**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**

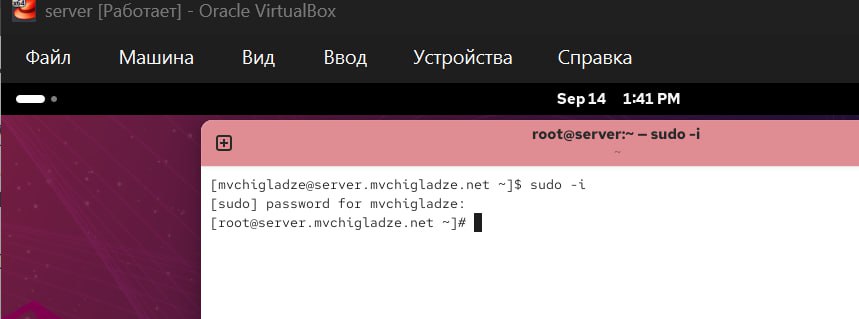
**Установка DHCP-сервера**

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

cd /var/tmp/mvchigladze/vagrant

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

make server-up

****

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

1. 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

****

**Рис. 1.2.** Переход в режим суперпользователя и установка dhcp.

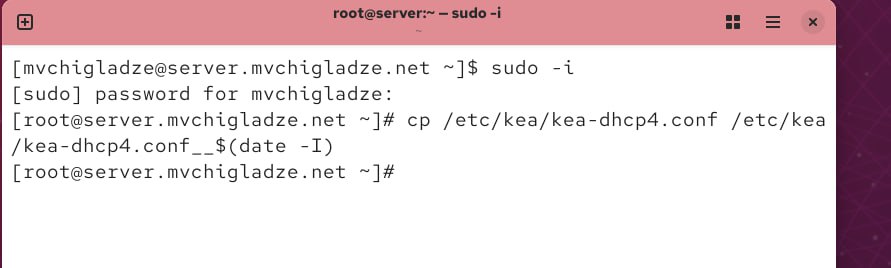
**Конфигурирование DHCP-сервера**

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

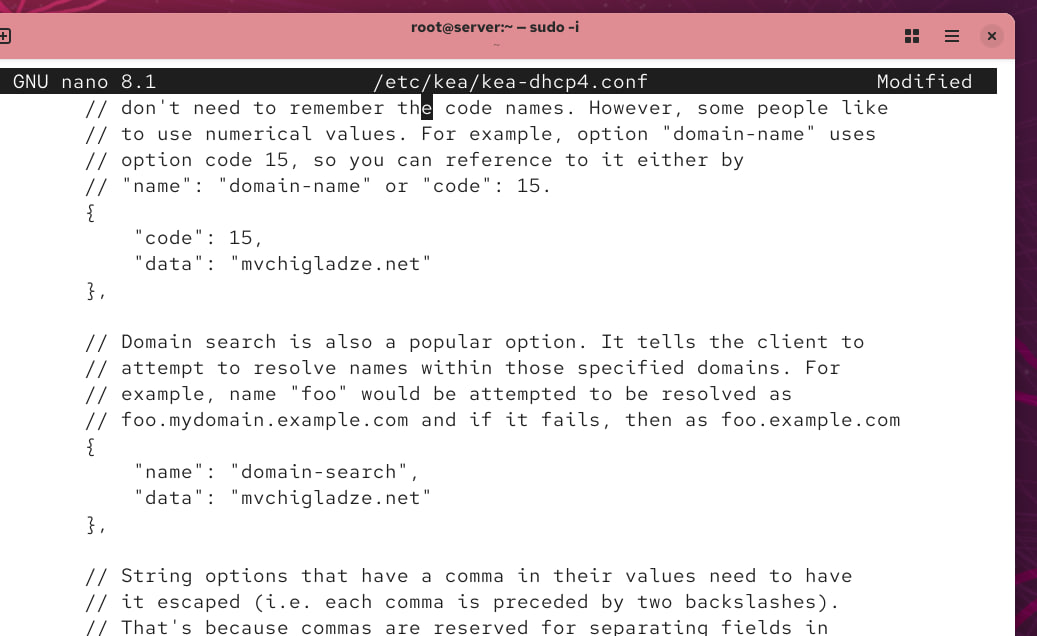
mv /etc/dhcp/dhcpd.conf.example /etc/dhcp/dhcpd.conf

****

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

1. 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).

****

**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.

1. Настройте привязку dhcpd к интерфейсу eth1 виртуальной машины server



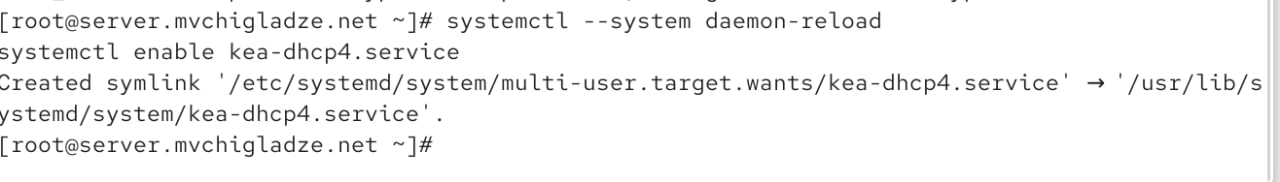
**Рис. 2.3**. Настройка привзяки

1. Проверьте правильность конфигурационного файла:

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



**Рис. 2.4**. Проверяем правильность

1. Перезагрузите конфигурацию dhcpd и разрешите загрузку DHCP-сервера при запуске виртуальной машины server

