Отчёт по лабораторной работе №4

Дисциплина: Администрирование сетевых подсистем

Мишина Анастасия Алексеевна

Содержание

# 1 Цель работы

Приобретение практических навыков по установке и базовому конфигурированию HTTP-сервера Apache.

# 2 Выполнение лабораторной работы

## 2.1 Установка HTTP-сервера

Загрузим нашу операционную систему и перейдем в рабочий каталог с проектом. Далее запустим виртуальную машину server: vagrant up server. На виртуальной машине server войдём под созданным нами в предыдущей работе пользователем и откроем терминал. Перейдём в режим суперпользователя: sudo -i и установим из репозитория стандартный веб-сервер (HTTP-сервер и утилиты httpd, криптоутилиты и пр.): LANG=C yum grouplist и dnf -y groupinstall “Basic Web Server” (рис. 1).

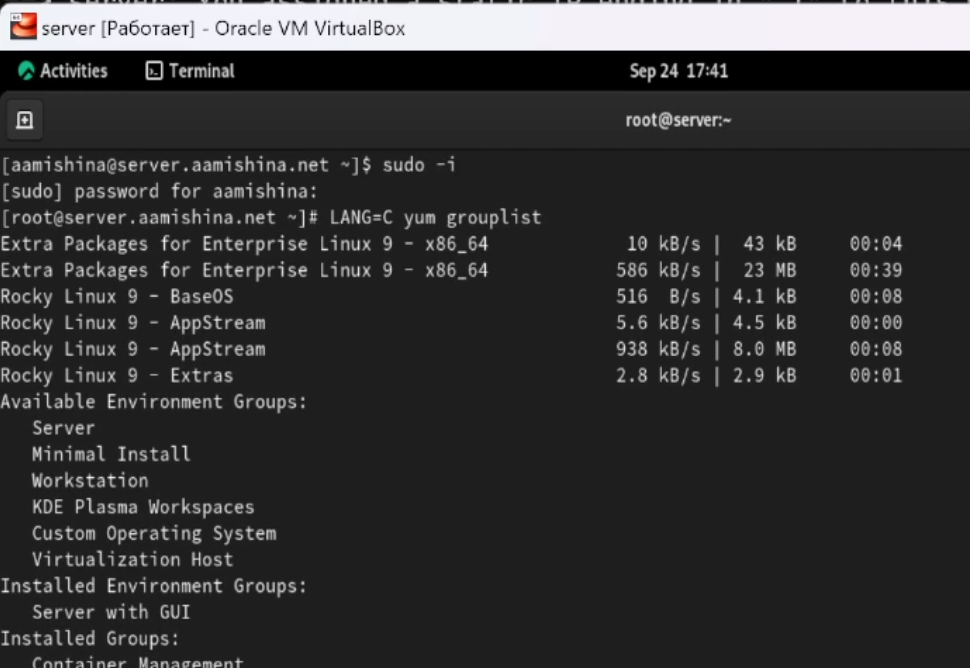


Рис. 1: Переход в режим суперпользователя и установка стандартного веб-сервера

## 2.2 Базовое конфигурирование HTTP-сервера

Просматриваем содержание конфигурационных файлов в каталогах /etc/httpd/conf и /etc/httpd/conf.d (рис. 2).

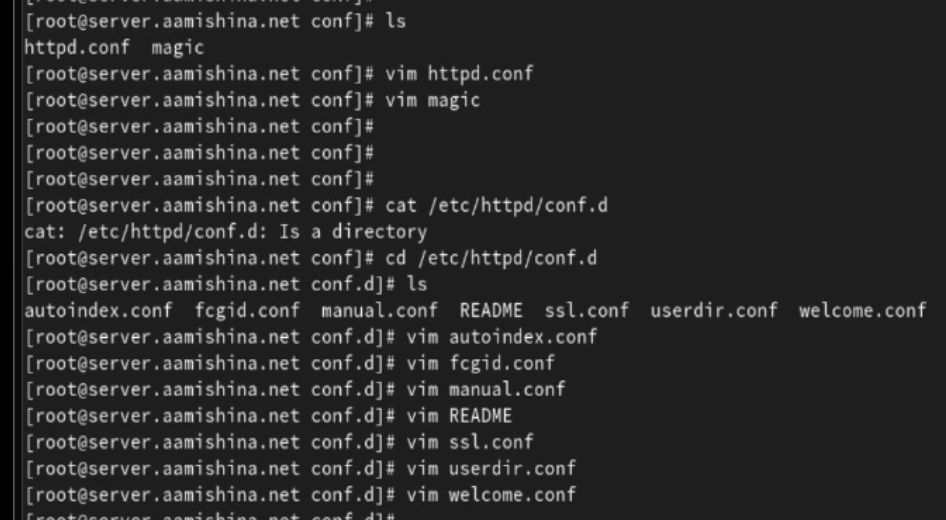


Рис. 2: Просмотр содержимого каталогов /etc/httpd/conf etc/httpd/conf.d

Вносим изменения в настройки межсетевого экрана узла server, разрешив работу с http. Во втором терминале открываем расширенный лог системных сообщений. В первом терминале активируем и запускаем HTTP-сервер: systemctl enable httpd и systemctl start httpd (рис. 3).

