РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра теории вероятностей и кибербезопасности

ОТЧЁТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №5

дисциплина: Администрирование сетевых подсистем

Студент: Пакавира Арсениу Висенте Луиш

Студ. билет № 1032225105

Группа: НФИбд-02-23

МОСКВА
2025 г.

Цель работы:

Целью данной работы является приобретение практических навыков по расширенному конфигурированию HTTP-сервера Apache в части безопасности и возможности использования PHP.

Выполнение работы:

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

cd /var/tmp/claudely/vagrant

Далее запустим виртуальную машину server (Рис. 1.1):

make server-up

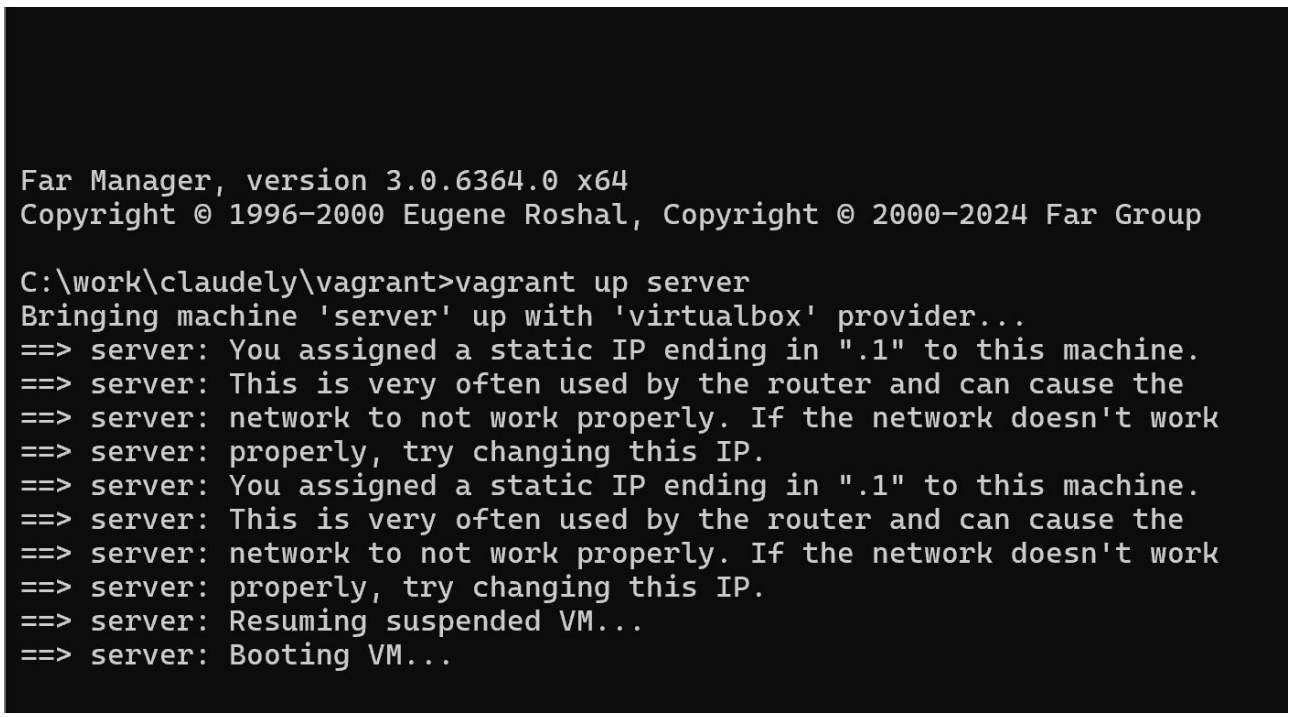

Рис. 1.1. Открытие рабочего каталога с проектом и запуск виртуальной машины server.

На виртуальной машине server войдём под нашим пользователем и откроем терминал. Далее перейдём в режим суперпользователя (Рис. 1.2):

sudo -i

В каталоге /etc/ssl создадим каталог private:

