

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра теории вероятностей и кибербезопасности

ОТЧЁТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №3

дисциплина: Администрирование сетевых подсистем

Студент: Чигладзе Майя Владиславовна

Студ. билет № 1132239399

Группа: НПИбд-02-23

МОСКВА

2025 г.

Цель работы:

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

Выполнение работы

Установка DHCP-сервера

1. Загрузим нашу операционную систему и перейдём в рабочий каталог с проектом:

```
cd /var/tmp/mvchigladze/vagrant
```

2. Далее запустим виртуальную машину server (Рис. 1.1):

```
make server-up
```

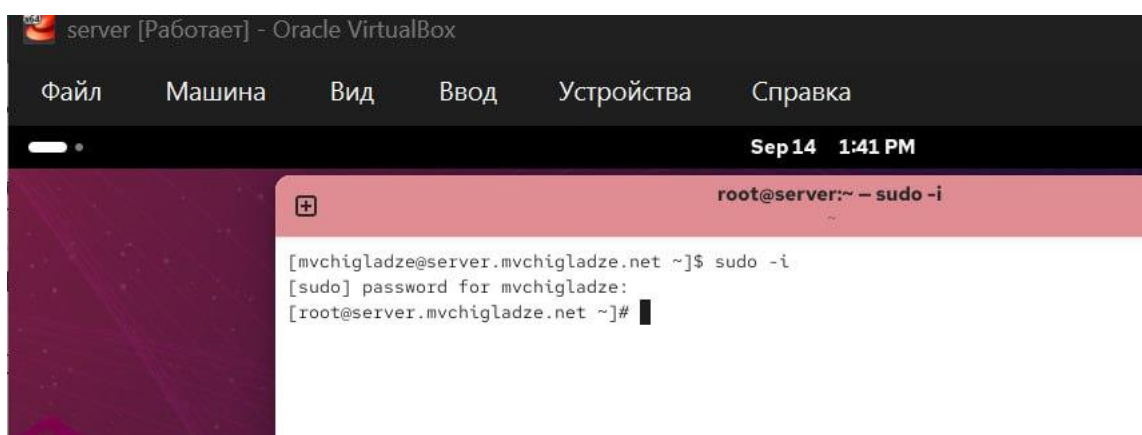


Рис. 1.1. Открытие рабочего каталога с проектом и запуск виртуальной машины server.

3. На виртуальной машине server войдём под нашим пользователем и откроем терминал. Перейдём в режим суперпользователя:

```
sudo -i
```

И установим dhcp (Рис. 1.2):

```
dnf -y install dhcp-server
```

```
Running scriptlet: kea-2.6.3-1.el10_0.x86_64
Installing       : kea-2.6.3-1.el10_0.x86_64
Running scriptlet: kea-2.6.3-1.el10_0.x86_64

Installed:
  kea-2.6.3-1.el10_0.x86_64          kea-`
  mariadb-connector-c-3.4.4-1.el10.x86_64 marii

Complete!
[root@server.mvchigladze.net ~]#
```

Рис. 1.2. Переход в режим суперпользователя и установка dhcр.

Конфигурирование DHCP-сервера

1. Скопируем файл примера конфигурации DHCP `dhcpd.conf.example` из каталога `/usr/share/doc/dhcp*` в каталог `/etc/dhcp` и переименуем его в файл с названием `dhcpd.conf` (Рис. 2.1):

```
cd /etc/dhcp
```

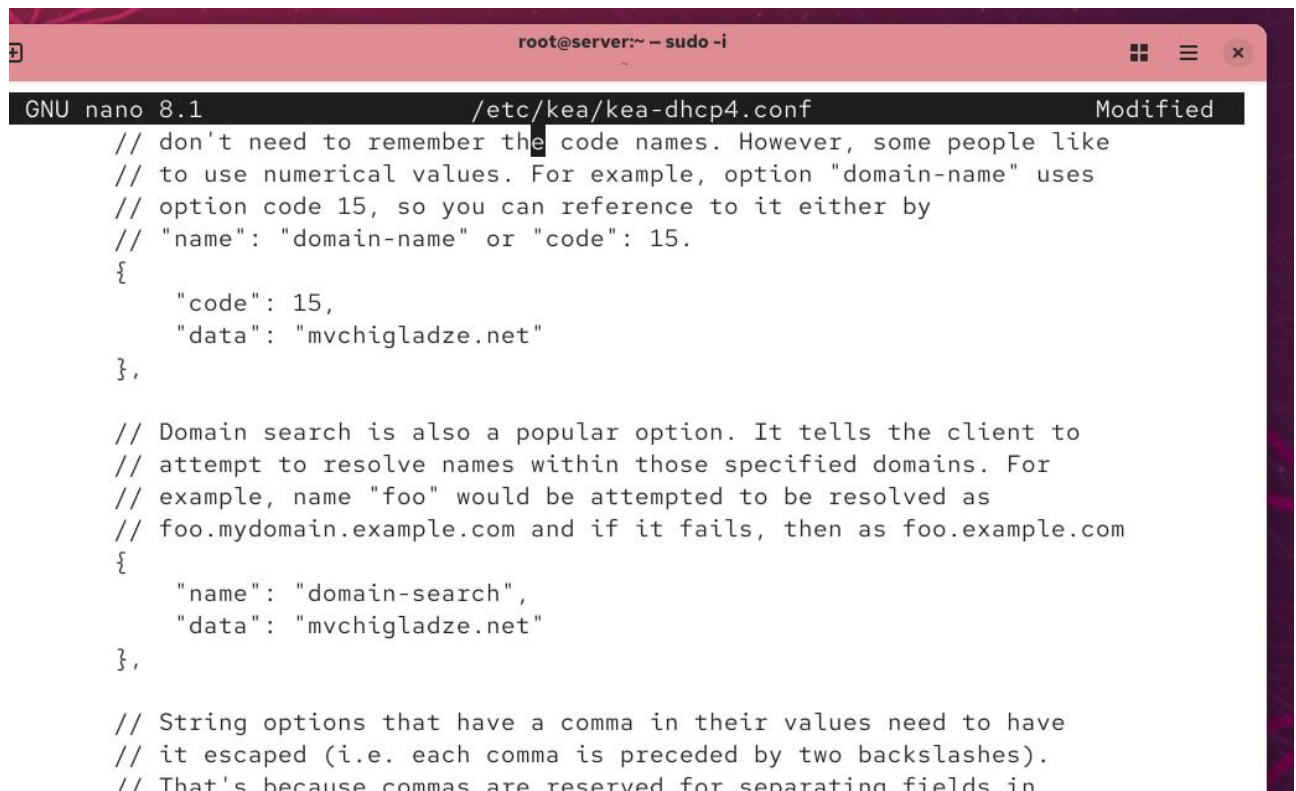
```
cp /usr/share/doc/dhcp*/dhcpd.conf.example /etc/dhcp
```

```
mv /etc/dhcp/dhcpd.conf.example /etc/dhcp/dhcpd.conf
```



Рис. 2.1. Копирование файла примера конфигурации DHCP и изменение его названия.

2. Откроем файл `/etc/dhcp/dhcpd.conf` на редактирование. В этом файле:
 - Заменим строку `option domain-name`
 - Заменим строку `option domain-name-servers`
 - Раскомментируем строку `authoritative`
 - 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`-адрес (Рис. 2.2).



```
root@server:~ - sudo -i
GNU nano 8.1 /etc/kea/kea-dhcp4.conf Modified
// don't need to remember the code names. However, some people like
// to use numerical values. For example, option "domain-name" uses
// option code 15, so you can reference to it either by
// "name": "domain-name" or "code": 15.
{
    "code": 15,
    "data": "mvchigladze.net"
},

// Domain search is also a popular option. It tells the client to
// attempt to resolve names within those specified domains. For
// example, name "foo" would be attempted to be resolved as
// foo.mydomain.example.com and if it fails, then as foo.example.com
{
    "name": "domain-search",
    "data": "mvchigladze.net"
},

// String options that have a comma in their values need to have
// it escaped (i.e. each comma is preceded by two backslashes).
// That's because commas are reserved for separating fields in
```

Рис. 2.2. Открытие файла /etc/dhcp/dhcpd.conf на редактирование. Замена строки option domain-name и option domain-name-servers, снятие комментария со строки authoritative, создание собственной конфигурации dhcp-сети.