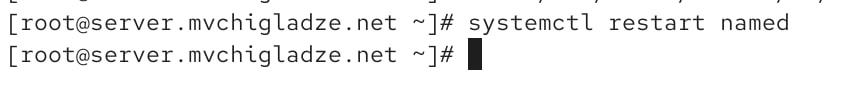
**Рис. 2.5**. Перезагрузка конфигурации

1. Добавьте запись для DHCP-сервера в конце файла прямой DNS-зоны



**Рис. 2.6**. Добавляем запись

1. Перезапустите named



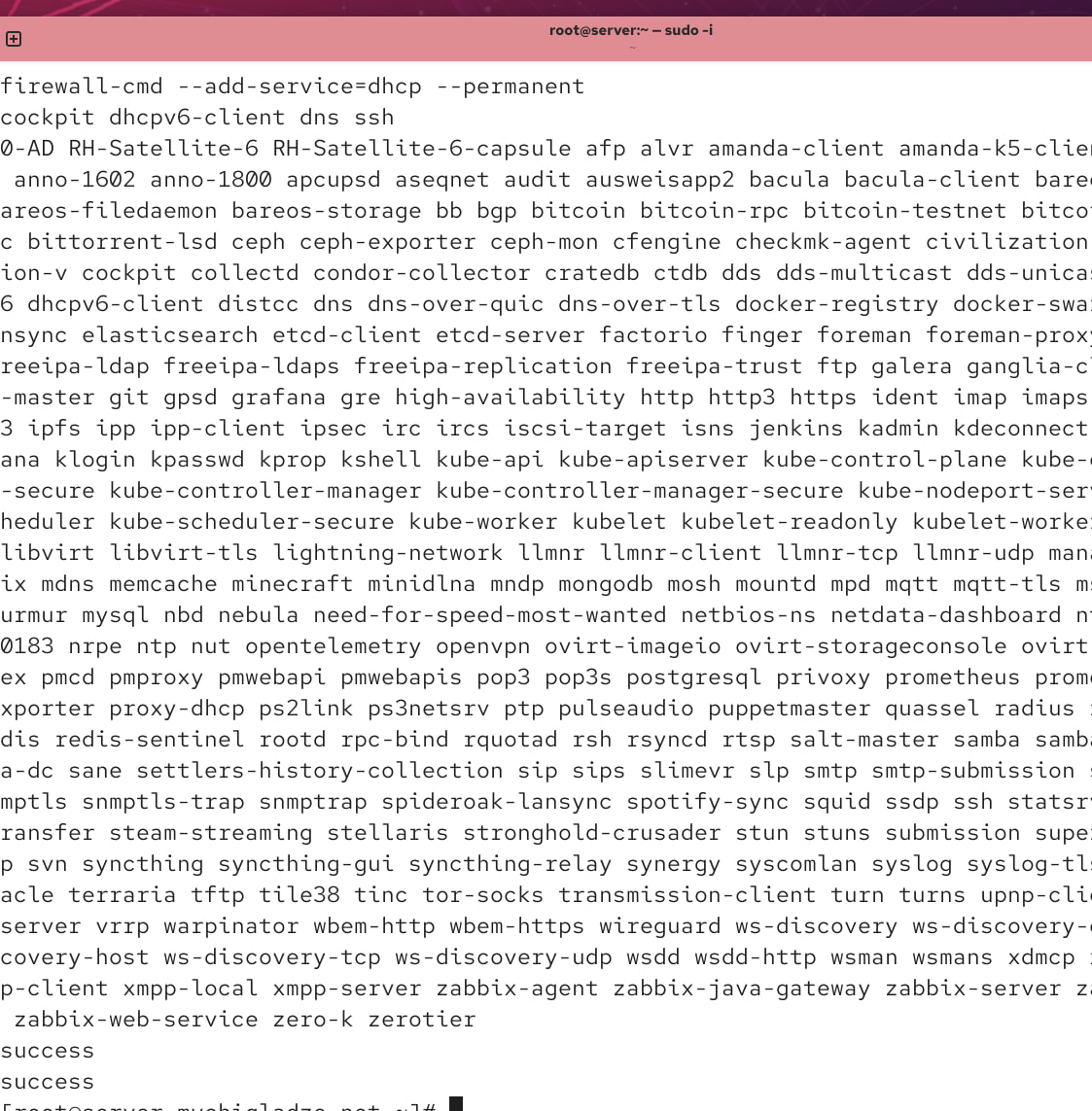
**Рис. 2.7**. Перезапускаем

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



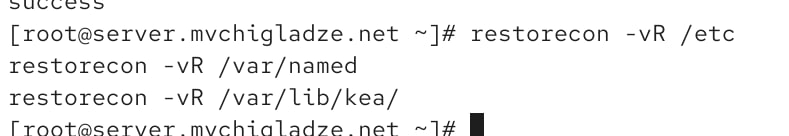
**Рис. 2.8**. Проверяем

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



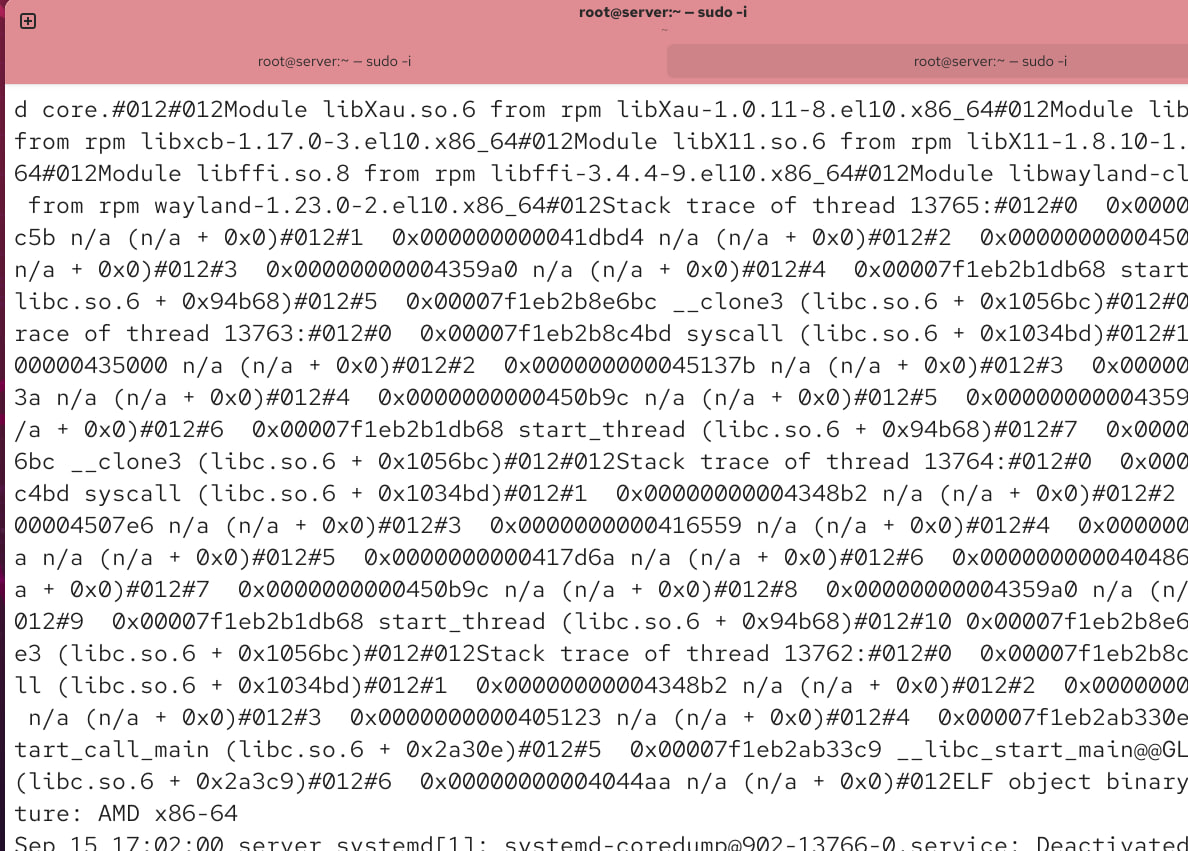
**Рис. 2.9**. Вносим изменения

1. Восстановите контекст безопасности в SELinux



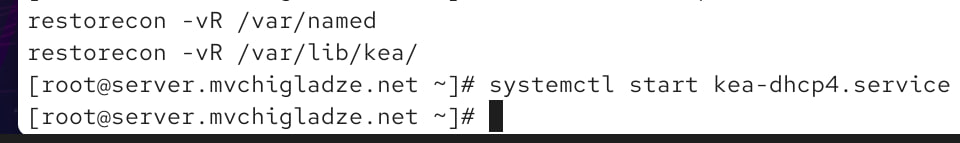
**Рис. 2.10**. Восстанавливаем контекст

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



**Рис. 2.11**. Запускаем мониторинг

1. В основном рабочем терминале запустите DHCP-сервер

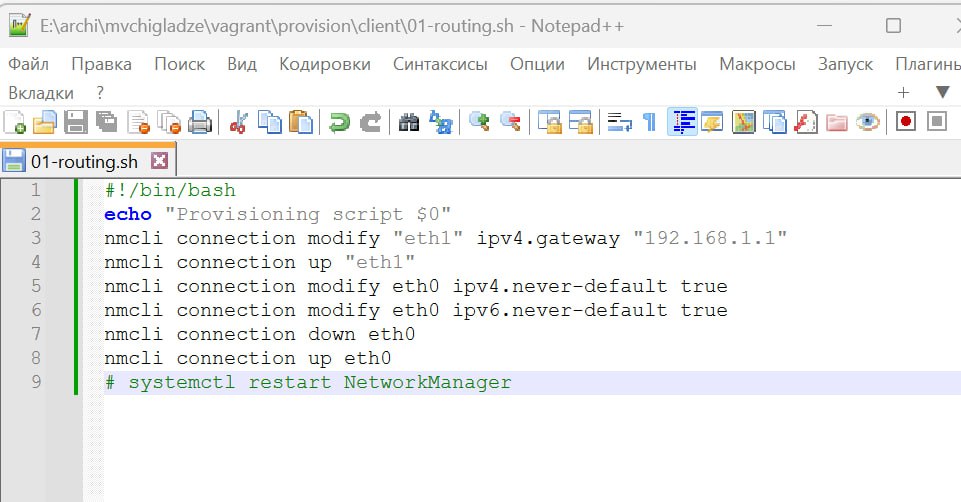


**Рис. 2.12**. Запускаем сервер

1. 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



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

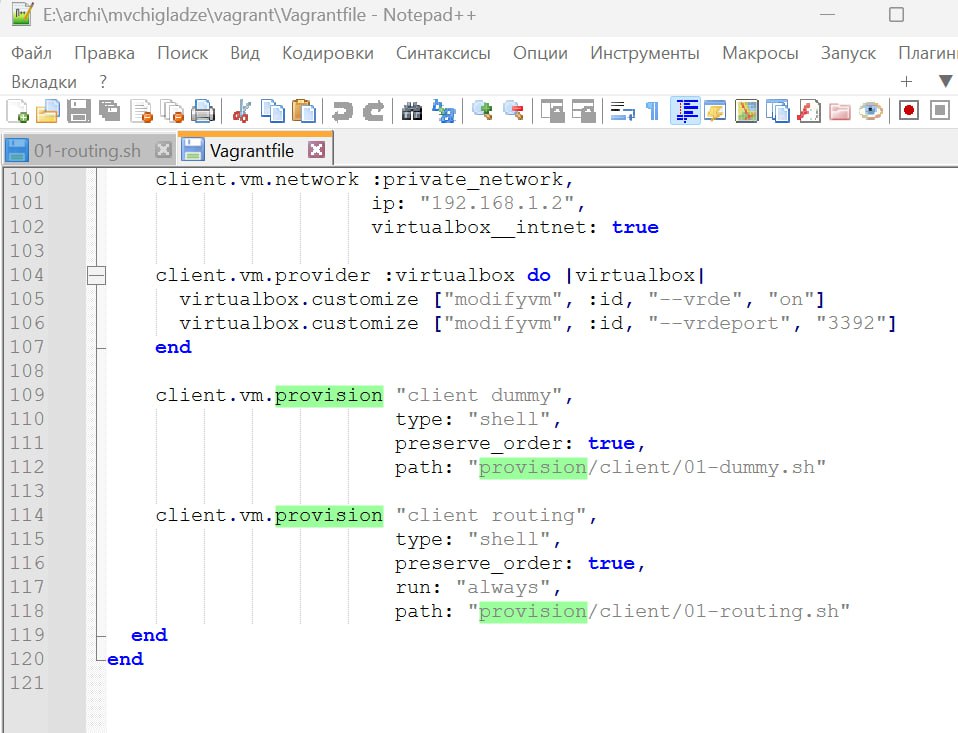
client.vm.provision "client routing",

type: "shell",

preserve\_order: true,

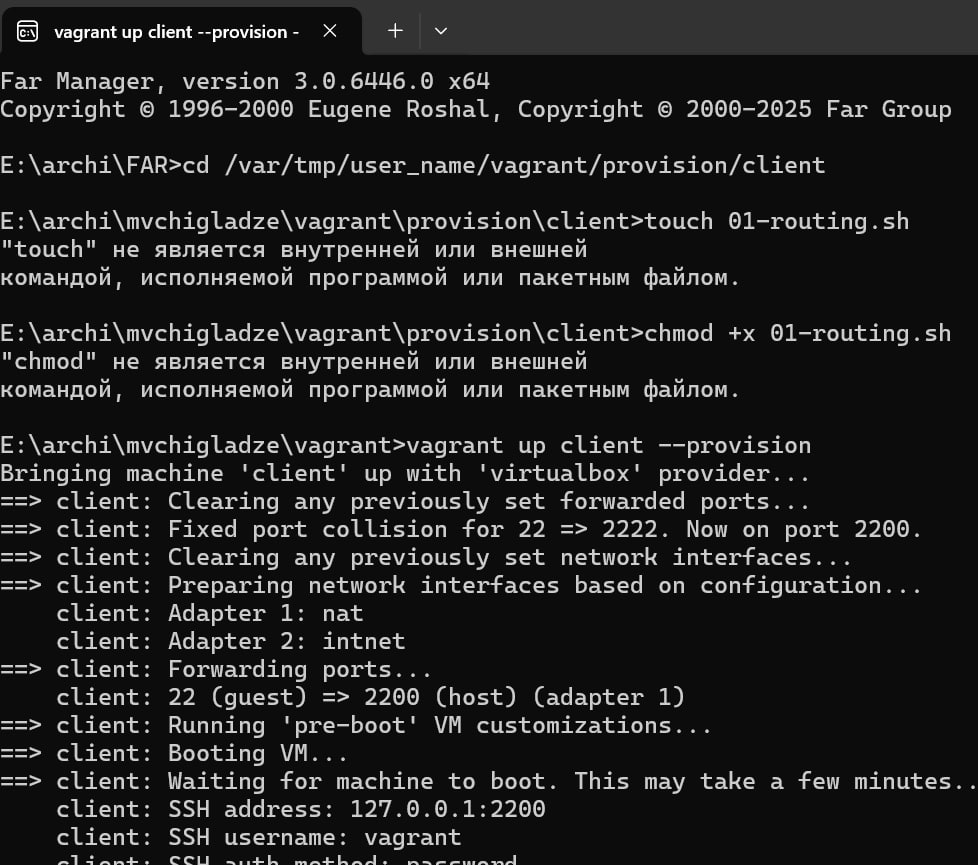
run: "always",

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



**Рис. 3.2**. Подключаем скрипт

1. Зафиксируйте внесённые изменения для внутренних настроек виртуальной машины client и запустите её, введя в терминале