mkdir -p /etc/pki/tls/private ln -s

/etc/pki/tls/private /etc/ssl/private

cd /etc/pki/tls/private

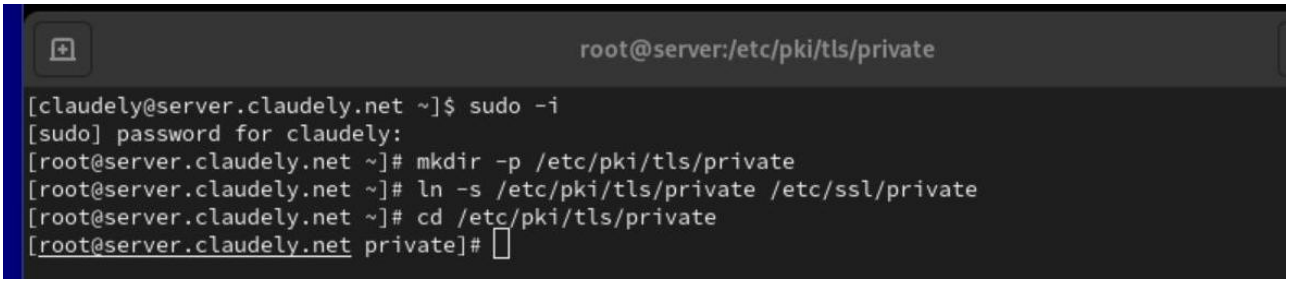

Рис. 1.2. Переход в режим суперпользователя и создание в каталоге /etc/ssl каталога private.

Сгенерируем ключ (Рис. 1.3) и сертификат (Рис. 1.4), используя следующую команду: openssl req -x509 -nodes -newkey rsa:2048 -keyout www.claudely.net.key -out

www.claudely.net.crt mv www.claudely.net.crt

/etc/pki/tls/certs

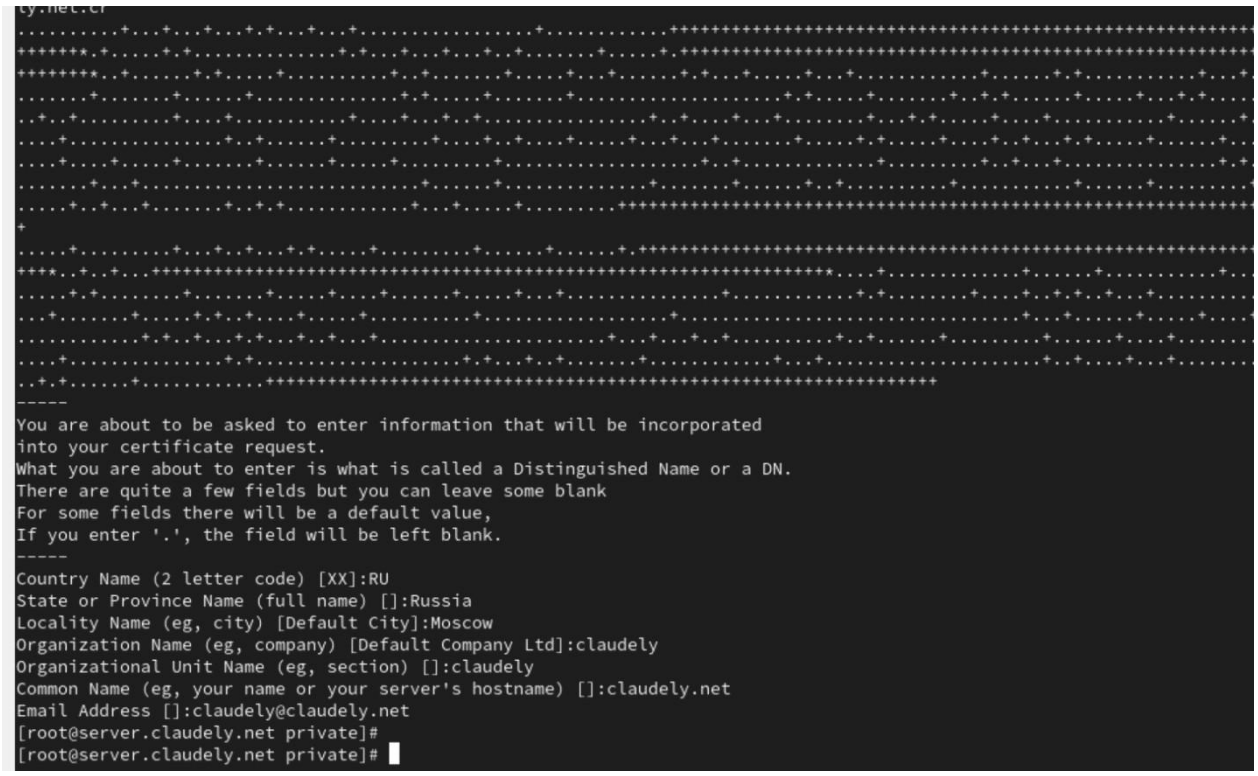

Рис. 1.3. Генерация ключа.

Рис. 1.4. Генерация сертификата.