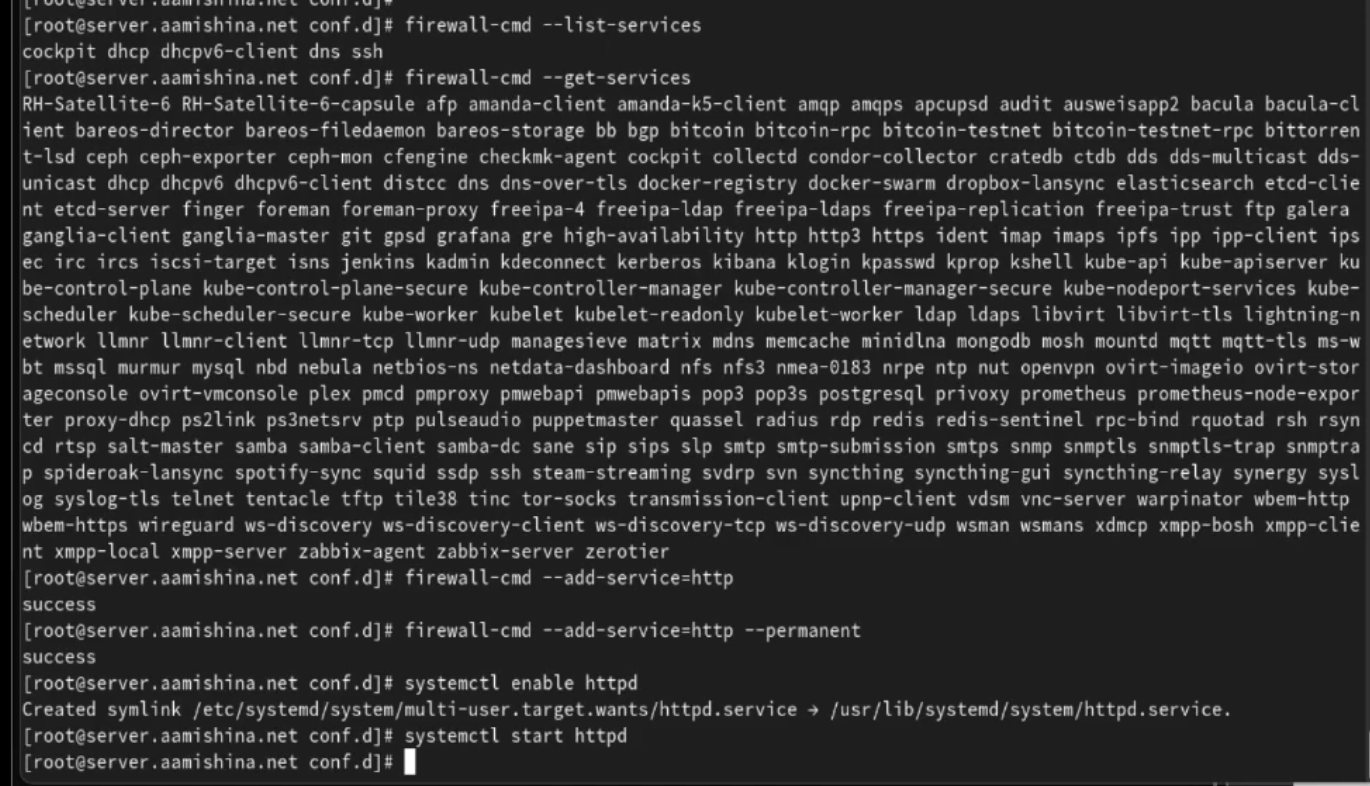


Рис. 3: Внесение изменений в настройки межсетевого экрана, запуск HTTP-сервера

В логе системных сообщений видим, что веб-сервер запущен (рис. 4)