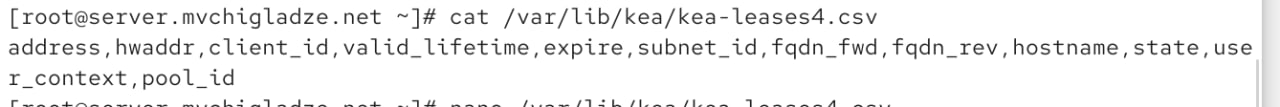


**Рис. 3.3**. Фиксируем внесенные изменения

1. 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.
2. Войдите в систему виртуальной машины client под вашим пользователем и откройте терминал.
3. На машине server посмотрите список выданных адресов

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

Прокомментируйте построчно в отчёте.



**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.

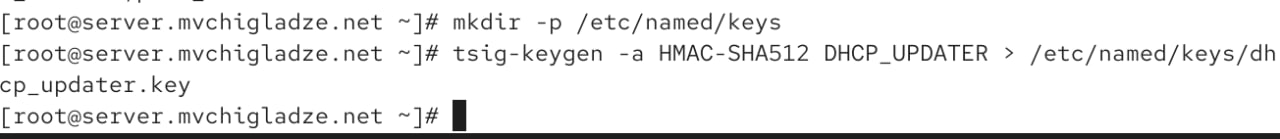
**3.4.4. Настройка обновления DNS-зоны**

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

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

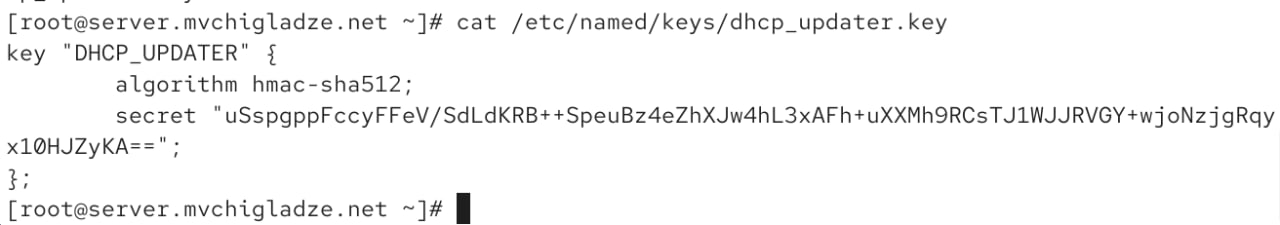
mkdir -p /etc/named/keys

tsig-keygen -a HMAC-SHA512 DHCP\_UPDATER > /etc/named/keys/dhcp\_updater.key



**Рис. 4.1**. Создаем ключ

1. Файл /etc/named/keys/dhcp\_updater.key будет иметь следующий вид



**Рис. 4.2**. Файл апдейтер

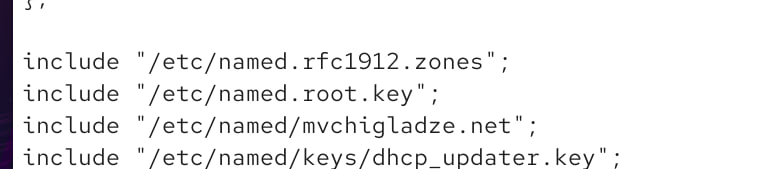
3. Поправим права доступа:

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



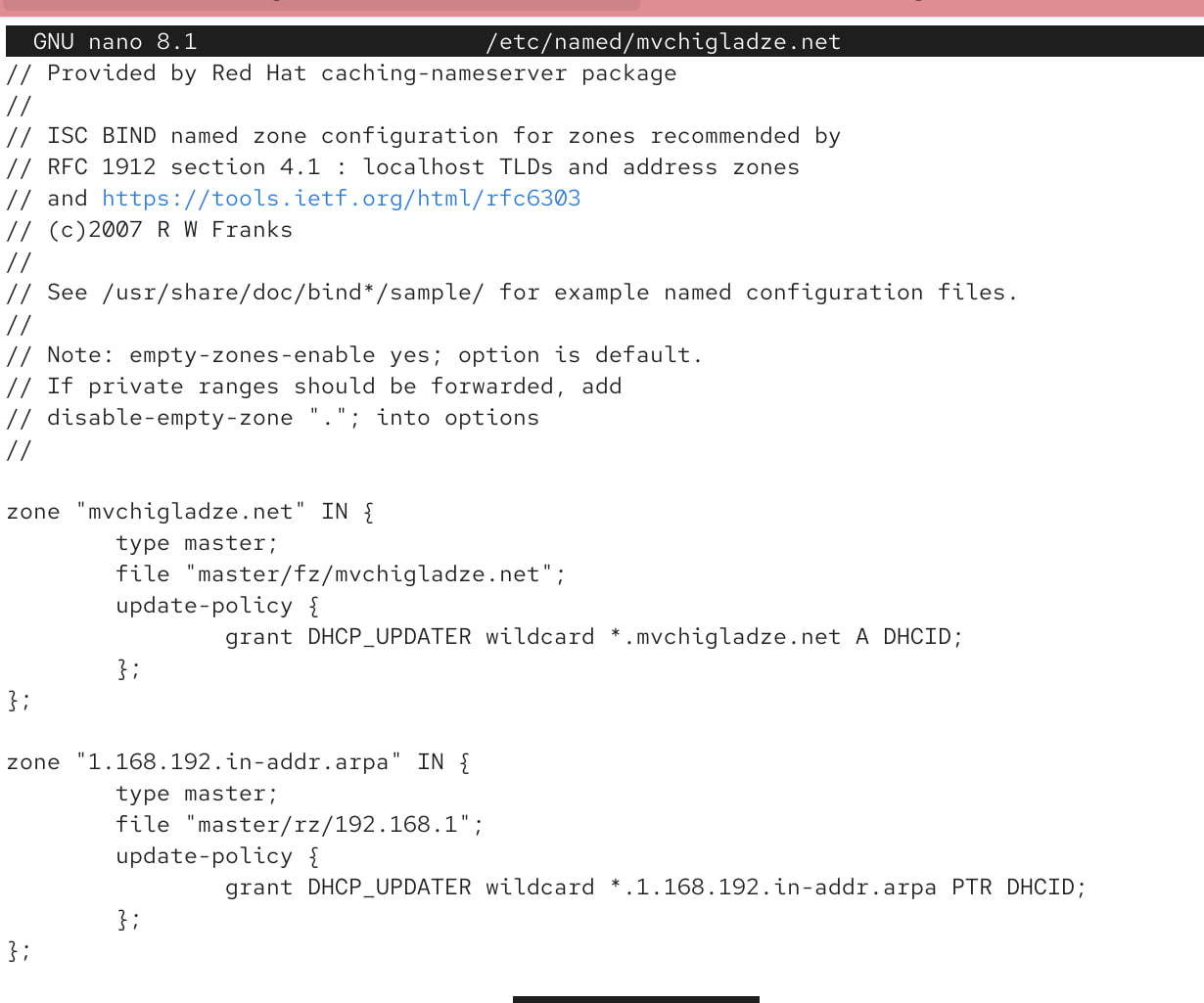
**Рис. 4.3**. Права

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



**Рис. 4.4**. Ключ

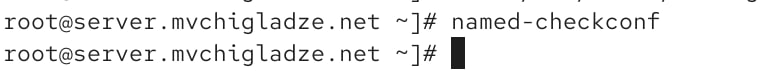
1. На виртуальной машине server под пользователем с правами суперпользователя отредактируйте файл /etc/named/user.net (вместо user укажите свой логин), разрешив обновление зоны