Для перехода веб-сервера www.claudely.net на функционирование через протокол HTTPS требуется изменить его конфигурационный файл. Для этого перейдём в каталог с конфигурационными файлами (Рис. 1.5):

cd /etc/httpd/conf

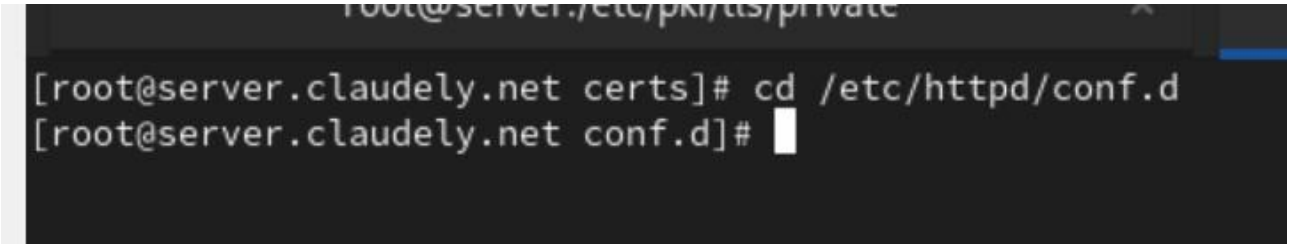

Рис. 1.5. Открытие каталога с конфигурационными файлами.

Откроем на редактирование файл /etc/httpd/conf.d/www.claudely.net.conf и заменим его содержимое на то, которое дано нам в лабораторной работе (Рис. 1.6):

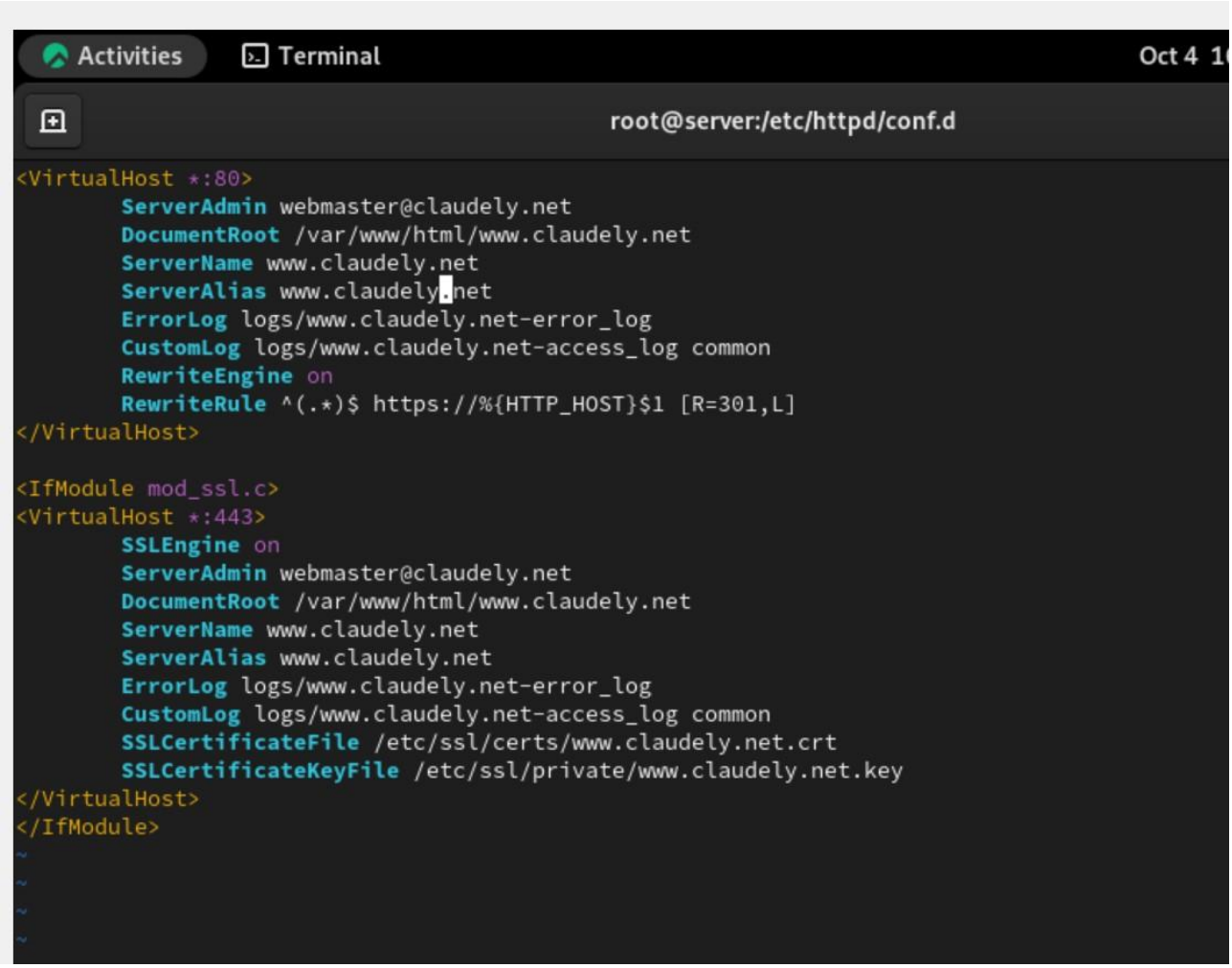

Рис. 1.6. Открытие файла /etc/httpd/conf.d/www.claudely.net.conf на редактирование и замена содержимого.