3. Настройте привязку dhcpd к интерфейсу eth1 виртуальной машины server

```
"Dhcp4": {
    "interfaces-config": {
        "interfaces": [eth1 ]
    },

    // "control-socket": {
```

Рис. 2.3. Настройка привязки

4. Проверьте правильность конфигурационного файла:

```
kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
```



```
root@server:~ - sudo -i
GNU nano 8.1 /var/named/master/fz/mvchigladze.net
$TTL 1D
@      IN SOA  @ server.mvchigladze.net. (
                                2025090900      ; serial
                                1D      ; refresh
                                1H      ; retry
                                1W      ; expire
                                3H )   ; minimum

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

Рис. 2.6. Добавляем запись

7. Перезапустите named

```
[root@server.mvchigladze.net ~]# systemctl restart named
[root@server.mvchigladze.net ~]#
```

Рис. 2.7. Перезапускаем

8. Проверьте, что можно обратиться к DHCP-серверу по имени

```
[root@server.mvchigladze.net ~]# ping dhcp.mvchigladze.net
ping: dhcp.mvchigladze.net: Name or service not known
[root@server.mvchigladze.net ~]# nano /etc/resolv.conf
[root@server.mvchigladze.net ~]# nmcli connection modify eth1 ipv4.dns "127.0.0.1"
[root@server.mvchigladze.net ~]# nmcli connection up eth1
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/4)
[root@server.mvchigladze.net ~]# nano /etc/resolv.conf
[root@server.mvchigladze.net ~]# ping dhcp.mvchigladze.net
PING dhcp.mvchigladze.net (192.168.1.1): 56(84) bytes of data.
64 bytes from dhcp.mvchigladze.net (192.168.1.1): icmp_seq=1 ttl=64 time=0.246 ms
```

Рис. 2.8. Проверяем

9. Внесите изменения в настройки межсетевого экрана узла server, разрешив работу с DHCP


```
root@server:~ - sudo -i
firewall-cmd --add-service=dhcp --permanent
cockpit dhcpv6-client dns ssh
0-AD RH-Satellite-6 RH-Satellite-6-capsule afp alvr amanda-client amanda-k5-client
anno-1602 anno-1800 apcupsd aseqnet audit ausweisapp2 bacula bacula-client bareos
bareos-filedaemon bareos-storage bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitco
c bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent civilization
ion-v cockpit collectd condor-collector cratedb ctdb dds dds-multicast dds-unica
6 dhcpv6-client distcc dns dns-over-quic dns-over-tls docker-registry docker-swa
nsync elasticsearch etcd-client etcd-server factorio finger foreman foreman-prox
reeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp galera ganglia-c
-master git gpsd grafana gre high-availability http http3 https ident imap imaps
3 ipfs ipp ipp-client ipsec irc ircs iscsi-target isns jenkins kadmin kdeconnect
ana klogin kpasswd kprop kshell kube-api kube-apiserver kube-control-plane kube-
-secure kube-controller-manager kube-controller-manager-secure kube-nodeport-ser
heduler kube-scheduler-secure kube-worker kubelet kubelet-readonly kubelet-work
libvirt libvirt-tls lightning-network llmnr llmnr-client llmnr-tcp llmnr-udp man
ix mdns memcache minecraft minidlna mndp mongodb mosh mountd mpd mqtt mqtt-tls m
urmur mysql nbd nebula need-for-speed-most-wanted netbios-ns netdata-dashboard n
0183 nrpe ntp nut opentelemetry openvpn ovirt-imageio ovirt-storageconsole ovirt
ex pmcd pmproxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy prometheus prom
xporter proxy-dhcp ps2link ps3netsrv ptp pulseaudio puppetmaster quassel radius
dis redis-sentinel rootd rpc-bind rquotad rsh rsyncd rtsp salt-master samba samba
a-dc sane settlers-history-collection sip sips slimevr slp smtp smtp-submission
mptls snmptls-trap snmptrap spideroak-lansync spotify-sync squid ssdp ssh statsr
ansfer steam-streaming stellaris stronghold-crusader stun stuns submission supe
p svn syncthing syncthing-gui syncthing-relay synergy syscomlan syslog syslog-tl
acle terraria tftp tile38 tinc tor-socks transmission-client turn turns upnp-clie
server vrrp warpinator wbem-http wbem-https wireguard ws-discovery ws-discovery-
covery-host ws-discovery-tcp ws-discovery-udp wsdd wsdd-http wsman wsmans xdmcp
p-client xmpp-local xmpp-server zabbix-agent zabbix-java-gateway zabbix-server z
zabbix-web-service zero-k zerotier
success
success
root@server.mvchigladze.net ~# █
```

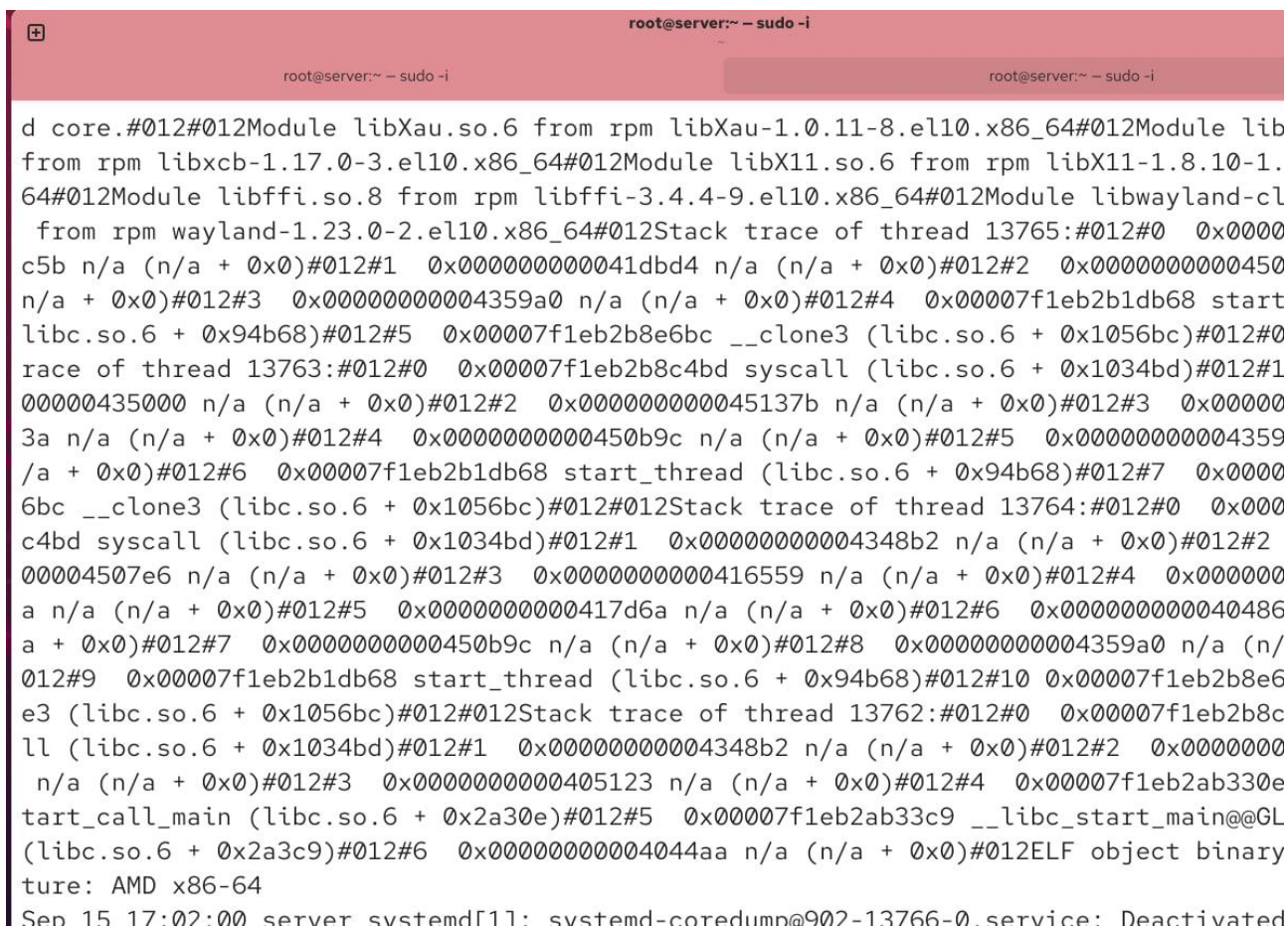
Рис. 2.9. Вносим изменения

10. Восстановите контекст безопасности в SELinux