**Рис. 4.5**. Редактируем файл

6. Сделаем проверку конфигурационного файла:

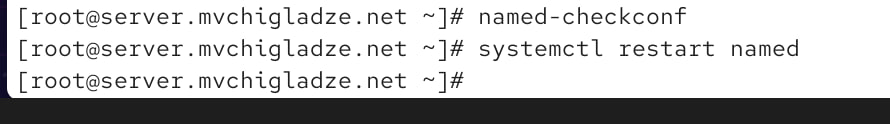
named-checkconf



**Рис. 4.6**. Проверка конфигурационного файла

7. Перезапустите DNS-сервер:

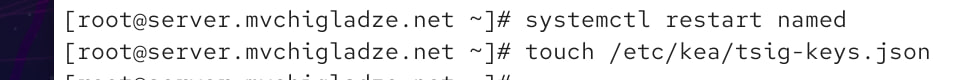
systemctl restart named



**Рис. 4.7**. Перезаупск сервера

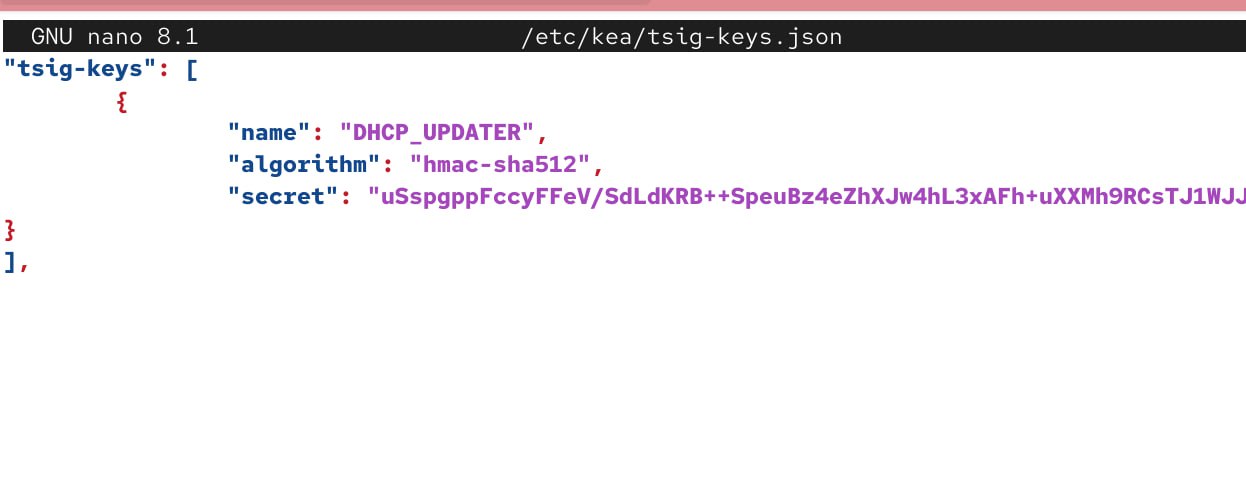
8. Сформируем ключ для Kea. Файл ключа назовём /etc/kea/tsig-keys.json:

touch /etc/kea/tsig-keys.json



**Рис. 4.8**. Формируем ключ

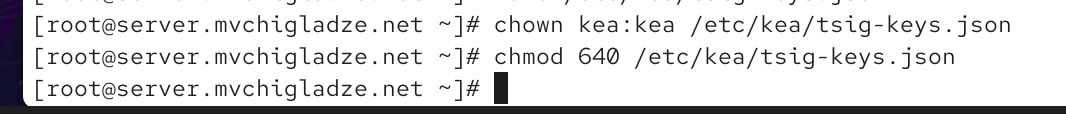
1. Перенесём ключ на сервер Kea DHCP и перепишем его в формате json



**Рис. 4.9**. Переносим ключ

10. Сменим владельца:

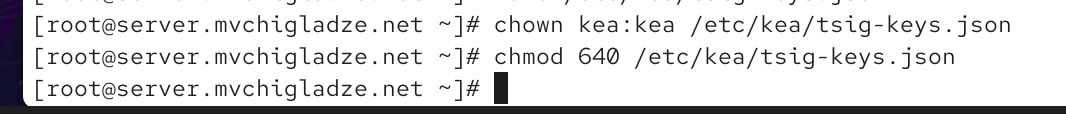
chown kea:kea /etc/kea/tsig-keys.json



**Рис. 4.10**. Меняем владельца

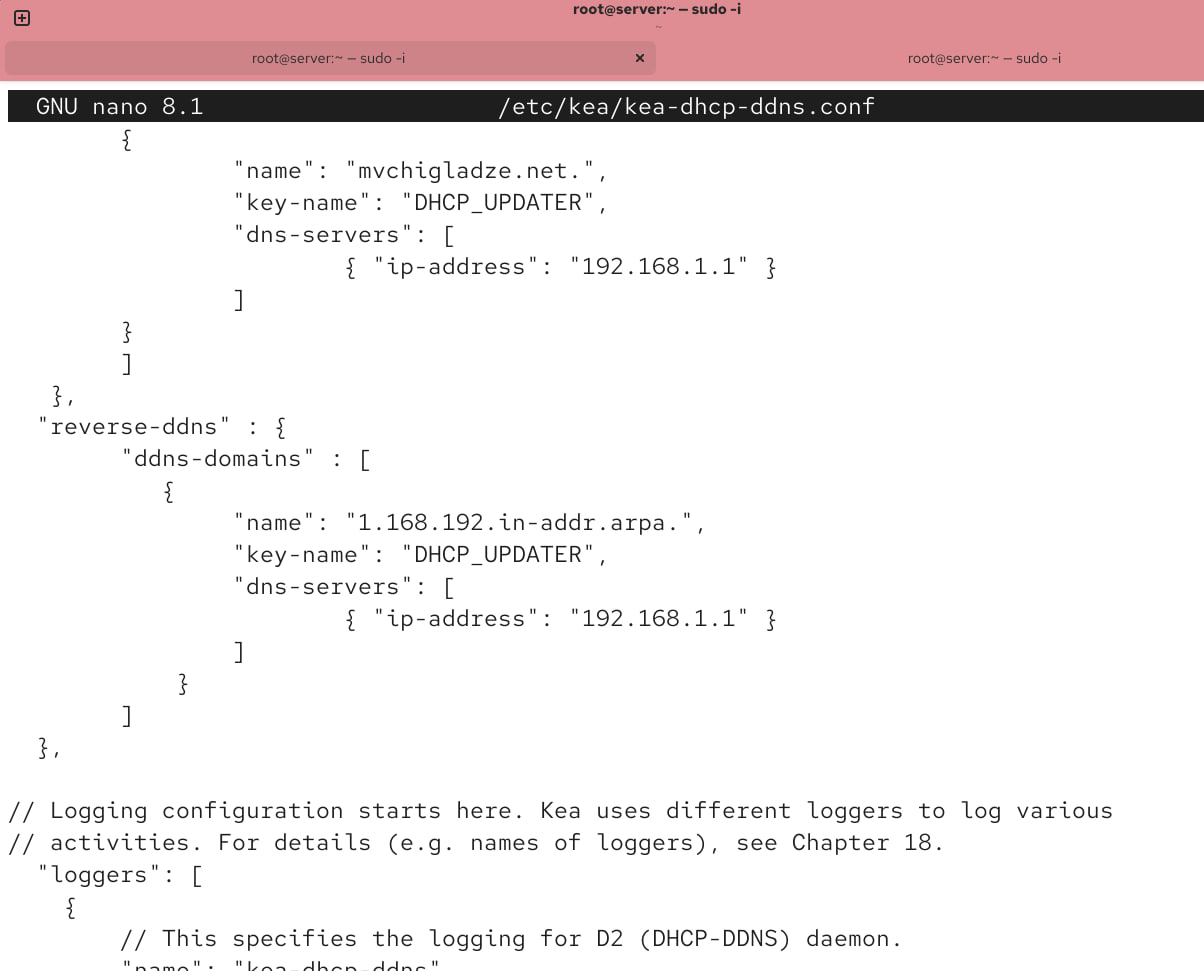
11. Поправим права доступа:

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



**Рис. 4.11**. Правим права доступа

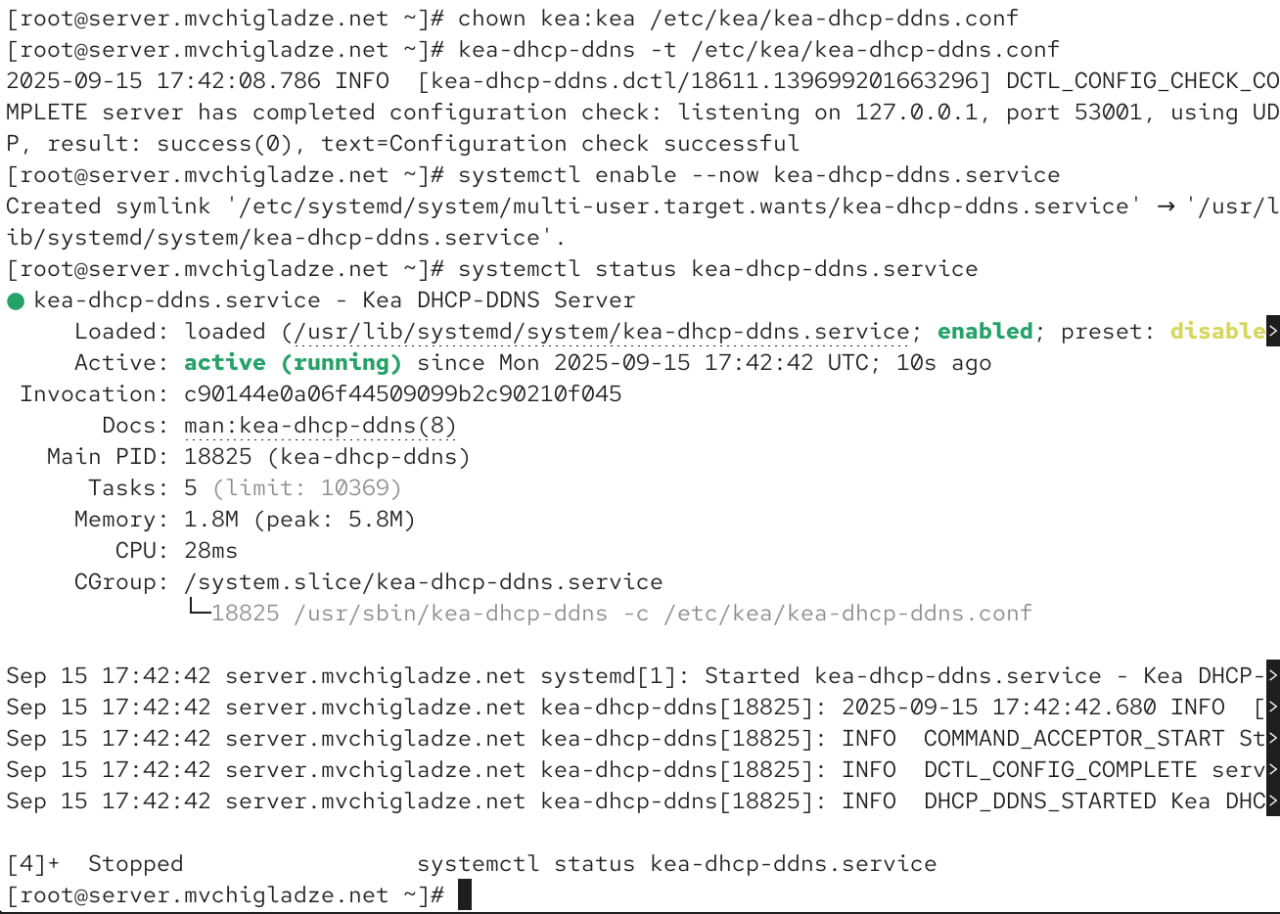
1. Настройка происходит в файле /etc/kea/kea-dhcp-ddns.conf



**Рис. 4.12**. Настройка в файле

13. Изменим владельца файла:

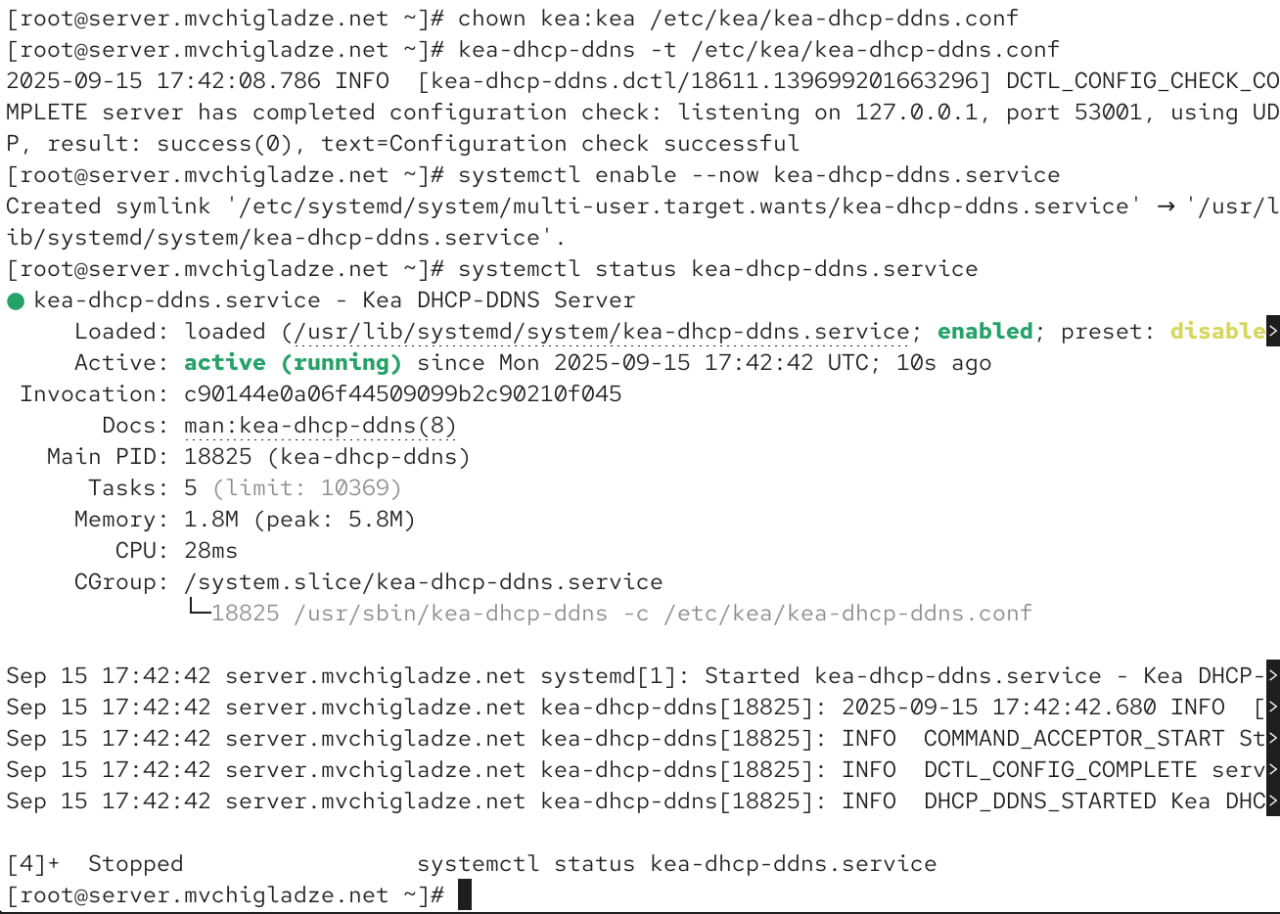
chown kea:kea /etc/kea/kea-dhcp-ddns.conf



**Рис. 4.13**. Меняем владельца

14. Проверим файл на наличие возможных синтаксических ошибок:

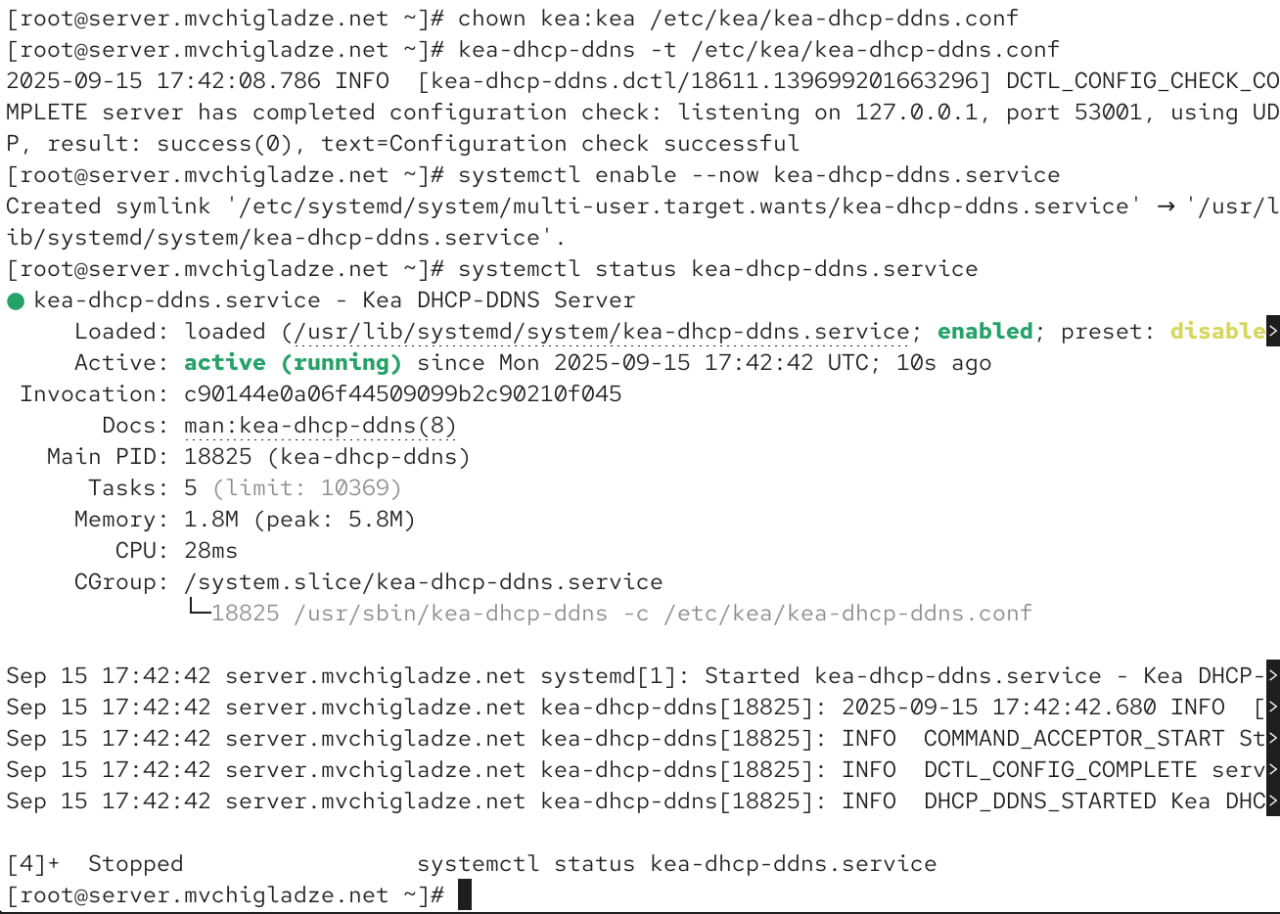
kea-dhcp-ddns -t /etc/kea/kea-dhcp-ddns.conf



**Рис. 4.14**. Проверяем ошибки

15. Запустим службу ddns:

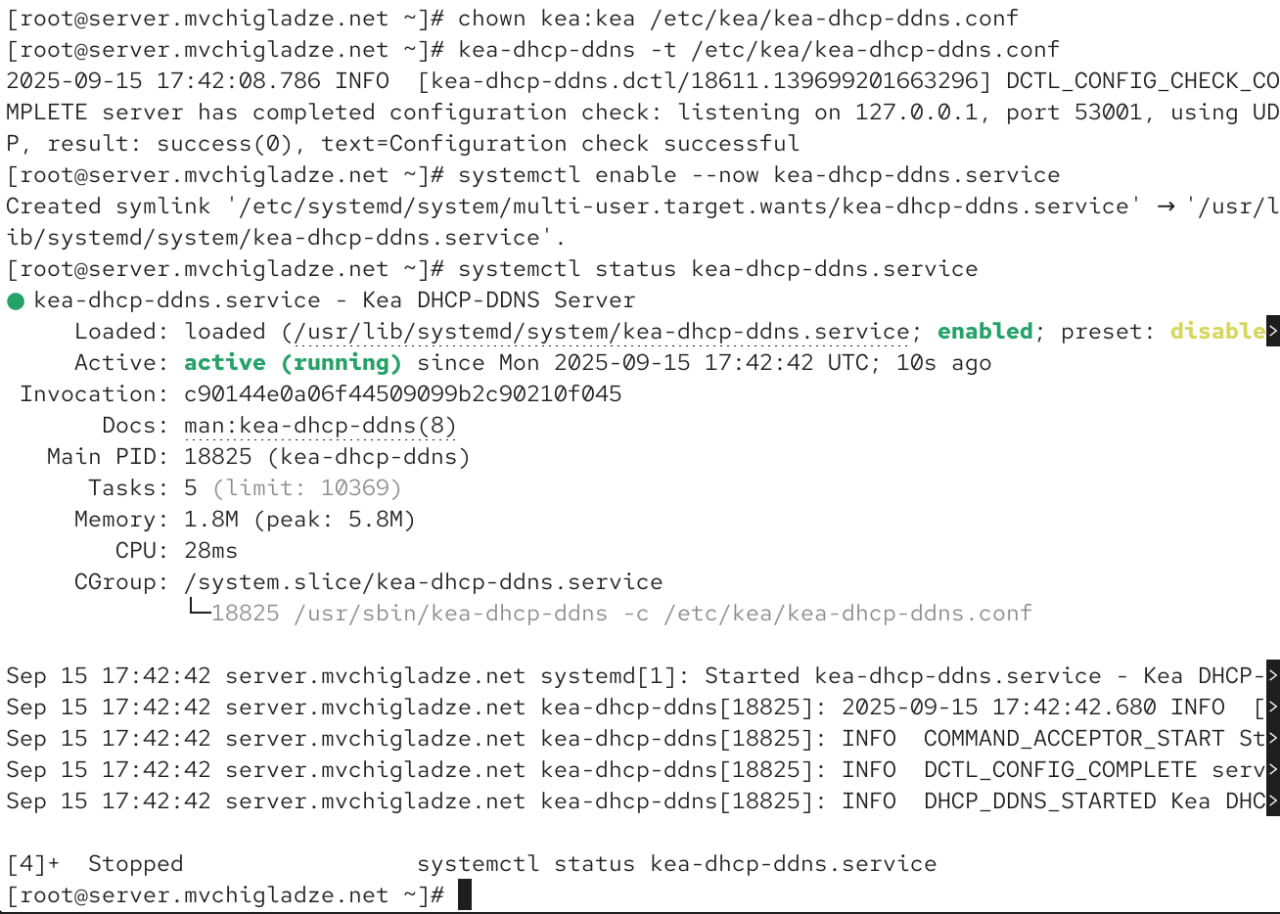
systemctl enable --now kea-dhcp-ddns.service



**Рис. 4.15**. Запускаем службу

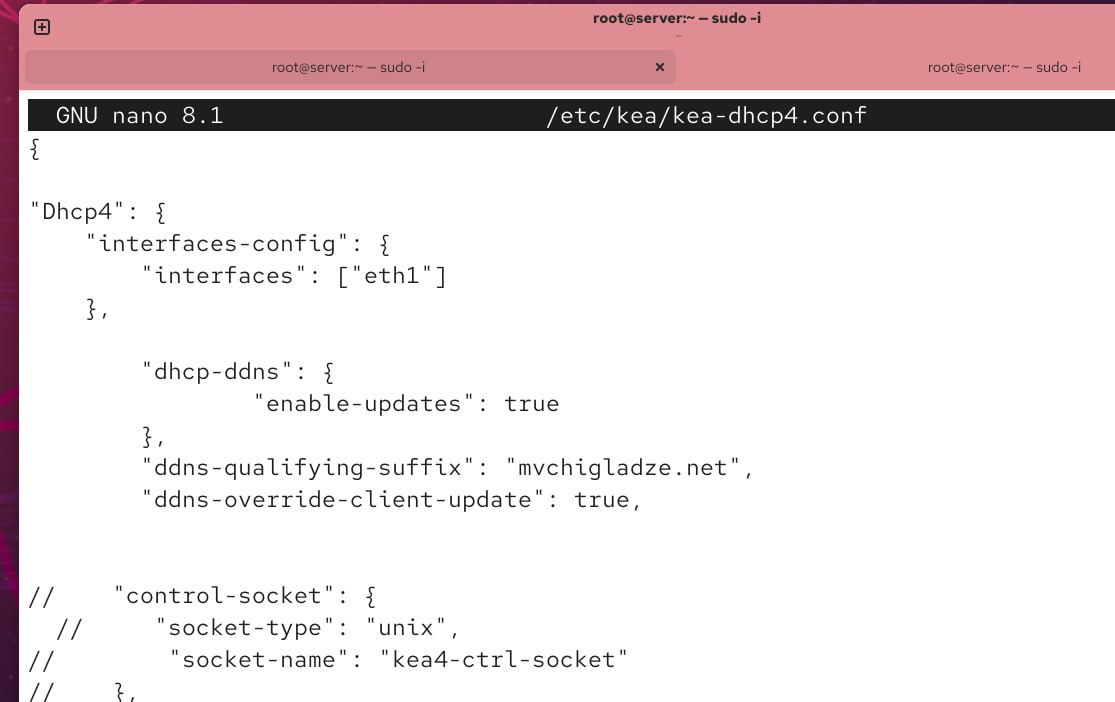
16. Проверим статус работы службы:

systemctl status kea-dhcp-ddns.service



**Рис. 4.16**. Проверяем статус службы

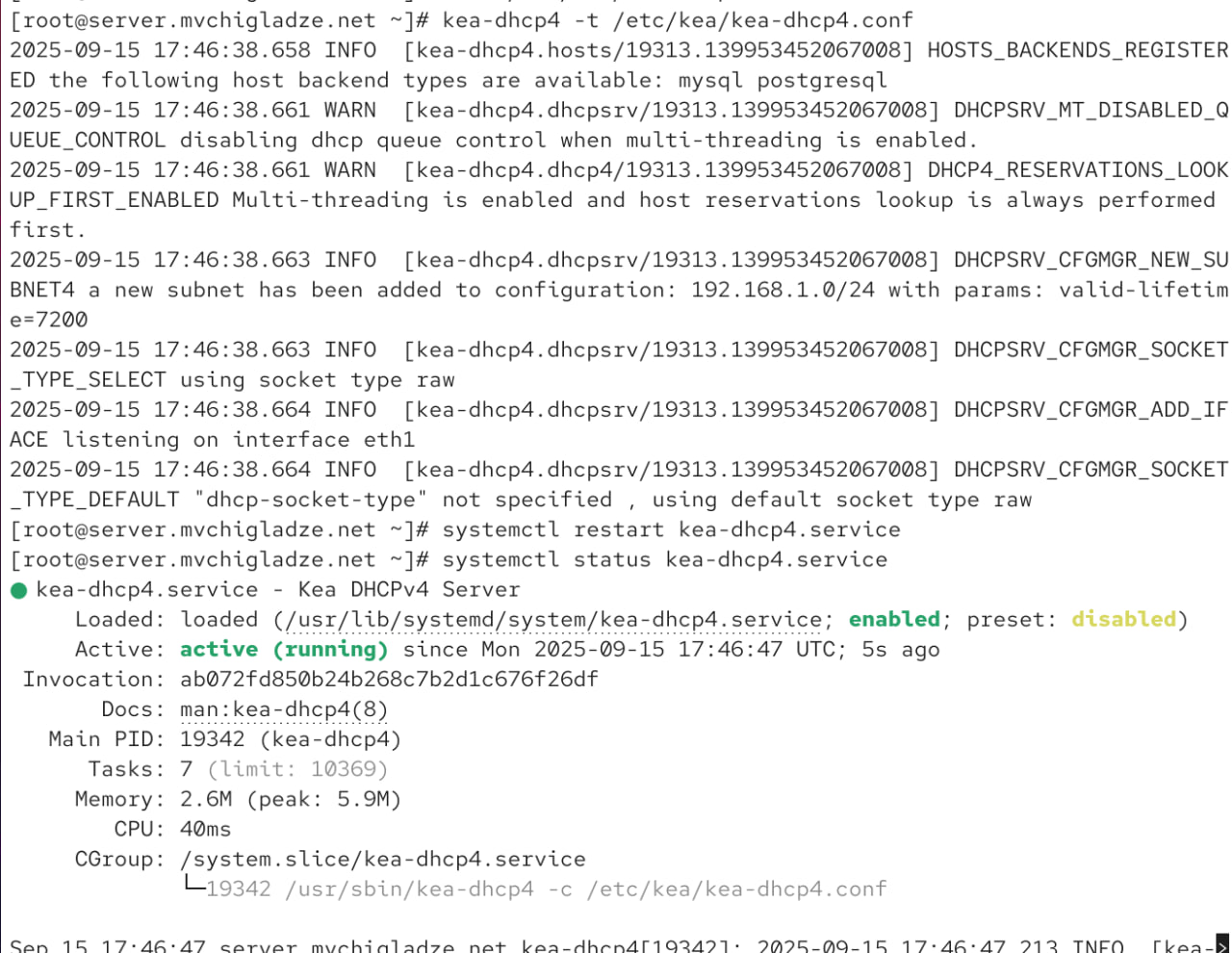
1. Внесите изменения в конфигурационный файл /etc/kea/kea-dhcp4.conf, добавив в него разрешение на динамическое обновление DNS-записей с локального узла прямой и обратной зон



**Рис. 4.17**. Вносим изменения в файл

18. Проверим файл на наличие возможных синтаксических ошибок:

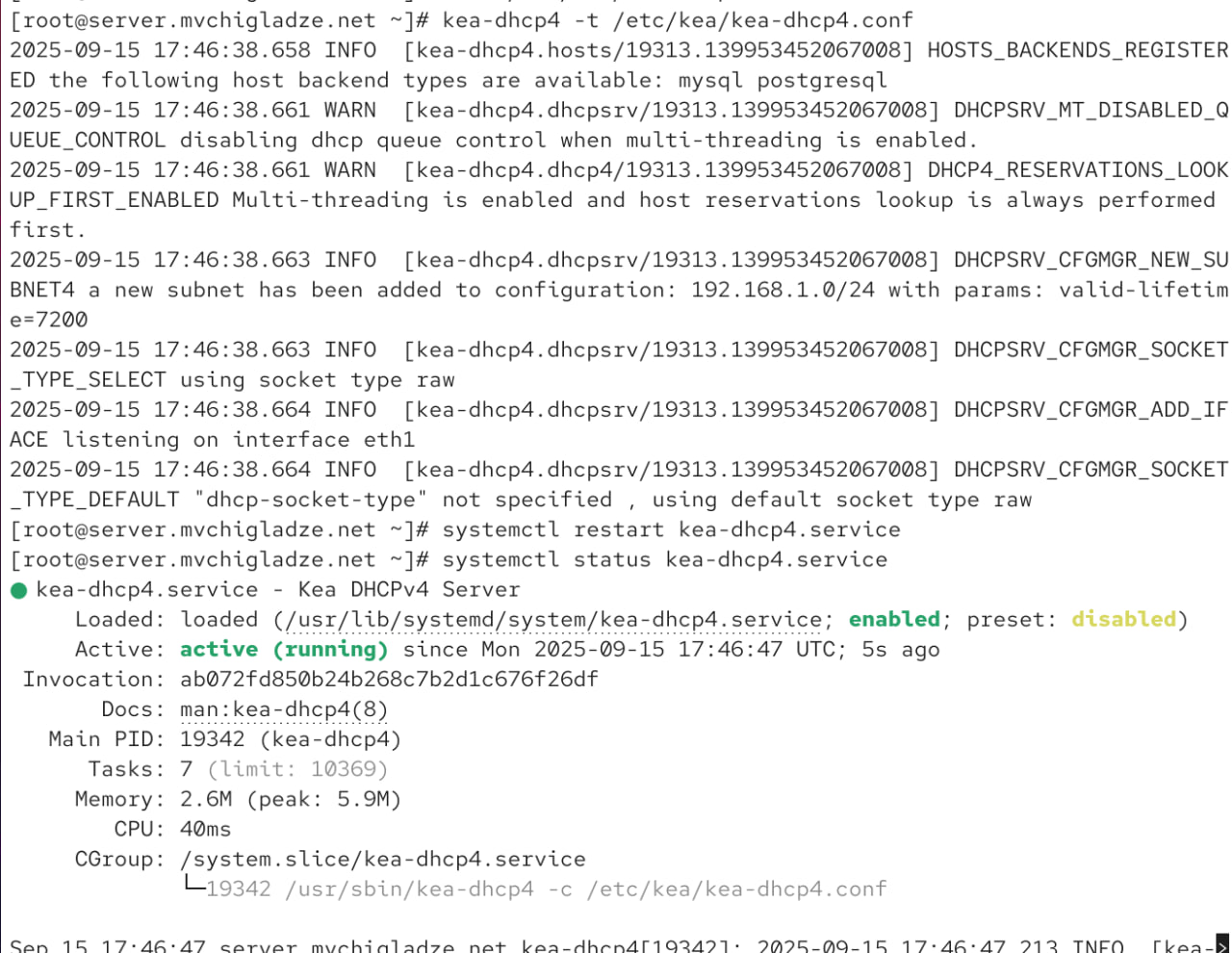
kea-dhcp4 -t /etc/kea/kea-dhcp4.conf



**Рис. 4.18**. Проверяем файл на наличие ошибок

19. Перезапустите DHCP-сервер:

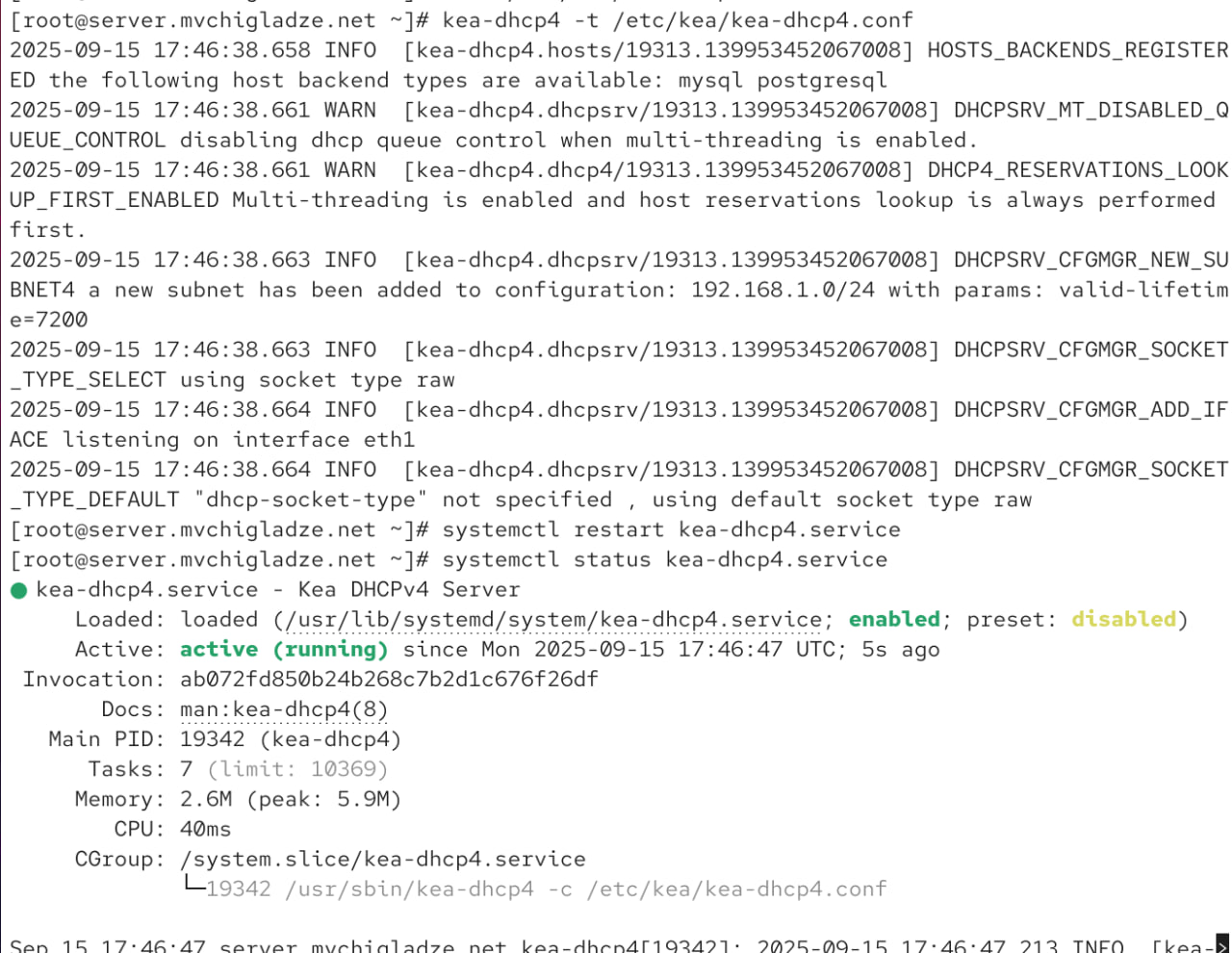
systemctl restart kea-dhcp4.service



**Рис. 4.19**. Перезапуск

20. Проверим статус:

systemctl status kea-dhcp4.service

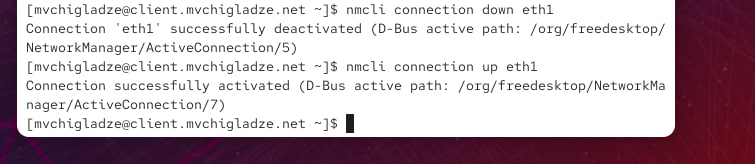


**Рис. 4.20**. Проверяем статус

21. На машине client переполучите адрес:

nmcli connection down eth1

nmcli connection up eth1



**Рис. 4.21**. Переполучаем адрес

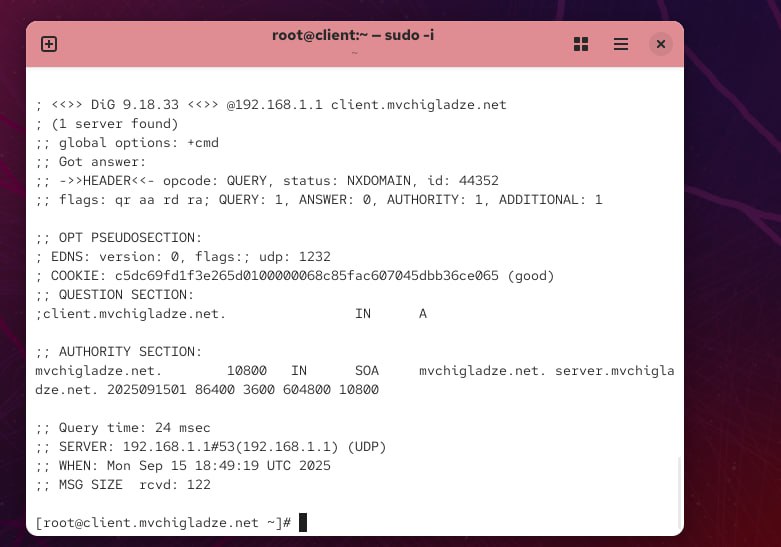
1. В каталоге прямой DNS-зоны /var/named/master/fz должен появиться файл user.net.jnl, в котором в бинарном файле автоматически вносятся изменения записей зоны.

**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.



**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.

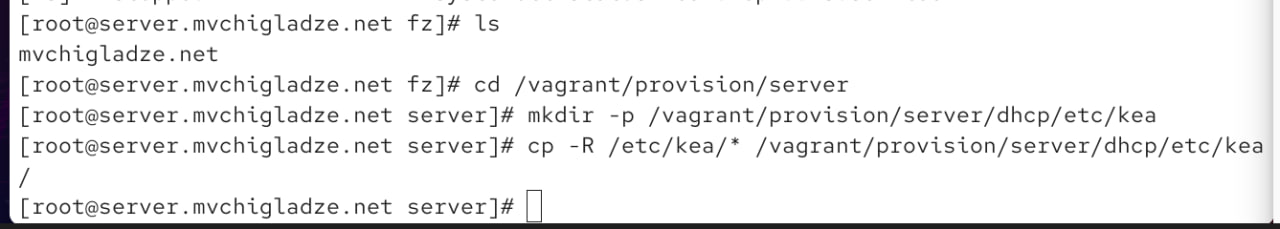
•   ;; РАЗМЕР СООБЩЕНИЯ получено: 122: Размер DNS-ответа составил 122 байта.

**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



**Рис. 6.1**. Переходим в каталог

1. Замените конфигурационные файлы DNS-сервера:



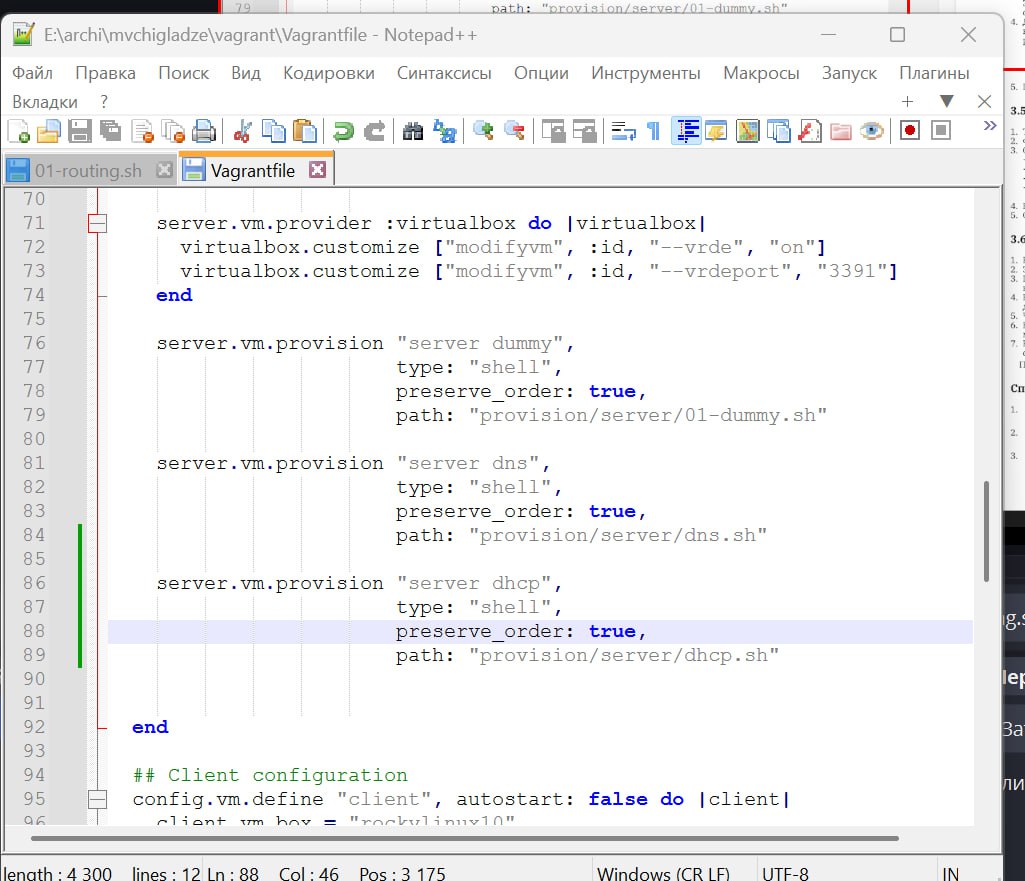
**Рис. 6.2**. Заменяем файлы

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



**Рис. 6.3**. Создаем файл

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



**Рис. 6.4**. Добавляем в раздел конфигурации

1. 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/.**

1. 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.**
2. 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.**

1. 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.**

1. 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.**
2. 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).**

1. 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.**