоне: dig 192.168.1.1 client.user.net. (рис. 4.41).

```
[antoychubekova@client.antoychubekova.net ~]$ dig @192.168.1.1 client.antoychubekova.net

; <<>> DiG 9.18.33 <<>> @192.168.1.1 client.antoychubekova.net
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 34605
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: f272d6cc3dfad9830100000068cdcfcc6d47e45632b4777b (good)
;; QUESTION SECTION:
;client.antoychubekova.net.      IN      A

;; ANSWER SECTION:
client.antoychubekova.net. 1200 IN      A      192.168.1.30

;; Query time: 11 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Fri Sep 19 21:49:01 UTC 2025
;; MSG SIZE rcvd: 98

[antoychubekova@client.antoychubekova.net ~]$
```

Рисунок 4.41: DNS-записи

Команда выводит следующие сведения:

Общие сведения о запросе. Используется утилита dig, которая выполняет DNS-запрос к серверу 192.168.1.1. Запрашивалось имя client.antoychubekova.net с типом записи A (IPv4-адрес). Сервер был найден и дал ответ.

Заголовок ответа. В блоке HEADER указано, что запрос выполнен успешно (status: NOERROR). Флаги qr и aa означают, что это ответ (qr) и он авторитетный (aa). Флаги rd и ra показывают, что рекурсия была запрошена и разрешена.

Дополнительные параметры. В секции OPT PSEUDOSECTION видим поддержку расширенного протокола EDNS, максимальный размер пакета 1232 байта. Присутствует cookie для проверки целостности обмена.

Секция вопроса. Здесь повторяется суть запроса: для домена client.antoychubekova.net нужен IPv4-адрес (тип A).

Секция ответа. DNS-сервер вернул запись: client.antoychubekova.net имеет IP-адрес 192.168.1.30. Время жизни записи (TTL) — 1200 секунд.

Дополнительная информация. Запрос выполнен за 11 миллисекунд. Ответ пришёл от сервера 192.168.1.1 по порту 53 (UDP). Время выполнения запроса зафиксировано — Fri Sep 19 21:49:01 UTC 2025. Размер сообщения — 98 байт.

На виртуальной машине server перейдем в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создадим в нём каталог dhcp, в который поместите в соответствующие подкаталоги конфигурационные файлы DHCP. (рис. 4.42).

```
[root@server.antoychubekova.net fz]# cd /vagrant/provision/server
[root@server.antoychubekova.net server]# mkdir -p /vagrant/provision/server/dhcp/etc/kea
[root@server.antoychubekova.net server]# cp -R /etc/kea/* /vagrant/provision/server/dhcp/etc/kea/
[root@server.antoychubekova.net server]#
```

Рисунок 4.42: Создание каталога и подкаталогов

Заменяем конфигурационные файлы DNS-сервера. (рис. 4.43).

```
[root@server.antoychubekova.net server]# cd /vagrant/provision/server/dns/
[root@server.antoychubekova.net dns]# cp -R /var/named/* /vagrant/provision/server/dns/var/named/
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/antoychubekova.net'? y
cp: overwrite '/vagrant/provision/server/dns/var/named/master/iz/192.168.1'? yes
[root@server.antoychubekova.net dns]# cp -R /etc/named/* /vagrant/provision/server/dns/etc/named/
cp: overwrite '/vagrant/provision/server/dns/etc/named/antoychubekova.net'? y
[root@server.antoychubekova.net dns]# █
```