```
success
[root@server.mvchigladze.net ~]# restorecon -vR /etc
restorecon -vR /var/named
restorecon -vR /var/lib/kea/
[root@server.mvchigladze.net ~]# █
```

Рис. 2.10. Восстанавливаем контекст

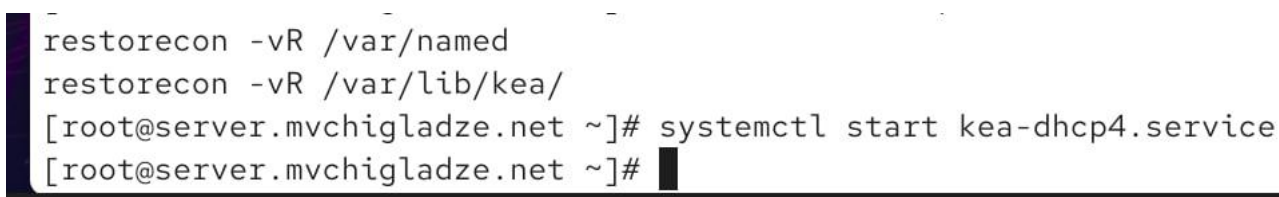
11. В дополнительном терминале запустите мониторинг происходящих в системе процессов в реальном времени



```
root@server:~ - sudo -i
root@server:~ - sudo -i
root@server:~ - sudo -i
d core.#012#012Module libXau.so.6 from rpm libXau-1.0.11-8.el10.x86_64#012Module lib
from rpm libxcb-1.17.0-3.el10.x86_64#012Module libX11.so.6 from rpm libX11-1.8.10-1.
64#012Module libffi.so.8 from rpm libffi-3.4.4-9.el10.x86_64#012Module libwayland-cl
from rpm wayland-1.23.0-2.el10.x86_64#012Stack trace of thread 13765:#012#0 0x0000
c5b n/a (n/a + 0x0)#012#1 0x000000000041dbd4 n/a (n/a + 0x0)#012#2 0x0000000000450
n/a + 0x0)#012#3 0x00000000004359a0 n/a (n/a + 0x0)#012#4 0x00007f1eb2b1db68 start
libc.so.6 + 0x94b68)#012#5 0x00007f1eb2b8e6bc __clone3 (libc.so.6 + 0x1056bc)#012#0
race of thread 13763:#012#0 0x00007f1eb2b8c4bd syscall (libc.so.6 + 0x1034bd)#012#1
00000435000 n/a (n/a + 0x0)#012#2 0x000000000045137b n/a (n/a + 0x0)#012#3 0x000000
3a n/a (n/a + 0x0)#012#4 0x0000000000450b9c n/a (n/a + 0x0)#012#5 0x00000000004359
/a + 0x0)#012#6 0x00007f1eb2b1db68 start_thread (libc.so.6 + 0x94b68)#012#7 0x0000
6bc __clone3 (libc.so.6 + 0x1056bc)#012#012Stack trace of thread 13764:#012#0 0x000
c4bd syscall (libc.so.6 + 0x1034bd)#012#1 0x00000000004348b2 n/a (n/a + 0x0)#012#2
00004507e6 n/a (n/a + 0x0)#012#3 0x0000000000416559 n/a (n/a + 0x0)#012#4 0x0000000
a n/a (n/a + 0x0)#012#5 0x0000000000417d6a n/a (n/a + 0x0)#012#6 0x000000000040486
a + 0x0)#012#7 0x0000000000450b9c n/a (n/a + 0x0)#012#8 0x00000000004359a0 n/a (n/
012#9 0x00007f1eb2b1db68 start_thread (libc.so.6 + 0x94b68)#012#10 0x00007f1eb2b8e6
e3 (libc.so.6 + 0x1056bc)#012#012Stack trace of thread 13762:#012#0 0x00007f1eb2b8c
ll (libc.so.6 + 0x1034bd)#012#1 0x00000000004348b2 n/a (n/a + 0x0)#012#2 0x00000000
n/a (n/a + 0x0)#012#3 0x0000000000405123 n/a (n/a + 0x0)#012#4 0x00007f1eb2ab330e
tart_call_main (libc.so.6 + 0x2a30e)#012#5 0x00007f1eb2ab33c9 __libc_start_main@GL
(libc.so.6 + 0x2a3c9)#012#6 0x00000000004044aa n/a (n/a + 0x0)#012ELF object binary
ture: AMD x86-64
Sep 15 17:02:00 server systemd[1]: systemd-coredump@902-13766-0.service: Deactivated
```

Рис. 2.11. Запускаем мониторинг

12. В основном рабочем терминале запустите DHCP-сервер



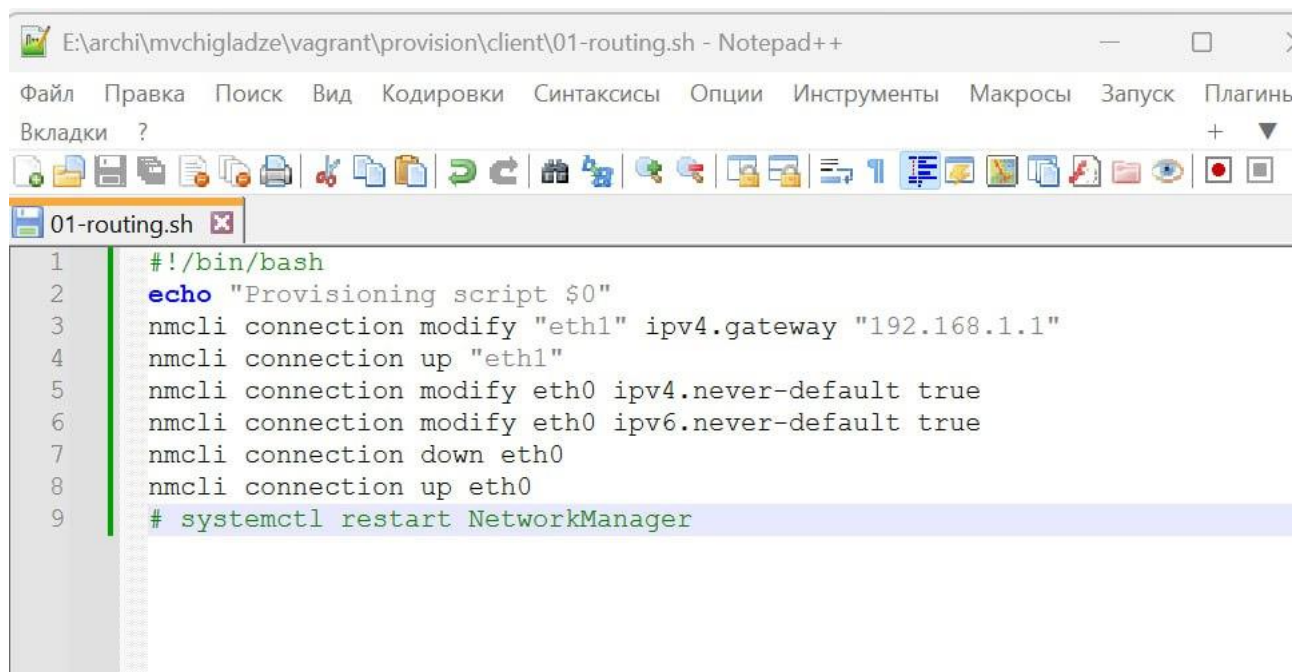
```
restorecon -vR /var/named
restorecon -vR /var/lib/kea/
[root@server.mvchigladze.net ~]# systemctl start kea-dhcp4.service
[root@server.mvchigladze.net ~]#
```

Рис. 2.12. Запускаем сервер

13. Если запуск DHCP-сервера прошёл успешно, то, не выключая виртуальной машины server и не прерывая на ней мониторинга происходящих в системе процессов, приступите к анализу работы DHCP-сервера на клиенте

3.4.3. Анализ работы DHCP-сервера

1. Перед запуском виртуальной машины client в каталоге с проектом в вашей операционной системе в подкаталоге `vagrant/provision/client` создайте файл `01-routing.sh`



The screenshot shows a Notepad++ window titled "E:\archi\mvchigladze\vagrant\provision\client\01-routing.sh - Notepad++". The menu bar includes "Файл", "Правка", "Поиск", "Вид", "Кодировки", "Синтаксис", "Опции", "Инструменты", "Макросы", "Запуск", and "Плагин". The toolbar contains various icons for file operations and editing. The script content is as follows:

```
1  #!/bin/bash
2  echo "Provisioning script $0"
3  nmcli connection modify "eth1" ipv4.gateway "192.168.1.1"
4  nmcli connection up "eth1"
5  nmcli connection modify eth0 ipv4.never-default true
6  nmcli connection modify eth0 ipv6.never-default true
7  nmcli connection down eth0
8  nmcli connection up eth0
9  # systemctl restart NetworkManager
```

2. В Vagrantfile подключите этот скрипт в разделе конфигурации для клиента:

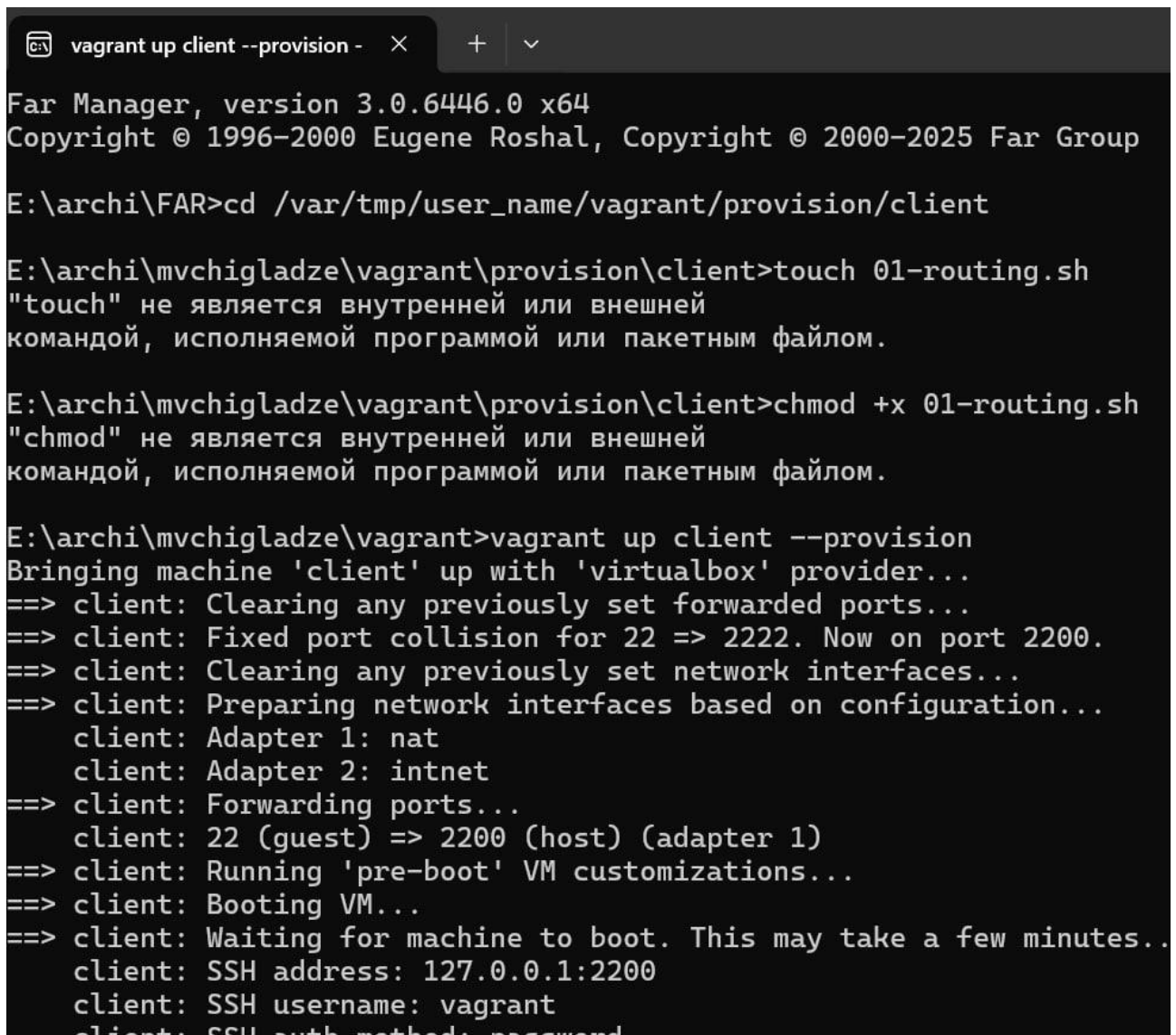
`client.vm.provision "client routing",`

`type: "shell",`

`preserve_order: true,`

`run: "always",`

`path: "provision/client/01-routing.sh"`



```
vagrant up client --provision - X + v
Far Manager, version 3.0.6446.0 x64
Copyright © 1996–2000 Eugene Roshal, Copyright © 2000–2025 Far Group

E:\archi\FAR>cd /var/tmp/user_name/vagrant/provision/client

E:\archi\mvchigladze\vagrant\provision\client>touch 01-routing.sh
"touch" не является внутренней или внешней
командой, исполняемой программой или пакетным файлом.

E:\archi\mvchigladze\vagrant\provision\client>chmod +x 01-routing.sh
"chmod" не является внутренней или внешней
командой, исполняемой программой или пакетным файлом.

E:\archi\mvchigladze\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
```

Рис. 3.3. Фиксируем внесенные изменения

- После загрузки виртуальной машины `client` вы можете увидеть на виртуальной машине `server` на терминале с мониторингом происходящих в системе процессов записи о подключении к виртуальной внутренней сети узла `client` и выдачи ему IP-адреса из соответствующего диапазона адресов. Также информацию о работе DHCP-сервера можно наблюдать в файле `/var/lib/kea/kea-leases4.csv`. В отчёте прокомментируйте построчно информацию из этого файла.

6. Войдите в систему виртуальной машины client под вашим пользователем и откройте терминал.
7. На машине server посмотрите список выданных адресов

```
cat /var/lib/kea/kea-leases4.csv
```

Прокомментируйте построчно в отчёте.

```
[root@server.mvchigladze.net ~]# cat /var/lib/kea/kea-leases4.csv
address,hwaddr,client_id,valid_lifetime,expire,subnet_id,fqdn_fwd,fqdn_rev,hostname,state,use
r_context,pool_id
[...]
```

Рис. 3.7. Смотрим список выданных адресов

Теперь построчно поясним каждое поле:

address – выданный клиенту IP-адрес. Это основной результат работы DHCP-сервера.

hwaddr – MAC-адрес сетевой карты клиента, по которому DHCP-сервер идентифицирует устройство.

client_id – уникальный идентификатор клиента (DHCP Client Identifier). Может содержать дополнительную информацию, которая позволяет отличать клиентов даже с одинаковыми MAC-адресами.

valid_lifetime – время жизни аренды (в секундах). Определяет, на какой срок клиент получил IP-адрес.

expire – точное время (timestamp или дата), когда аренда закончится, если клиент её не продлит.

subnet_id – идентификатор подсети, из которой выделен адрес. Это помогает серверу понимать, в какой сети был задействован IP.

fqdn_fwd – признак, выполнял ли сервер прямое обновление DNS-записи (Forward DNS).

fqdn_rev – признак, выполнял ли сервер обратное обновление DNS-записи (Reverse DNS).

hostname – имя хоста клиента (если он его предоставил при DHCP-запросе).

state – состояние аренды

0 – активная аренда,

1 – аренда освобождена,

2 – ожидает подтверждения и т.п.

user_context – дополнительная информация в формате JSON (если используется). Обычно пустое поле.

pool_id – идентификатор пула адресов, из которого был выдан IP.

3.4.4. Настройка обновления DNS-зоны

Требуется настроить обновление DNS-зоны при появлении в виртуальной внутренней сети новых узлов.

1. Создадим ключ на сервере с Bind9 (на виртуальной машине server):