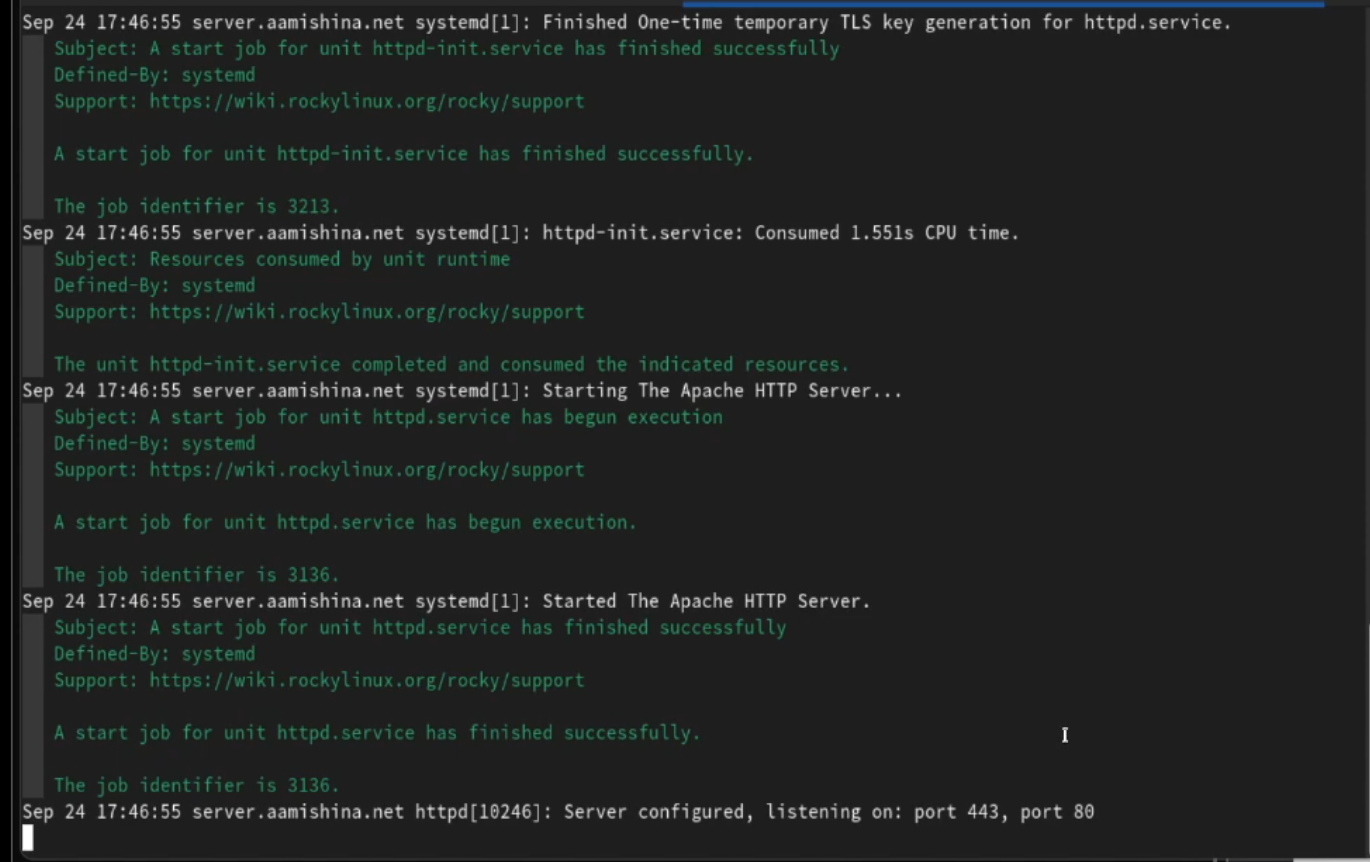


Рис. 4: Расширенный лог системных сообщений

## 2.3 Анализ работы HTTP-сервера

На виртуальной машине server открываем лог ошибок работы веб-сервера: tail -f /var/log/httpd/error\_log; мониторинг доступа к веб-серверу: tail -f /var/log/httpd/access\_log.

Запустив ВМ client, открывем браузер и в адресной строке вводим 192.168.1.1 (рис. 5)