Внесём изменения в настройки межсетевого экрана на сервере, разрешив работу с https (Рис. 1.7):

firewall-cmd --list-services firewall-cmd

--get-services firewall-cmd --add-

service=https firewall-cmd --add-

service=https --permanent

firewall-cmd --reload

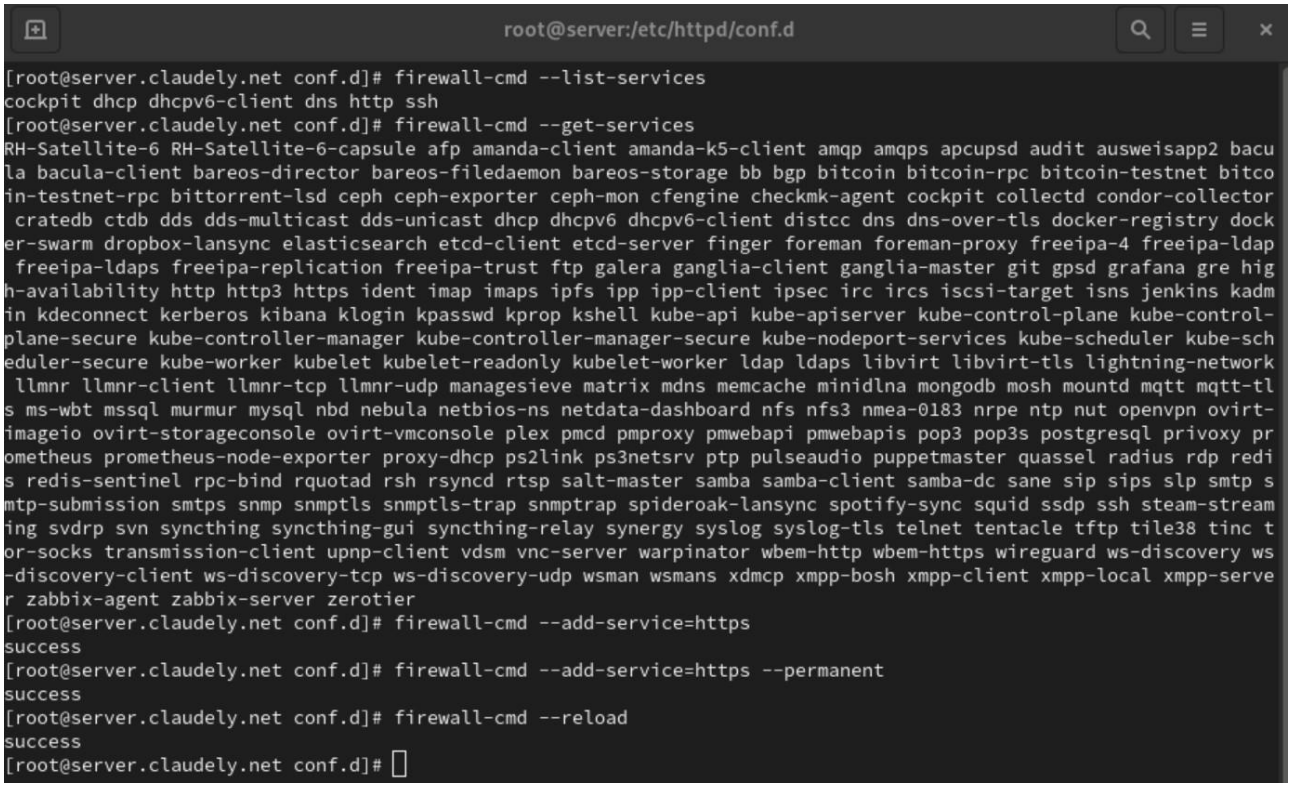

Рис. 1.7. Внесение изменений в настройки межсетевого экрана на сервере, разрешив работу с https.

Перезапустим веб-сервер (Рис. 1.8):

systemctl restart httpd

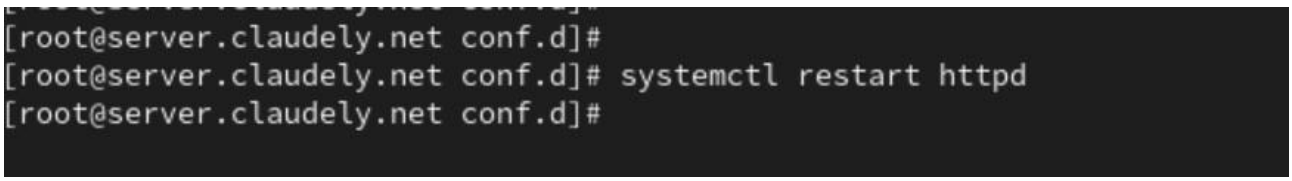

Рис. 1.8. Перезапуск веб-сервера.

