Администрирование сетевых подсистем

Лабораторная работа №3

Тойчубекова Асель Нурлановна

Содержание

# 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 вм server. ([рис. 1](#fig-001)).

|  |
| --- |
| Рисунок 1: Запуск server |

На вм перейдем в режим суперпользователя и установим dhcp. ([рис. 2](#fig-002)).

|  |
| --- |
| Рисунок 2: Установка dhcp |

Сохраним на всякий случай конфигурационный файл. ([рис. 3](#fig-003)).

|  |
| --- |
| Рисунок 3: Сохранение конф файла |

Откроем файл /etc/kea/kea-dhcp4.conf на редактирование. В этом файле заменим шаблон для domain-name. ([рис. 4](#fig-004) и [рис. 5](#fig-005) и [рис. 6](#fig-006) ).

|  |
| --- |
| Рисунок 4: Редактирование kea-dhcp4.conf |

|  |
| --- |
| Рисунок 5: Редактирование kea-dhcp4.conf |

|  |
| --- |
| Рисунок 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-адре. ([рис. 7](#fig-007)).

|  |
| --- |
| Рисунок 7: Редактирование kea-dhcp4.conf |

Настроим привязку dhcpd к интерфейсу eth1 виртуальной машины server. ([рис. 8](#fig-008)).

|  |
| --- |
| Рисунок 8: Настройка привязки dhcp |

Проверим правильность конфигурационного файла. Мы видим, что все корректно отрабатывается. ([рис. 9](#fig-009)).

|  |
| --- |
| Рисунок 9: Проверка работы конф файла |

Перезагрузим конфигурацию dhcpd и разрешим загрузку DHCP-сервера при запуске виртуальной машины server. ([рис. 10](#fig-010)).

|  |
| --- |
| Рисунок 10: Перезагрузка конф dhcpd |

Добавим запись для DHCP-сервера в конце файла прямой DNS-зоны /var/named/master/fz/antoychubekova.net. ([рис. 11](#fig-011)).

|  |
| --- |
| Рисунок 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. 12](#fig-012)).

|  |
| --- |
| Рисунок 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. 13](#fig-013)).

|  |
| --- |
| Рисунок 13: Перезапуск named и проверка обращения к DHCP |

Внесиv изменения в настройки межсетевого экрана узла server, разрешив работу с DHCP. ([рис. 14](#fig-014) и [рис. 15](#fig-015) ).

|  |
| --- |
| Рисунок 14: Настройка межсетевого узла server |

|  |
| --- |
| Рисунок 15: Настройка межсетевого узла server |

Восстановим контекст безопасности в SELinux. ([рис. 16](#fig-016)).

|  |
| --- |
| Рисунок 16: Восстановление контекста безопасности |

В дополнительном терминале запустим мониторинг происходящих в системе процессов в реальном времени. ([рис. 17](#fig-017)).

|  |
| --- |
| Рисунок 17: Мониторинг процессов в системе |

В основном рабочем терминале запустим DHCP-сервер. ([рис. 18](#fig-018)).

|  |
| --- |
| Рисунок 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. 19](#fig-019)).

|  |
| --- |
| Рисунок 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. 20](#fig-020)).

|  |
| --- |
| Рисунок 20: Редактирование Vagrantfie |

Зафиксируем внесённые изменения для внутренних настроек виртуальной машины client и запустите её. ([рис. 21](#fig-021)).

|  |
| --- |
| Рисунок 21: Запуск clien |

После загрузки виртуальной машины client мы видим информацию о работе DHCP-сервера в файле /var/lib/kea/kea-leases4.csv. ([рис. 22](#fig-022)).

|  |
| --- |
| Рисунок 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).