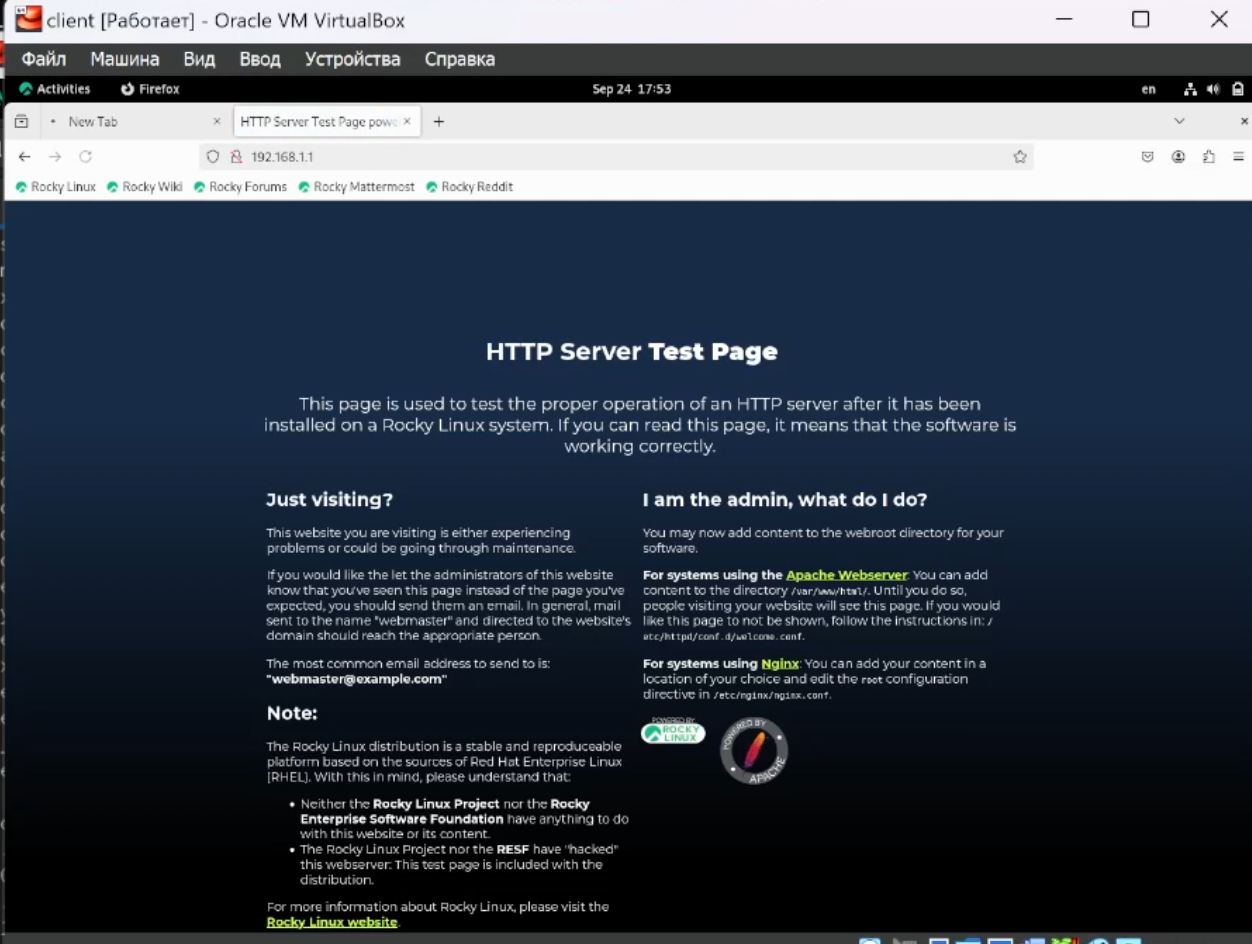


Рис. 5: Тестовая страницы HTTP-сервера

Видим записи в мониторинге доступа и в логе ошибок о подключении к серверу (рис. 6).

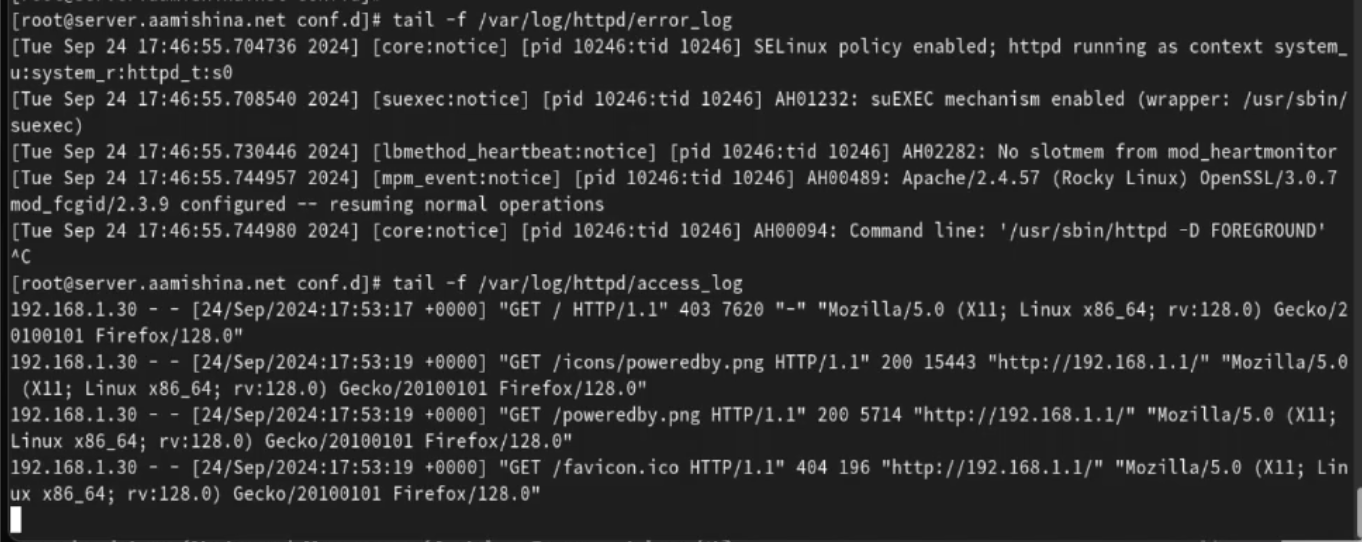


Рис. 6: Запись в мониторинге доступа. Запись в логе ошибок

## 2.4 Настройка виртуального хостинга для HTTP-сервера

Останавливаем работу DNS-сервера: systemctl stop named. Добавляем запись для HTTP-сервера в конце файла прямой DNS-зоны (рис. 7).