На виртуальной машине client в строке браузера введём название вебсервера www.claudely.net и убедимся, что произошло автоматическое

переключение на работу по протоколу HTTPS (Рис. 1.9). На открывшейся странице с сообщением о незащищённости соединения нажмём кнопку «Дополнительно», затем добавим адрес нашего сервера в постоянные исключения. Затем просмотрим содержание сертификата.

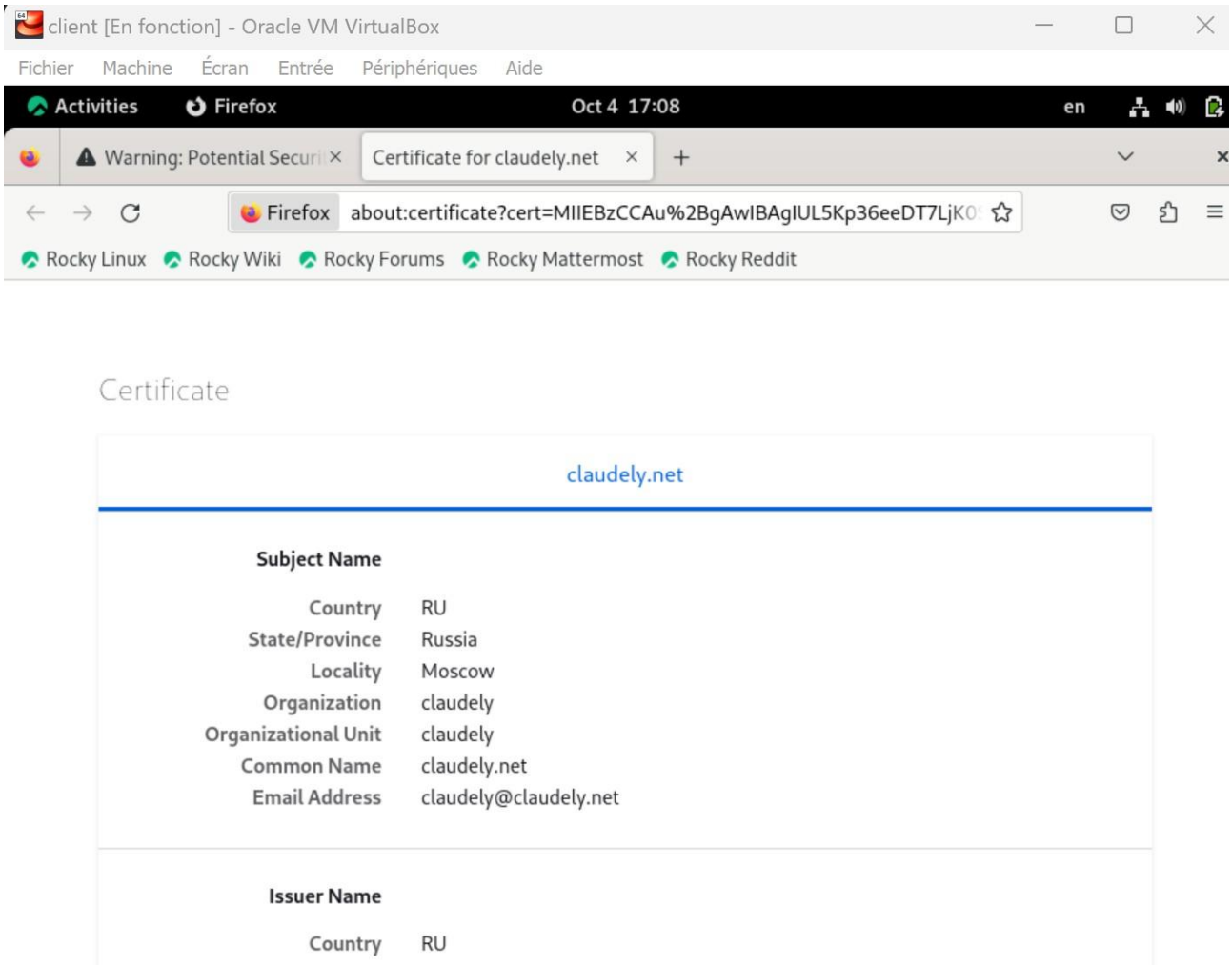

Рис. 1.9. Открытие веб-сервера www.claudely.net и автоматическое переключение на работу по протоколу HTTPS.