```
mkdir -p /etc/named/keys
```

```
tsig-keygen -a HMAC-SHA512 DHCP_UPDATER >
```

```
/etc/named/keys/dhcp_updater.key
```

```
[root@server.mvchigladze.net ~]# mkdir -p /etc/named/keys
[root@server.mvchigladze.net ~]# tsig-keygen -a HMAC-SHA512 DHCP_UPDATER > /etc/named/keys/dh
cp_updater.key
[root@server.mvchigladze.net ~]# █
```

Рис. 4.1. Создаем ключ

2. Файл /etc/named/keys/dhcp_updater.key будет иметь следующий вид

```
[root@server.mvchigladze.net ~]# cat /etc/named/keys/dhcp_updater.key
key "DHCP_UPDATER" {
    algorithm hmac-sha512;
    secret "uSspgppFccyFFeV/SdLdKRB++SpeuBz4eZhXJw4hL3xAFh+uXXMh9RCsTJ1WJJRVGY+wjoNzjgRqy
x10HJZyKA==";
};
[root@server.mvchigladze.net ~]# █
```

Рис. 4.2. Файл апдейтер

3. Поправим права доступа:

```
chown -R named:named /etc/named/keys
```

```
[root@server.mvchigladze.net ~]# chown -R named:named /etc/named/keys
[root@server.mvchigladze.net ~]#
```

Рис. 4.3. Права

4. Подключим ключ в файле /etc/named.conf

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

Рис. 4.4. Ключ

5. На виртуальной машине server под пользователем с правами суперпользователя отредактируйте файл /etc/named/user.net (вместо user укажите свой логин), разрешив обновление зоны

```
GNU nano 8.1 /etc/named/mvchigladze.net
// Provided by Red Hat caching-nameserver package
//
// ISC BIND named zone configuration for zones recommended by
// RFC 1912 section 4.1 : localhost TLDs and address zones
// and https://tools.ietf.org/html/rfc6303
// (c)2007 R W Franks
//
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//
// Note: empty-zones-enable yes; option is default.
// If private ranges should be forwarded, add
// disable-empty-zone "."; into options
//

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

Рис. 4.5. Редактируем файл

6. Сделаем проверку конфигурационного файла:

`named-checkconf`

```
root@server.mvchigladze.net ~]# named-checkconf
root@server.mvchigladze.net ~]#
```

Рис. 4.6. Проверка конфигурационного файла

7. Перезапустите DNS-сервер:

`systemctl restart named`

```
[root@server.mvchigladze.net ~]# named-checkconf
[root@server.mvchigladze.net ~]# systemctl restart named
[root@server.mvchigladze.net ~]#
```

Рис. 4.7. Перезапуск сервера

8. Сформируем ключ для Кеа. Файл ключа назовём `/etc/kea/tsig-keys.json`:

`touch /etc/kea/tsig-keys.json`

```
[root@server.mvchigladze.net ~]# systemctl restart named
[root@server.mvchigladze.net ~]# touch /etc/kea/tsig-keys.json
```

Рис. 4.8. Формируем ключ

9. Перенесём ключ на сервер Кеа ДНСП и перепишем его в формате json

```
GNU nano 8.1 /etc/kea/tsig-keys.json
"tsig-keys": [
  {
    "name": "DHCP_UPDATER",
    "algorithm": "hmac-sha512",
    "secret": "uSspgppFccyFFeV/SdLdKRB++SpeuBz4eZhXJw4hL3xAFh+uXXMh9RCsTJ1WJJ"
  }
],
```

Рис. 4.9. Переносим ключ

10. Сменим владельца:

`chown kea:kea /etc/kea/tsig-keys.json`

```
[root@server.mvchigladze.net ~]# chown kea:kea /etc/kea/tsig-keys.json
[root@server.mvchigladze.net ~]# chmod 640 /etc/kea/tsig-keys.json
[root@server.mvchigladze.net ~]#
```

Рис. 4.10. Меняем владельца

11. Поправим права доступа:

`chmod 640 /etc/kea/tsig-keys.json`

```
[root@server.mvchigladze.net ~]# chown kea:kea /etc/kea/tsig-keys.json
[root@server.mvchigladze.net ~]# chmod 640 /etc/kea/tsig-keys.json
[root@server.mvchigladze.net ~]#
```

Рис. 4.11. Правим права доступа

12. Настройка происходит в файле `/etc/kea/kea-dhcp-ddns.conf`

```
root@server:~ - sudo -i
root@server:~ - sudo -i
GNU nano 8.1 /etc/kea/kea-dhcp-ddns.conf
{
    "name": "mvchigladze.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" }
            ]
        }
    ]
},
// Logging configuration starts here. Kea uses different loggers to log various
// activities. For details (e.g. names of loggers), see Chapter 18.
"loggers": [
    {
        // This specifies the logging for D2 (DHCP-DDNS) daemon.
        "name": "kea-dhcp-ddns"
```

Рис. 4.12. Настройка в файле

13. Изменим владельца файла:

```
chown kea:kea /etc/kea/kea-dhcp-ddns.conf
```

```

[root@server.mvchigladze.net ~]# chown kea:kea /etc/kea/kea-dhcp-ddns.conf
[root@server.mvchigladze.net ~]# kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf
2025-09-15 17:42:08.786 INFO [kea-dhcp-ddns.dctl/18611.139699201663296] DCTL_CONFIG_CHECK_CO
MPLETE server has completed configuration check: listening on 127.0.0.1, port 53001, using UD
P, result: success(0), text=Configuration check successful
[root@server.mvchigladze.net ~]# systemctl enable --now kea-dhcp-ddns.service
Created symlink '/etc/systemd/system/multi-user.target.wants/kea-dhcp-ddns.service' → '/usr/l
ib/systemd/system/kea-dhcp-ddns.service'.
[root@server.mvchigladze.net ~]# 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: disable>
   Active: active (running) since Mon 2025-09-15 17:42:42 UTC; 10s ago
 Invocation: c90144e0a06f44509099b2c90210f045
    Docs: man:kea-dhcp-ddns(8)
  Main PID: 18825 (kea-dhcp-ddns)
    Tasks: 5 (limit: 10369)
   Memory: 1.8M (peak: 5.8M)
      CPU: 28ms
   CGroup: /system.slice/kea-dhcp-ddns.service
           └─18825 /usr/sbin/kea-dhcp-ddns -c /etc/kea/kea-dhcp-ddns.conf

Sep 15 17:42:42 server.mvchigladze.net systemd[1]: Started kea-dhcp-ddns.service - Kea DHCP->
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: 2025-09-15 17:42:42.680 INFO [>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO COMMAND_ACCEPTOR_START St>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO DCTL_CONFIG_COMPLETE serv>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO DHCP_DDNS_STARTED Kea DHC>

[4]+ Stopped                                systemctl status kea-dhcp-ddns.service
[root@server.mvchigladze.net ~]# █

```

Рис. 4.13. Меняем владельца

14. Проверим файл на наличие возможных синтаксических ошибок:

```
kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf
```



```

[root@server.mvchigladze.net ~]# chown kea:kea /etc/kea/kea-dhcp-ddns.conf
[root@server.mvchigladze.net ~]# kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf
2025-09-15 17:42:08.786 INFO [kea-dhcp-ddns.dctl/18611.139699201663296] DCTL_CONFIG_CHECK_CO
MPLETE server has completed configuration check: listening on 127.0.0.1, port 53001, using UD
P, result: success(0), text=Configuration check successful
[root@server.mvchigladze.net ~]# systemctl enable --now kea-dhcp-ddns.service
Created symlink '/etc/systemd/system/multi-user.target.wants/kea-dhcp-ddns.service' -> '/usr/l
ib/systemd/system/kea-dhcp-ddns.service'.
[root@server.mvchigladze.net ~]# 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: disable>
   Active: active (running) since Mon 2025-09-15 17:42:42 UTC; 10s ago
 Invocation: c90144e0a06f44509099b2c90210f045
    Docs: man:kea-dhcp-ddns(8)
 Main PID: 18825 (kea-dhcp-ddns)
   Tasks: 5 (limit: 10369)
  Memory: 1.8M (peak: 5.8M)
     CPU: 28ms
    CGroup: /system.slice/kea-dhcp-ddns.service
            └─18825 /usr/sbin/kea-dhcp-ddns -c /etc/kea/kea-dhcp-ddns.conf

Sep 15 17:42:42 server.mvchigladze.net systemd[1]: Started kea-dhcp-ddns.service - Kea DHCP->
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: 2025-09-15 17:42:42.680 INFO [>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO COMMAND_ACCEPTOR_START St>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO DCTL_CONFIG_COMPLETE serv>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO DHCP_DDNS_STARTED Kea DHC>

[4]+  Stopped                                systemctl status kea-dhcp-ddns.service
[root@server.mvchigladze.net ~]# █