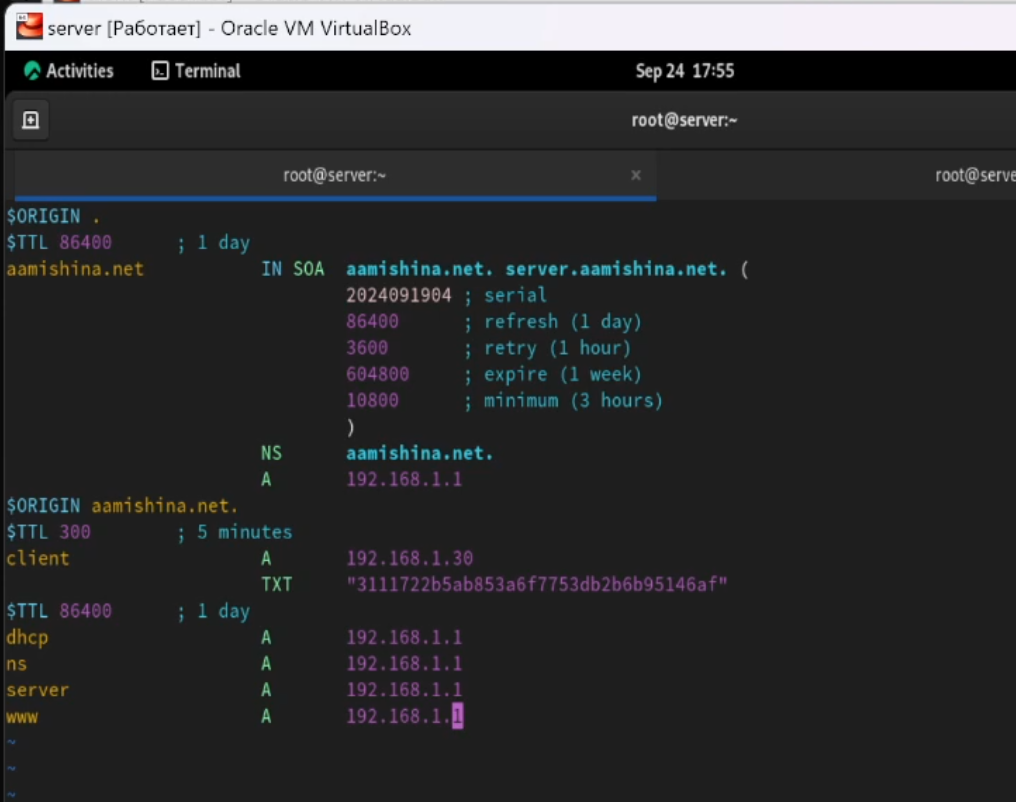


Рис. 7: Добавление записи в конец файла прямой DNS-зоны

Добавляем также запись в конце файла обратной зоны (рис. 8)