Установим пакеты для работы с PHP (Рис. 2.1):

dnf -y install php

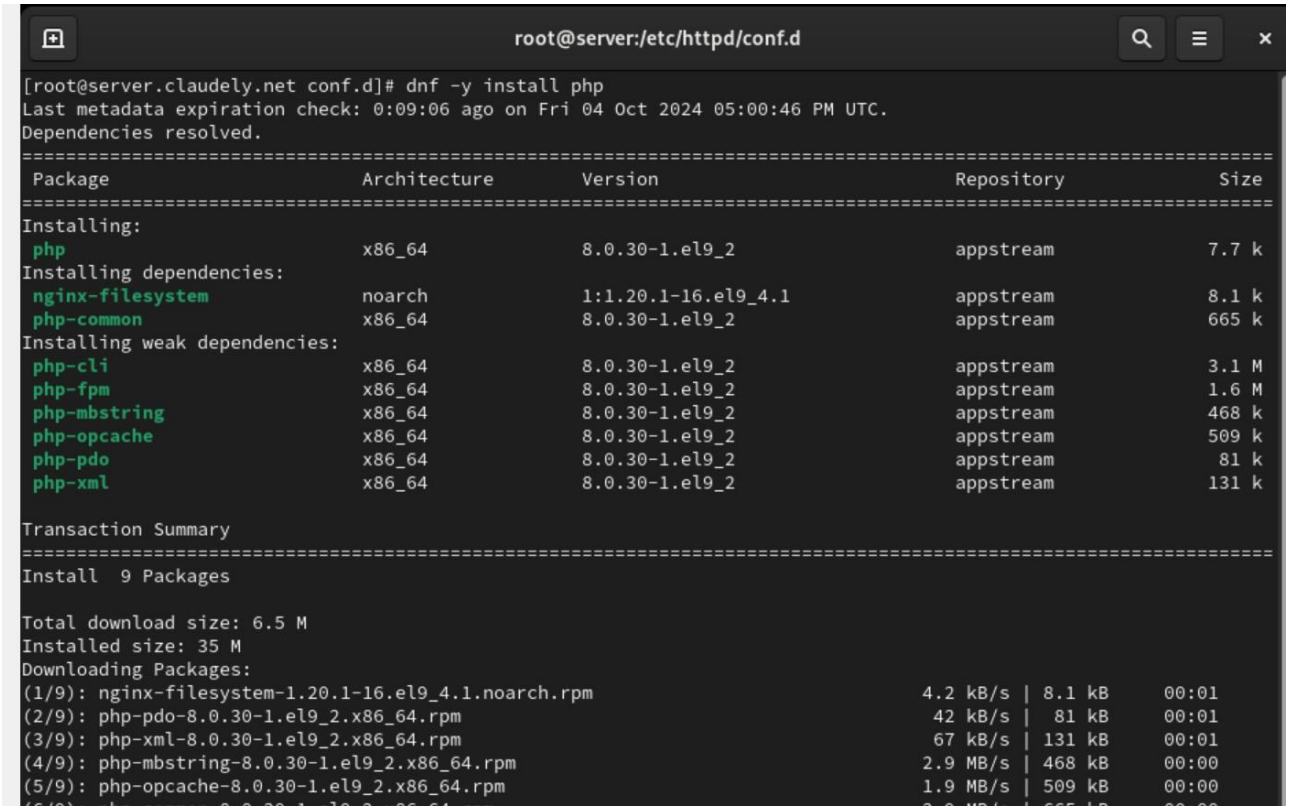

Рис. 2.1. Установка пакетов для работы с PHP.

В каталоге /var/www/html/www.claudely.net заменим файл index.html на index.php следующего содержания (рис. 2.2):