```

Рис. 4.14. Проверяем ошибки

15. Запустим службу ddns:

`systemctl enable --now kea-dhcp-ddns.service`

```

[root@server.mvchigladze.net ~]# chown kea:kea /etc/kea/kea-dhcp-ddns.conf
[root@server.mvchigladze.net ~]# kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf
2025-09-15 17:42:08.786 INFO [kea-dhcp-ddns.dctl/18611.139699201663296] DCTL_CONFIG_CHECK_CO
MPLETE server has completed configuration check: listening on 127.0.0.1, port 53001, using UD
P, result: success(0), text=Configuration check successful
[root@server.mvchigladze.net ~]# systemctl enable --now kea-dhcp-ddns.service
Created symlink '/etc/systemd/system/multi-user.target.wants/kea-dhcp-ddns.service' → '/usr/l
ib/systemd/system/kea-dhcp-ddns.service'.
[root@server.mvchigladze.net ~]# 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: disable>
   Active: active (running) since Mon 2025-09-15 17:42:42 UTC; 10s ago
 Invocation: c90144e0a06f44509099b2c90210f045
    Docs: man:kea-dhcp-ddns(8)
 Main PID: 18825 (kea-dhcp-ddns)
   Tasks: 5 (limit: 10369)
  Memory: 1.8M (peak: 5.8M)
     CPU: 28ms
   CGroup: /system.slice/kea-dhcp-ddns.service
           └─18825 /usr/sbin/kea-dhcp-ddns -c /etc/kea/kea-dhcp-ddns.conf

Sep 15 17:42:42 server.mvchigladze.net systemd[1]: Started kea-dhcp-ddns.service - Kea DHCP->
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: 2025-09-15 17:42:42.680 INFO [>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO COMMAND_ACCEPTOR_START St>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO DCTL_CONFIG_COMPLETE serv>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO DHCP_DDNS_STARTED Kea DHC>

[4]+ Stopped                                systemctl status kea-dhcp-ddns.service
[root@server.mvchigladze.net ~]# █

```

Рис. 4.15. Запускаем службу

16. Проверим статус работы службы:

```
systemctl status kea-dhcp-ddns.service
```

```

[root@server.mvchigladze.net ~]# chown kea:kea /etc/kea/kea-dhcp-ddns.conf
[root@server.mvchigladze.net ~]# kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf
2025-09-15 17:42:08.786 INFO [kea-dhcp-ddns.dctl/18611.139699201663296] 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.mvchigladze.net ~]# 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.mvchigladze.net ~]# 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 Mon 2025-09-15 17:42:42 UTC; 10s ago
 Invocation: c90144e0a06f44509099b2c90210f045
    Docs: man:kea-dhcp-ddns(8)
  Main PID: 18825 (kea-dhcp-ddns)
    Tasks: 5 (limit: 10369)
   Memory: 1.8M (peak: 5.8M)
      CPU: 28ms
   CGroup: /system.slice/kea-dhcp-ddns.service
           └─18825 /usr/sbin/kea-dhcp-ddns -c /etc/kea/kea-dhcp-ddns.conf

Sep 15 17:42:42 server.mvchigladze.net systemd[1]: Started kea-dhcp-ddns.service - Kea DHCP->
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: 2025-09-15 17:42:42.680 INFO [>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO COMMAND_ACCEPTOR_START St>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO DCTL_CONFIG_COMPLETE serv>
Sep 15 17:42:42 server.mvchigladze.net kea-dhcp-ddns[18825]: INFO DHCP_DDNS_STARTED Kea DHC>

[4]+  Stopped                                systemctl status kea-dhcp-ddns.service
[root@server.mvchigladze.net ~]# █

```

Рис. 4.16. Проверяем статус службы

17. Внесите 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



```
GNU nano 8.1 /etc/kea/kea-dhcp4.conf
{
"Dhcp4": {
  "interfaces-config": {
    "interfaces": ["eth1"]
  },
  "dhcp-ddns": {
    "enable-updates": true
  },
  "ddns-qualifying-suffix": "mvchigladze.net",
  "ddns-override-client-update": true,
//  "control-socket": {
//    "socket-type": "unix",
//    "socket-name": "kea4-ctrl-socket"
//  },
}
```

Рис. 4.17. Вносим изменения в файл

18. Проверим файл на наличие возможных синтаксических ошибок:

```
kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
```

```

[root@server.mvchigladze.net ~]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-09-15 17:46:38.658 INFO [kea-dhcp4.hosts/19313.139953452067008] HOSTS_BACKENDS_REGISTER
ED the following host backend types are available: mysql postgresql
2025-09-15 17:46:38.661 WARN [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_MT_DISABLED_Q
UEUE_CONTROL disabling dhcp queue control when multi-threading is enabled.
2025-09-15 17:46:38.661 WARN [kea-dhcp4.dhcp4/19313.139953452067008] DHCP4_RESERVATIONS_LOOK
UP_FIRST_ENABLED Multi-threading is enabled and host reservations lookup is always performed
first.
2025-09-15 17:46:38.663 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_NEW_SU
BNET4 a new subnet has been added to configuration: 192.168.1.0/24 with params: valid-lifetim
e=7200
2025-09-15 17:46:38.663 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_SOCKET
_TYPE_SELECT using socket type raw
2025-09-15 17:46:38.664 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_ADD_IF
ACE listening on interface eth1
2025-09-15 17:46:38.664 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_SOCKET
_TYPE_DEFAULT "dhcp-socket-type" not specified , using default socket type raw
[root@server.mvchigladze.net ~]# systemctl restart kea-dhcp4.service
[root@server.mvchigladze.net ~]# 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 Mon 2025-09-15 17:46:47 UTC; 5s ago
 Invocation: ab072fd850b24b268c7b2d1c676f26df
    Docs: man:kea-dhcp4(8)
 Main PID: 19342 (kea-dhcp4)
   Tasks: 7 (limit: 10369)
  Memory: 2.6M (peak: 5.9M)
     CPU: 40ms
   CGroup: /system.slice/kea-dhcp4.service
           └─19342 /usr/sbin/kea-dhcp4 -c /etc/kea/kea-dhcp4.conf

Sep 15 17:46:47 server.mvchigladze.net kea-dhcp4[19342]: 2025-09-15 17:46:47 213 INFO [kea-

```

Рис. 4.18. Проверяем файл на наличие ошибок

19. Перезапустите DHCP-сервер:

`systemctl restart kea-dhcp4.service`


```
[root@server.mvchigladze.net ~]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-09-15 17:46:38.658 INFO [kea-dhcp4.hosts/19313.139953452067008] HOSTS_BACKENDS_REGISTERED the following host backend types are available: mysql postgresql
2025-09-15 17:46:38.661 WARN [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_MT_DISABLED_QUEUE_CONTROL disabling dhcp queue control when multi-threading is enabled.
2025-09-15 17:46:38.661 WARN [kea-dhcp4.dhcp4/19313.139953452067008] DHCP4_RESERVATIONS_LOOKUP_FIRST_ENABLED Multi-threading is enabled and host reservations lookup is always performed first.
2025-09-15 17:46:38.663 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_NEW_SUBNET4 a new subnet has been added to configuration: 192.168.1.0/24 with params: valid-lifetime=7200
2025-09-15 17:46:38.663 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_SOCKET_TYPE_SELECT using socket type raw
2025-09-15 17:46:38.664 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_ADD_IFACE listening on interface eth1
2025-09-15 17:46:38.664 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_SOCKET_TYPE_DEFAULT "dhcp-socket-type" not specified , using default socket type raw
[root@server.mvchigladze.net ~]# systemctl restart kea-dhcp4.service
[root@server.mvchigladze.net ~]# 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 Mon 2025-09-15 17:46:47 UTC; 5s ago
 Invocation: ab072fd850b24b268c7b2d1c676f26df
    Docs: man:kea-dhcp4(8)
 Main PID: 19342 (kea-dhcp4)
   Tasks: 7 (limit: 10369)
  Memory: 2.6M (peak: 5.9M)
    CPU: 40ms
   CGroup: /system.slice/kea-dhcp4.service
           └─19342 /usr/sbin/kea-dhcp4 -c /etc/kea/kea-dhcp4.conf

Sep 15 17:46:47 server mvchigladze.net kea-dhcp4[19342]: 2025-09-15 17:46:47 213 INFO [kea-
```