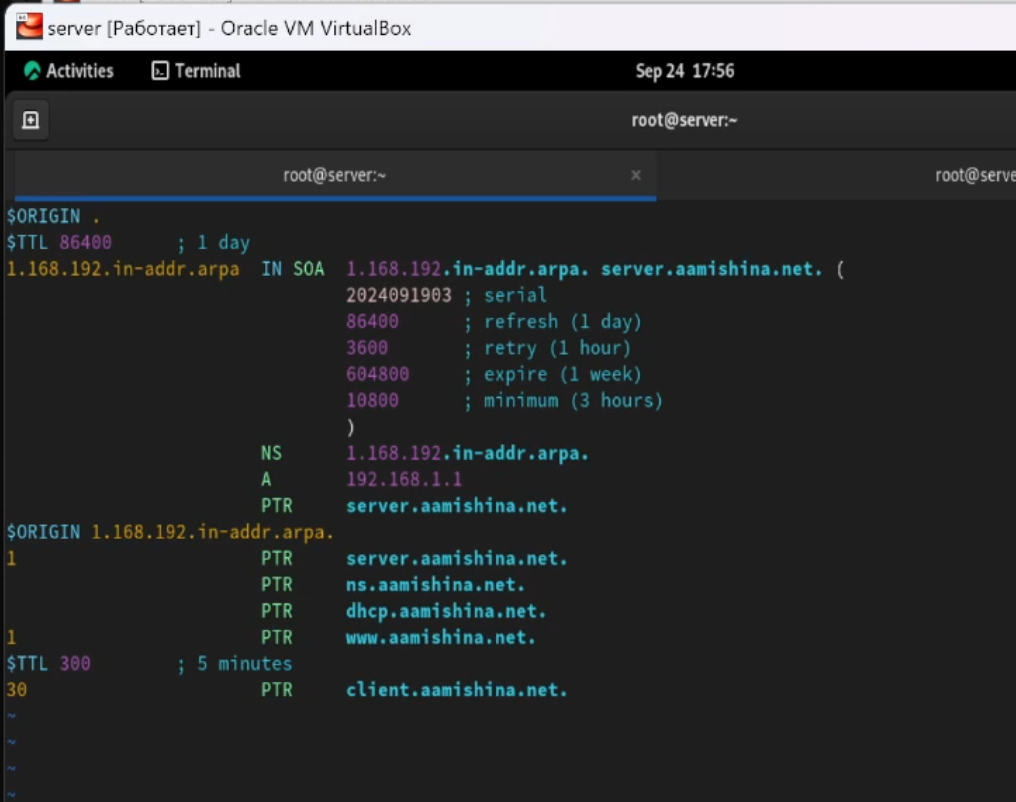


Рис. 8: Добавление записи в конец файла обратной DNS-зоны

Удаляем из соответствующих каталогов файлы журналов DNS. Перезапускаем DNS-сервер. В каталоге /etc/httpd/conf.d создаем файлы server.aamishina.net.conf и www.aamishina.net.conf: cd /etc/httpd conf.d, touch server.aamishina.net.conf и touch www.aamishina.net.conf/(рис. 9).

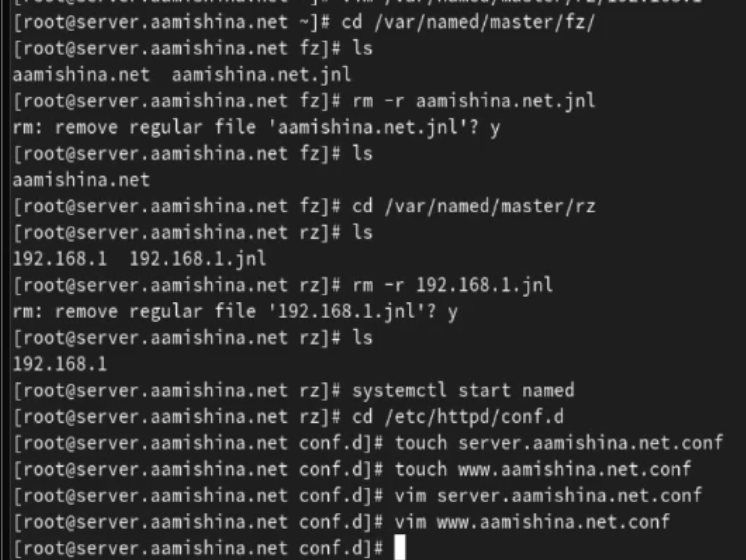


Рис. 9: Удаление файлов журналов DNS, перезапуск DNS-сервера, создание файлов

Редактируем server.aamishina.net.conf (рис. 10).

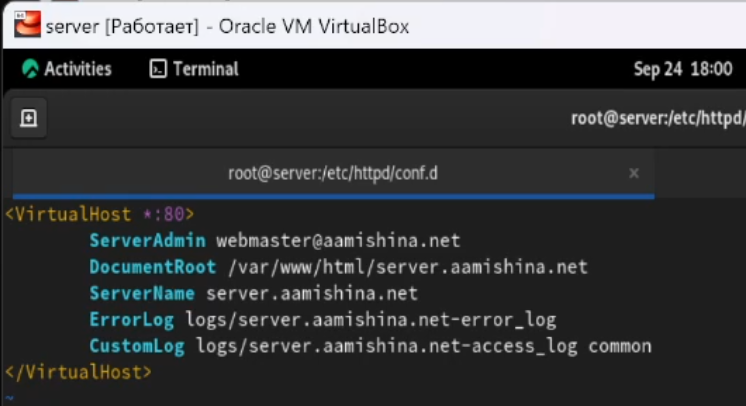


Рис. 10: Редактирование server.aamishina.net.conf

Редактируем www.aamishina.net.conf (рис. 11).

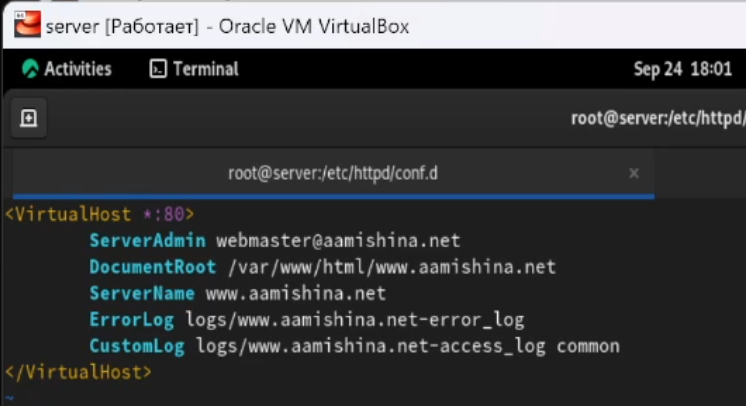


Рис. 11: Редактирование www.aamishina.net.conf

Переходим в /var/www/html создаем каталог server.aamishina.net и в нем файл index.html (рис. 12). Вводим в файл сообщение *Welcome to the server.aamishina.net server.* Аналогично создаем каталог и файл для www.aamishina.net.