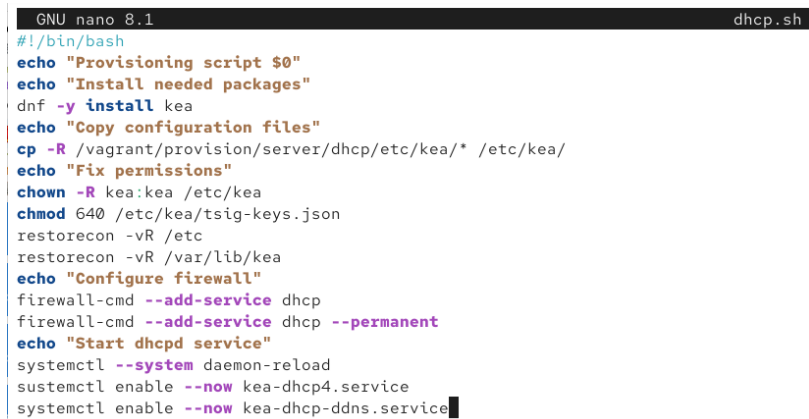
Рисунок 4.43: Редактирование конфигурационных файлов DNS-сервера

В каталоге /vagrant/provision/server создадим исполняемый файл dhcp.sh. (рис. 4.44).

```
{ [root@server.antoychubekova.net dns]# cd /vagrant/provision/server
[root@server.antoychubekova.net server]# touch dhcp.sh
bash: touch: command not found...
^C
[root@server.antoychubekova.net server]# touch dhcp.sh
[root@server.antoychubekova.net server]# chmod +x dhcp.sh
[root@server.antoychubekova.net server]#
```

Рисунок 4.44: Создание исполняемого файла

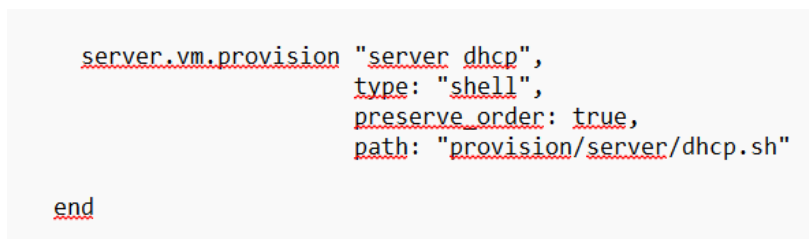
В нем напишем скрипт, по сути, повторяющий произведённые нами действия по установке и настройке DHCP-сервера. (рис. 4.45).



```
GNU nano 8.1 dhcp.sh
#!/bin/bash
echo "Provisioning script $0"
echo "Install needed packages"
dnf -y install kea
echo "Copy configuration files"
cp -R /vagrant/provision/server/dhcp/etc/kea/* /etc/kea/
echo "Fix permissions"
chown -R kea:kea /etc/kea
chmod 640 /etc/kea/tsig-keys.json
restorecon -vR /etc
restorecon -vR /var/lib/kea
echo "Configure firewall"
firewall-cmd --add-service dhcp
firewall-cmd --add-service dhcp --permanent
echo "Start dhcpd service"
systemctl --system daemon-reload
systemctl enable --now kea-dhcp4.service
systemctl enable --now kea-dhcp-ddns.service
```

Рисунок 4.45: Редактирование исполняемого файла

Для отработки созданного скрипта во время загрузки виртуальной машины server в конфигурационном файле Vagrantfile необходимо добавить в разделе конфигурации для сервера. (рис. 4.46).



```
server.vm.provision "server dhcp",
  type: "shell",
  preserve_order: true,
  path: "provision/server/dhcp.sh"

end
```

Рисунок 4.46: Редактирование Vagrantfile

Выключаем виртуальные машины.

5 Выводы

В ходе выполнения лабораторной работы №3 я приобрела практические навыки по установке и конфигурированию DHCP-сервера.

6 Список литературы

1. Barr D. Common DNS Operational and Configuration Errors: RFC / RFC Editor. — 02/1996. — DOI: 10.17487/rfc1912.
2. Droms R. Dynamic Host Configuration Protocol: RFC / RFC Editor. — 03/1997. — P. 1–45. — DOI: 10.17487/rfc2131.
3. Dynamic Updates in the Domain Name System (DNS UPDATE), RFC 2136: RFC / P. Vixie, S. Thomson, Y. Rekhter, J. Bound; RFC Editor. — 04/1997. — DOI: 10.17487/RFC2136.