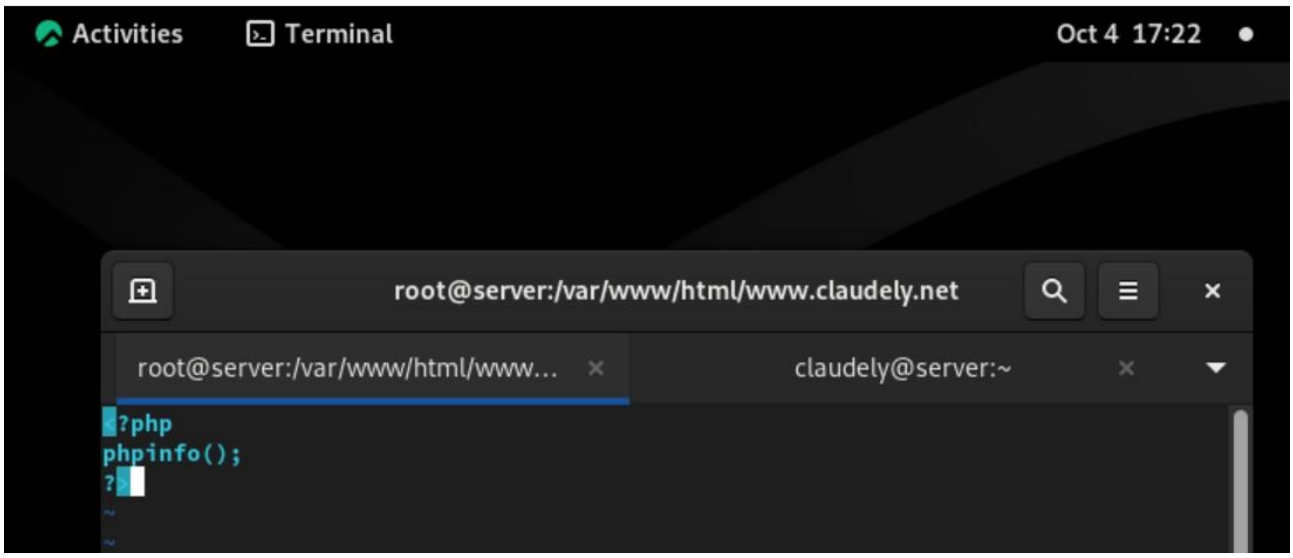

Рис. 2.2. Замена файла index.html на index.php с содержанием из лабораторной работы.

Скорректируем права доступа в каталог с веб-контентом:

chown -R apache:apache /var/www

После чего восстановим контекст безопасности в SELinux:

restorecon -vR /etc restorecon

-vR /var/www

И перезапустим HTTP-сервер (рис. 2.3):

systemctl restart httpd

```
[root@server.claudely.net www.claudely.net]# chown -R apache:apache /var/www
[root@server.claudely.net www.claudely.net]#
[root@server.claudely.net www.claudely.net]# restorecon -vR /etc
[root@server.claudely.net www.claudely.net]# restorecon -vR /var/www
[root@server.claudely.net www.claudely.net]# systemctl restart httpd
[root@server.claudely.net www.claudely.net]#
```

Рис. 2.3. Корректировка прав доступа в каталог с веб-контентом, восстановление контекста безопасности в SELinux и перезапуск HTTP-сервера.

На виртуальной машине client в строке браузера введём название вебсервера www.claudely.net и убедимся, что будет выведена страница с информацией об используемой на веб-сервере версии PHP (рис. 2.4):