Рис. 4.19. Перезапуск

20. Проверим статус:

```
systemctl status kea-dhcp4.service
```



```
[root@server.mvchigladze.net ~]# kea-dhcp4 -t /etc/kea/kea-dhcp4.conf
2025-09-15 17:46:38.658 INFO [kea-dhcp4.hosts/19313.139953452067008] HOSTS_BACKENDS_REGISTER
ED the following host backend types are available: mysql postgresql
2025-09-15 17:46:38.661 WARN [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_MT_DISABLED_Q
UEUE_CONTROL disabling dhcp queue control when multi-threading is enabled.
2025-09-15 17:46:38.661 WARN [kea-dhcp4.dhcp4/19313.139953452067008] DHCP4_RESERVATIONS_LOOK
UP_FIRST_ENABLED Multi-threading is enabled and host reservations lookup is always performed
first.
2025-09-15 17:46:38.663 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_NEW_SU
BNET4 a new subnet has been added to configuration: 192.168.1.0/24 with params: valid-lifetim
e=7200
2025-09-15 17:46:38.663 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_SOCKET
_TYPE_SELECT using socket type raw
2025-09-15 17:46:38.664 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_ADD_IF
ACE listening on interface eth1
2025-09-15 17:46:38.664 INFO [kea-dhcp4.dhcpsrv/19313.139953452067008] DHCP4_SRV_CFGMGR_SOCKET
_TYPE_DEFAULT "dhcp-socket-type" not specified , using default socket type raw
[root@server.mvchigladze.net ~]# systemctl restart kea-dhcp4.service
[root@server.mvchigladze.net ~]# 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 Mon 2025-09-15 17:46:47 UTC; 5s ago
 Invocation: ab072fd850b24b268c7b2d1c676f26df
    Docs: man:kea-dhcp4(8)
  Main PID: 19342 (kea-dhcp4)
    Tasks: 7 (limit: 10369)
   Memory: 2.6M (peak: 5.9M)
      CPU: 40ms
   CGroup: /system.slice/kea-dhcp4.service
           └─19342 /usr/sbin/kea-dhcp4 -c /etc/kea/kea-dhcp4.conf

Sep 15 17:46:47 server.mvchigladze.net kea-dhcp4[19342]: 2025-09-15 17:46:47 213 INFO [kea-
```

Рис. 4.20. Проверяем статус

21. На машине client переполучите адрес:

`nmcli connection down eth1`

`nmcli connection up eth1`

```
[mvchigladze@client.mvchigladze.net ~]$ nmcli connection down eth1
Connection 'eth1' successfully deactivated (D-Bus active path: /org/freedesktop/
NetworkManager/ActiveConnection/5)
[mvchigladze@client.mvchigladze.net ~]$ nmcli connection up eth1
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkMa
nager/ActiveConnection/7)
[mvchigladze@client.mvchigladze.net ~]$
```

Рис. 4.21. Переполучаем адрес

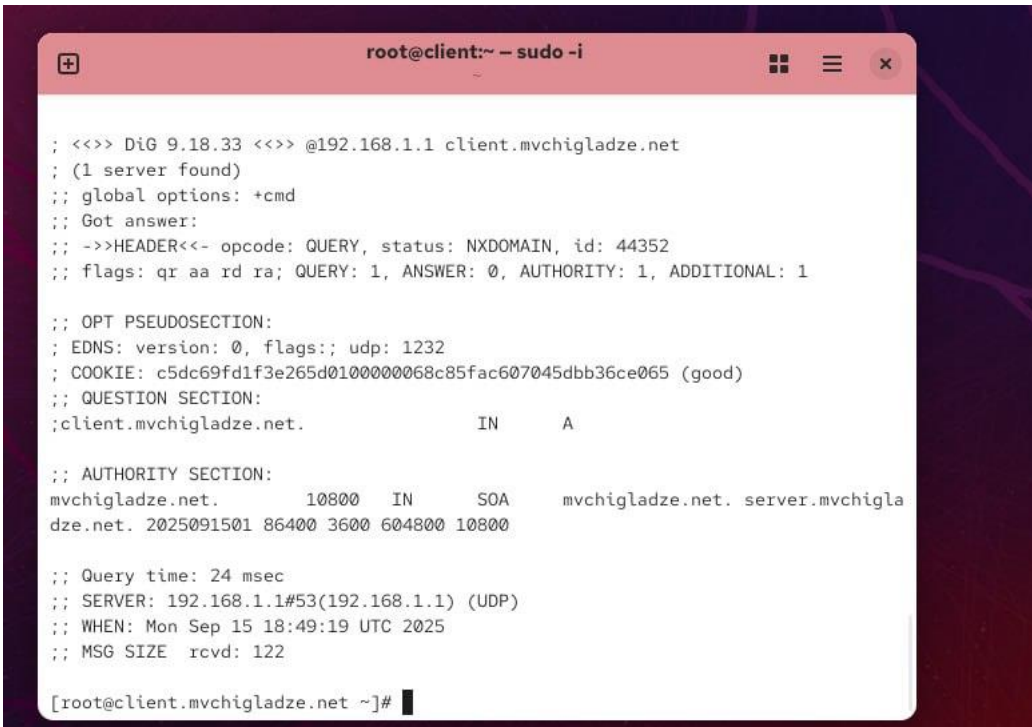
22. В каталоге прямой DNS-зоны /var/named/master/fz должен появиться файл user.net.jnl, в котором в бинарном файле автоматически вносятся изменения записей зоны.

3.4.5. Анализ работы DHCP-сервера после настройки обновления DNS-зоны

На виртуальной машине client под вашим пользователем откройте терминал и с помощью утилиты dig убедитесь в наличии DNS-записи о клиенте в прямой DNS-зоне

```
dig @192.168.1.1 client.user.net
```

В отчёте построчно прокомментируйте выведенную на экран информацию.



```
root@client:~ - sudo -i
; <<>> DiG 9.18.33 <<>> @192.168.1.1 client.mvchigladze.net
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 44352
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:;, udp: 1232
; COOKIE: c5dc69fd1f3e265d0100000068c85fac607045dbb36ce065 (good)
;; QUESTION SECTION:
;client.mvchigladze.net.                IN      A

;; AUTHORITY SECTION:
mvchigladze.net.        10800   IN      SOA     mvchigladze.net. server.mvchigla
dze.net. 2025091501 86400 3600 604800 10800

;; Query time: 24 msec
;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP)
;; WHEN: Mon Sep 15 18:49:19 UTC 2025
;; MSG SIZE rcvd: 122

[root@client.mvchigladze.net ~]#
```

Рис. 5.1. Анализ работы

Контекст:

- Пользователь вошел в систему как root на машине с именем `client.mvchigladze.net`.
- Он использовал `sudo -i`, чтобы стать пользователем root (вероятно, для обеспечения чистого окружения root).
- Основная часть вывода - это результат команды `dig`.

Анализ команды `dig`:

- `dig 9.18.33`: Указывает версию утилиты `dig`.
- `@192.168.1.1`: Указывает, что DNS-запрос отправляется на DNS-сервер с IP-адресом `192.168.1.1`. Вероятно, это локальный DNS-сервер.
- `client.mvchigladze.net`: Это доменное имя, для которого выполняется запрос.
- `IN A`: Указывает, что запрос предназначен для "A" записи (IPv4-адреса) домена `client.mvchigladze.net`.