Войдим в систему виртуальной машины client под пользователем и откройте терминал. Используя комнаду ifconfig выведем на экран информацию об имеющихся интерфейсах. ([рис. 23](#fig-023)).

|  |
| --- |
| Рисунок 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. 24](#fig-024)).

|  |
| --- |
| Рисунок 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. 25](#fig-025)).

|  |
| --- |
| Рисунок 25: Создание ключа на сервере с Bind9 |

Файл /etc/named/keys/dhcp\_updater.key выглядит следующим образом. ([рис. 26](#fig-026)).

|  |
| --- |
| Рисунок 26: Ключ на сервере с Bind9 |

Поправим права доступа. ([рис. 27](#fig-027)).

|  |
| --- |
| Рисунок 27: Изменение прав доступа |

Подключим ключ в файле /etc/named.conf. ([рис. 28](#fig-028)).

|  |
| --- |
| Рисунок 28: Подключение ключ |

На виртуальной машине server под пользователем с правами суперпользователя отредактируем файл /etc/named/antoychubekova.net, разрешив обновление зоны. ([рис. 29](#fig-029)).

|  |
| --- |
| Рисунок 29: Разрешение обновление зоны |

Сделаем проверку конфигурационного файла и перезапустим DNS-сервер. ([рис. 30](#fig-030)).

|  |
| --- |
| Рисунок 30: Проверка конф файла и перезапуск DNS |

Сформируем ключ для Kea. Файл ключа назовём /etc/kea/tsig-keys.json. Перенесём ключ на сервер Kea DHCP(с файла /etc/named/keys/dhcp\_updater.key ) и перепишем его в формате json.([рис. 31](#fig-031)).

|  |
| --- |
| Рисунок 31: Формирование ключа для Kea |

Сменим владельца и поправим права доступа ([рис. 32](#fig-032)).

|  |
| --- |
| Рисунок 32: Редактирование владельца и прав доступа |

Настройка происходит в файле /etc/kea/kea-dhcp-ddns.conf. ([рис. 33](#fig-033)).

|  |
| --- |
| Рисунок 33: Настройка DHCP,DNS |

Изменим владельца файла и проверим файл на наличие возможных синтаксических ошибок. ([рис. 34](#fig-034)).

|  |
| --- |
| Рисунок 34: Проверка на наличие синтаксических ошибок |

Запустим службу ddns и проверим статус работы службы, она имеет статус running. ([рис. 35](#fig-035)).

|  |
| --- |
| Рисунок 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. 36](#fig-036)).

|  |
| --- |
| Рисунок 36: Редактирование конф файла kea-dhcp4.conf |

Проверим файл на наличие возможных синтаксических ошибок. Все корректно отрабатывается. ([рис. 37](#fig-037)).

|  |
| --- |
| Рисунок 37: Проверка файла на ошибки |

Перезапустим DHCP-сервер и проверим статус. ([рис. 38](#fig-038)).

|  |
| --- |
| Рисунок 38: Запуск DHCP и проверка статуса |

На машине client переполучим адрес. ([рис. 39](#fig-039)).

|  |
| --- |
| Рисунок 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. 40](#fig-040)).

|  |
| --- |
| Рисунок 40: Файл antoychubekova.net.jnl |

На виртуальной машине client под наши 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. 41](#fig-041)).

|  |
| --- |
| Рисунок 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 выполнился 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. 42](#fig-042)).

|  |
| --- |
| Рисунок 42: Создание каталога и покаталогов |

Заменим конфигурационные файлы DNS-сервера. ([рис. 43](#fig-043)).

|  |
| --- |
| Рисунок 43: Редактирование конфигурационных файлов DNS-сервера |

В каталоге /vagrant/provision/server создадим исполняемый файл dhcp.sh. ([рис. 44](#fig-044)).

|  |
| --- |
| Рисунок 44: Создание исполняемого файла |

В нем напишем скрипт, по сути, повторяющий произведённые нами действия по установке и настройке DHCP-сервера. ([рис. 45](#fig-045)).

|  |
| --- |
| Рисунок 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. 46](#fig-046)).

|  |
| --- |
| Рисунок 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.