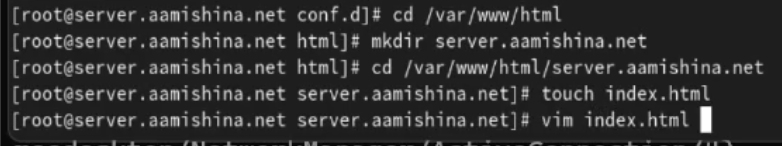


Рис. 12: Создание каталога и файла главной страницы

Корректируем права доступа в каталог с веб-контентом, восстанавливаем контекст безопасности в SELinux, перезагружаем HTTP-сервер (рис. 13)

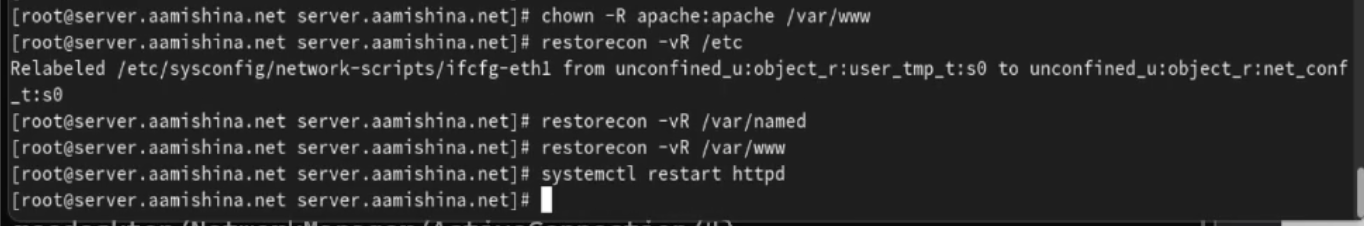


Рис. 13: Корректирование прав доступа, восстановление контекста безопасности, перезагрузка httpd

На ВМ client ввожу в адресную строку браузера server.aamishina.net (рис. 14) и www.aamishina.net (рис. 15).

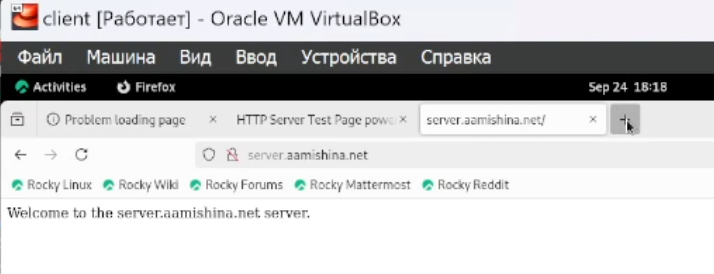


Рис. 14: Доступ к server.aamishina.net

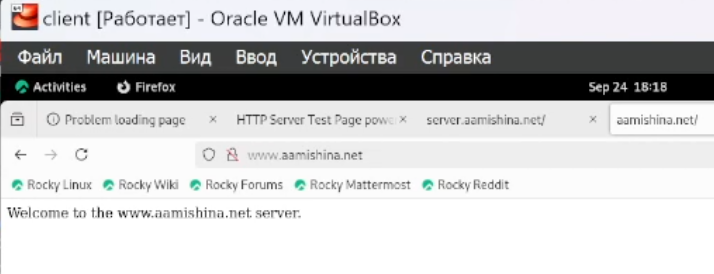


Рис. 15: Доступ к www.aamishina.net

## 2.5 Внесение изменений в настройки внутреннего окружения виртуальной машины

На ВМ server переходим в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создаем в нём каталог http, в который помещаем в соответствующие подкаталоги конфигурационные файлы HTTP-сервера. Также заменяем конфигурационные файлы DNS-сервера: cd /vagrant/provision/server/dns/ и cp -R /var/named/\* /vagrant/provision/server/dns/var/named/ (рис. 16).



Рис. 16: Создание каталога http, помещение конфигурационных файлов HTTP-сервера, заменя конфигурационных файлов DNS-сервера

В каталоге /vagrant/provision/server создаем исполняемый файл http.sh, заполняем его листингом из ТУИСа (рис. 17).

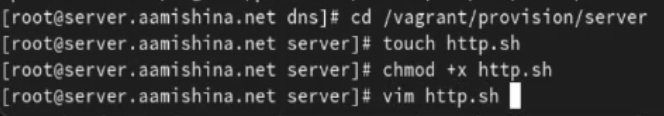


Рис. 17: Создание скрипта http.sh

Для отработки скрипта во время запуска добавляем в Vagrantfile в разделе конфигурации для сервера отрывок из мануала на ТУИСе (рис. 18). После этого выключаю ВМ: vagrant halt client и vagrant halt server.