Интерпретация вывода `dig`:

- `;; Получен ответ:` Указывает, что DNS-сервер ответил.
- `;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 44352`: Это самая важная часть.
 - * `opcode: QUERY`: Подтверждает, что это был DNS-запрос.
 - * `status: NXDOMAIN`: **Это ключевое!** `NXDOMAIN` означает "Non-Existent Domain" (Несуществующий домен). Это означает, что DNS-сервер по адресу `192.168.1.1` *не* имеет записи для `client.mvchigladze.net`.
- `;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1`: Это DNS-флаги.
 - * `qr`: Query Response (это ответ, а не запрос).
 - * `aa`: Authoritative Answer (DNS-сервер является авторитетным для зоны).
 - * `rd`: Recursion Desired (клиент попросил сервер выполнить рекурсивные запросы).
 - * `ra`: Recursion Available (сервер поддерживает рекурсивные запросы).
 - * `QUERY: 1`: Был сделан один запрос.
 - * `ANSWER: 0`: Не было найдено ни одного ответа. Это подтверждает статус `NXDOMAIN`.
 - * `AUTHORITY: 1`: Была возвращена одна авторитетная запись (SOA-запись).
 - * `ADDITIONAL: 1`: Была возвращена одна дополнительная запись.
- `;; OPT PSEUDOSECTION:`: Этот раздел предоставляет информацию об опциях EDNS (Extension Mechanisms for DNS), используемых в запросе.
- `;; QUESTION SECTION:`: Этот раздел повторяет вопрос, который был задан: `client.mvchigladze.net. IN A`.
- `;; AUTHORITY SECTION:`: Этот раздел содержит запись Start of Authority (SOA) для домена `mvchigladze.net`. Запись SOA предоставляет информацию о DNS-зоне, включая основной сервер имен, ответственное лицо и различные таймеры.
- `;; Время запроса: 24 мсек`: Запрос занял 24 миллисекунды.

- ;; SERVER: 192.168.1.1#53(192.168.1.1) (UDP): Запрос был отправлен на DNS-сервер по адресу 192.168.1.1 на порт 53 с использованием UDP.
- ;; КОГДА: Пн Сен 15 18:49:19 UTC 2025: Время выполнения запроса.
- ;; РАЗМЕР СООБЩЕНИЯ получено: 122: Размер DNS-ответа составил 122 байта.

В итоге:

Команда **dig** была использована для запроса к DNS-серверу по адресу 192.168.1.1 для получения A-записи **client.mvchigladze.net**. DNS-сервер ответил **NXDOMAIN**, что означает, что у него нет записи для этого домена. Это означает, что DNS-сервер не настроен для разрешения имени **client.mvchigladze.net** в IP-адрес. Это может быть связано с неправильной конфигурацией, отсутствующей DNS-записью или тем, что домен просто не зарегистрирован или неправильно настроен в DNS-зоне.

3.4.6. Внесение изменений в настройки внутреннего окружения виртуальной машины

1. На виртуальной машине server перейдите в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/`, создайте в нём каталог `dhcp`, в который поместите в соответствующие подкаталоги конфигурационные файлы DHCP

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

Рис. 6.1. Переходим в каталог

3. Замените конфигурационные файлы DNS-сервера:

```
[root@server.mvchigladze.net ~]# cp -R /etc/named/* /vagrant
/provision/server/dns/etc/named/
cp: overwrite '/vagrant/provision/server/dns/etc/named/mvchi
gladze.net'? y
[root@server.mvchigladze.net ~]# y
bash: y: command not found...
[root@server.mvchigladze.net ~]# cd /vagrant/provision/serve
r
[root@server.mvchigladze.net server]# touch dhcp.sh
[root@server.mvchigladze.net server]# chmod +x dhcp.sh
[root@server.mvchigladze.net server]#
```

Рис. 6.2. Заменяем файлы

4. В каталоге `/vagrant/provision/server` создайте исполняемый файл `dhcp.sh`

```
[root@server.mvchigladze.net ~]# cp -R /etc/named/* /vagrant
/provision/server/dns/etc/named/
cp: overwrite '/vagrant/provision/server/dns/etc/named/mvchi
gladze.net'? y
[root@server.mvchigladze.net ~]# y
bash: y: command not found...
[root@server.mvchigladze.net ~]# cd /vagrant/provision/serve
r
[root@server.mvchigladze.net server]# touch dhcp.sh
[root@server.mvchigladze.net server]# chmod +x dhcp.sh
[root@server.mvchigladze.net server]#
```

Рис. 6.3. Создаем файл

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


```
70
71 server.vm.provider :virtualbox do |virtualbox|
72   virtualbox.customize ["modifyvm", :id, "--vrde", "on"]
73   virtualbox.customize ["modifyvm", :id, "--vrdeport", "3391"]
74 end
75
76 server.vm.provision "server dummy",
77   type: "shell",
78   preserve_order: true,
79   path: "provision/server/01-dummy.sh"
80
81 server.vm.provision "server dns",
82   type: "shell",
83   preserve_order: true,
84   path: "provision/server/dns.sh"
85
86 server.vm.provision "server dhcp",
87   type: "shell",
88   preserve_order: true,
89   path: "provision/server/dhcp.sh"
90
91
92 end
93
94 ## Client configuration
95 config.vm.define "client", autostart: false do |client|
96   client.vm.box = "rockylinux10"
```

length: 4 300 lines: 12 Ln: 88 Col: 46 Pos: 3 175 Windows (CR LF) UTF-8 IN

Рис. 6.4. Добавляем в раздел конфигурации

6. После этого виртуальные машины client и server можно выключить.

Вывод:

В ходе выполнения лабораторной работы были приобретены практические навыки по установке и конфигурированию DHCP-сервера.

Ответы на контрольные вопросы:

1. В каких файлах хранятся настройки сетевых подключений? - В наиболее популярных операционных системах, таких как Windows и Linux, настройки сетевых подключений хранятся в различных файлах:

В Windows, основные настройки сетевых подключений, такие как IP-адрес, маска подсети, шлюз, DNS-серверы и другие, хранятся в реестре. Однако, конфигурационные данные также могут быть сохранены в текстовых файлах, таких как ipconfig или в файле конфигурации подключения.

В Linux, настройки сети обычно хранятся в текстовых файлах в директории /etc/network/ или /etc/sysconfig/network-scripts/.

2. За что отвечает протокол DHCP? - Протокол DHCP (Dynamic Host Configuration Protocol) отвечает за автоматическое присвоение сетевых настроек устройствам в сети, таких как IP-адресов, маски подсети, шлюза, DNS-серверов и других параметров.
3. Поясните принцип работы протокола DHCP. Какими сообщениями обмениваются клиент и сервер, используя протокол DHCP? - Принцип работы протокола DHCP:

Discover (Обнаружение): Клиент отправляет в сеть запрос на обнаружение DHCP-сервера.

Offer (Предложение): DHCP-сервер отвечает клиенту, предлагая ему конфигурацию сети.

Request (Запрос): Клиент принимает предложение и отправляет запрос на использование предложенной конфигурации.

Acknowledgment (Подтверждение): DHCP-сервер подтверждает клиенту, что предложенная конфигурация принята и может быть использована.

4. В каких файлах обычно находятся настройки DHCP-сервера? За что отвечает каждый из файлов? - **Настройки DHCP-сервера обычно хранятся в файлах конфигурации, таких как:**

В Linux: /etc/dhcp/dhcpd.conf

В Windows: %SystemRoot%\System32\dhcp\dhcpd.conf

Они содержат информацию о диапазонах IP-адресов, параметрах сети и других опциях DHCP.

5. Что такое DDNS? Для чего применяется DDNS? - **DDNS (Dynamic Domain Name System) - это система динамического доменного имени. Она используется для автоматического обновления записей DNS, когда IP-адрес узла изменяется. DDNS применяется, например, в домашних сетях, где IP-адреса часто изменяются посредством DHCP.**
6. Какую информацию можно получить, используя утилиту ifconfig? Приведите примеры с использованием различных опций. - **Утилита ifconfig используется для получения информации о сетевых интерфейсах.**

Примеры:

ifconfig: Показывает информацию обо всех активных сетевых интерфейсах.

ifconfig eth0: Показывает информацию о конкретном интерфейсе (в данном случае, eth0).

7. Какую информацию можно получить, используя утилиту ping? Приведите примеры с использованием различных опций. - Утилита ping используется для проверки доступности узла в сети.

Примеры:

ping google.com: Пингует домен google.com.

ping -c 4 192.168.1.1: Пингует IP-адрес 192.168.1.1 и отправляет 4 эхо-запроса.