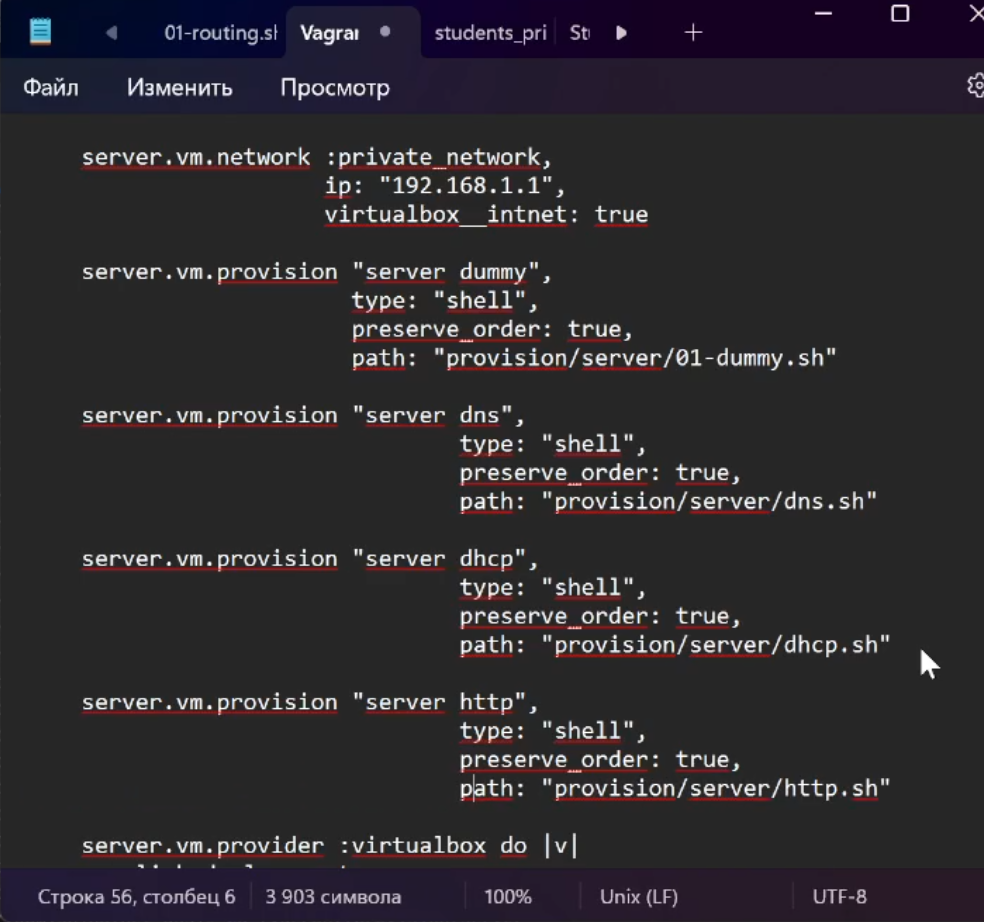


Рис. 18: Vagrantfile

# 3 Ответы на контрольные вопросы

1. Через какой порт по умолчанию работает Apache?

* По умолчанию Apache работает через порт 80 для HTTP и порт 443 для HTTPS.

1. Под каким пользователем запускается Apache и к какой группе относится этот пользователь?

* Apache обычно запускается от имени пользователя www-data (или apache, в зависимости от дистрибутива) и относится к группе с тем же именем.

1. Где располагаются лог-файлы веб-сервера? Что можно по ним отслеживать?

* Лог-файлы веб-сервера обычно располагаются в директории логов. Например, в Ubuntu логи Apache хранятся в /var/log/apache2/, а в CentOS - в /etc/httpd/logs/. Лог-файлы содержат информацию о запросах к серверу, ошибки, статусы запросов и другие события, что позволяет администраторам отслеживать активность и выявлять проблемы.

1. Где по умолчанию содержится контент веб-серверов?

* Контент веб-серверов по умолчанию обычно находится в директории, называемой “DocumentRoot”. Например, в Apache на Linux DocumentRoot по умолчанию установлен в /var/www/html/. В этой директории содержатся файлы, которые веб-сервер отдает при запросах.

1. Каким образом реализуется виртуальный хостинг? Что он даёт?

* Виртуальный хостинг в Apache позволяет хостить несколько сайтов на одном сервере. Разные сайты обслуживаются на одном IP-адресе, но на разных доменных именах. Это основывается на значении заголовка “Host” в HTTP-запросе, который используется для определения, какой виртуальный хост должен обработать запрос. Виртуальный хостинг позволяет хозяину сервера размещать несколько сайтов на одном физическом сервере, управлять ими независимо, и предоставлять услуги хостинга для различных клиентов или проектов.

# 4 Выводы

В результате выполнения работы были приобретены практические навыки по установке и базовому конфигурированию HTTP-сервера Apache.