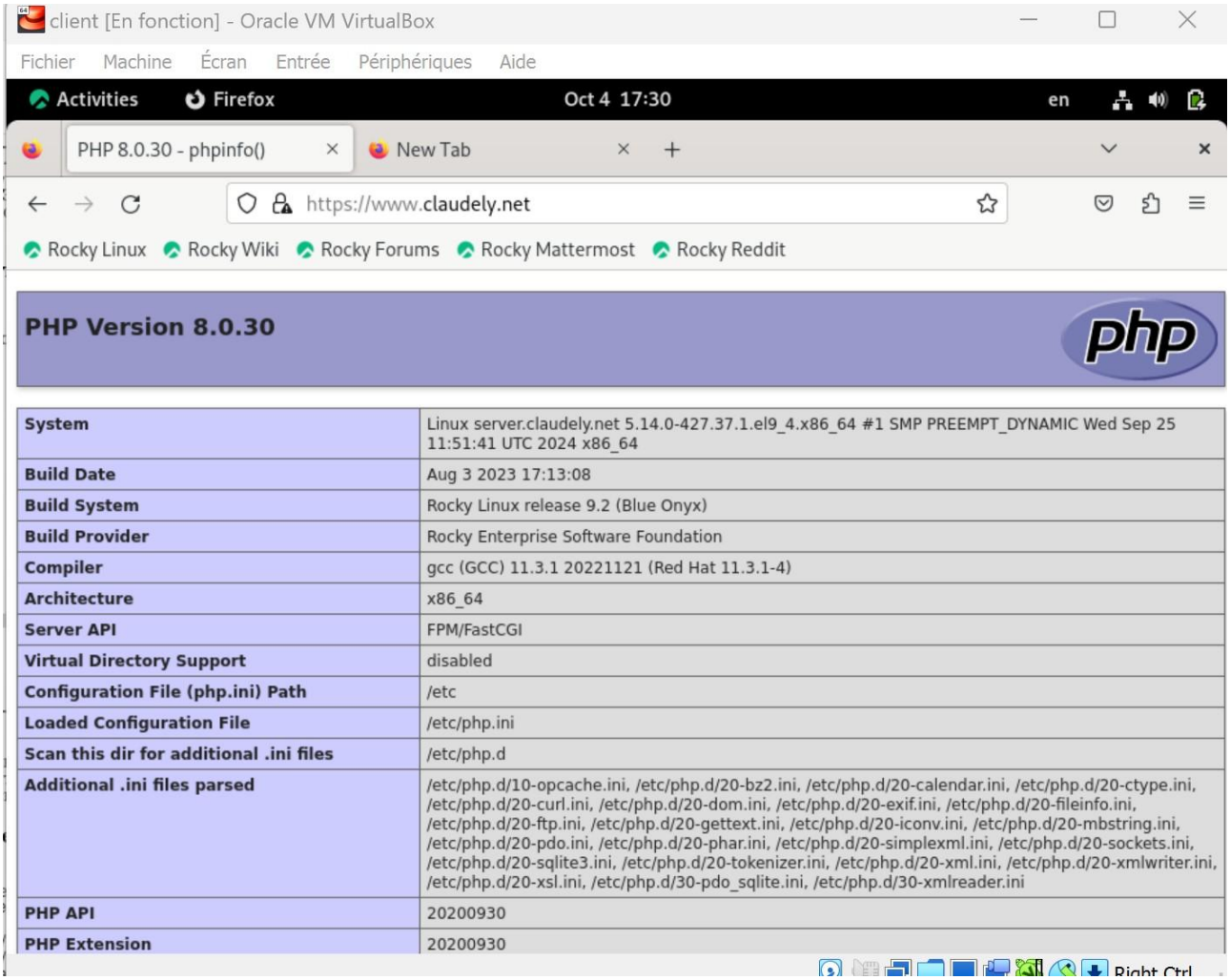

Рис. 2.4. Проверка вывода страницы с информацией об используемой на веб-сервере версии PHP.

На виртуальной машине server перейдём в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/http и в соответствующие каталоги скопируем конфигурационные файлы (рис. 3.1):

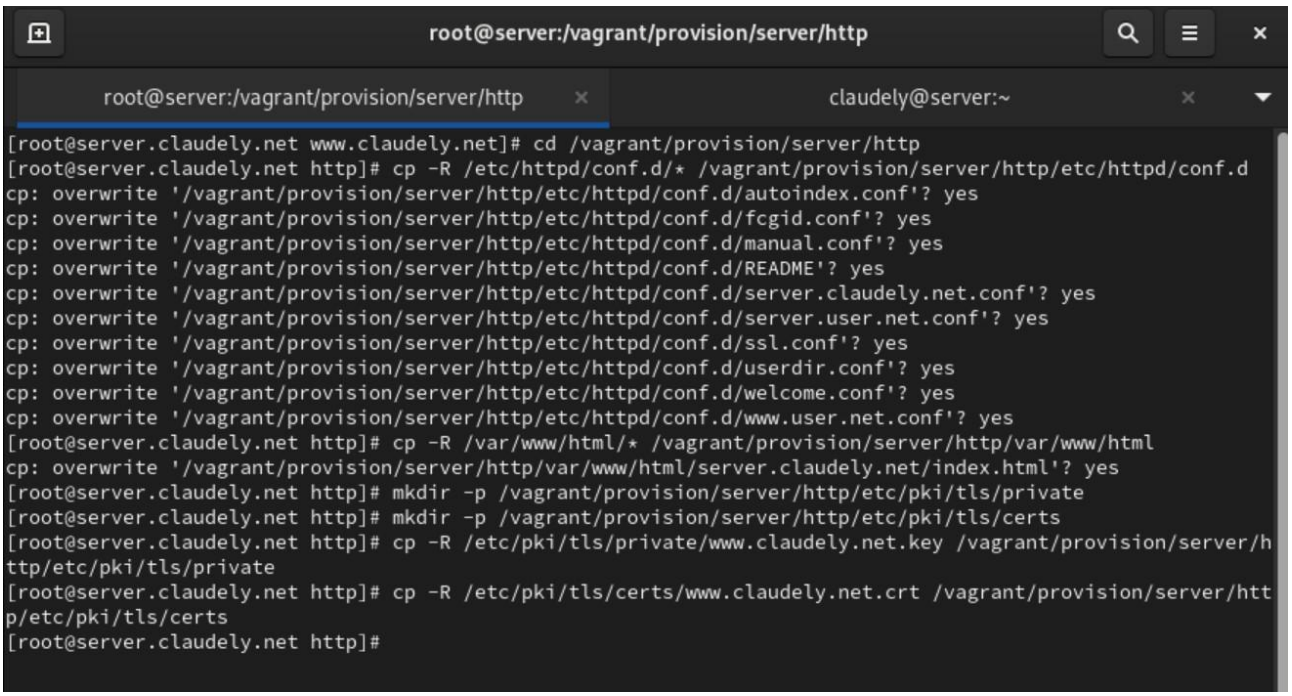

Рис. 3.1. Внесение изменений в настройки внутреннего окружения /vagrant/provision/server/http и копирование конфигурационных файлов в каталоги.

В имеющийся скрипт /vagrant/provision/server/http.sh внесём изменения, добавив установку PHP и настройку межсетевого экрана, разрешающую работать с https (рис. 3.2):

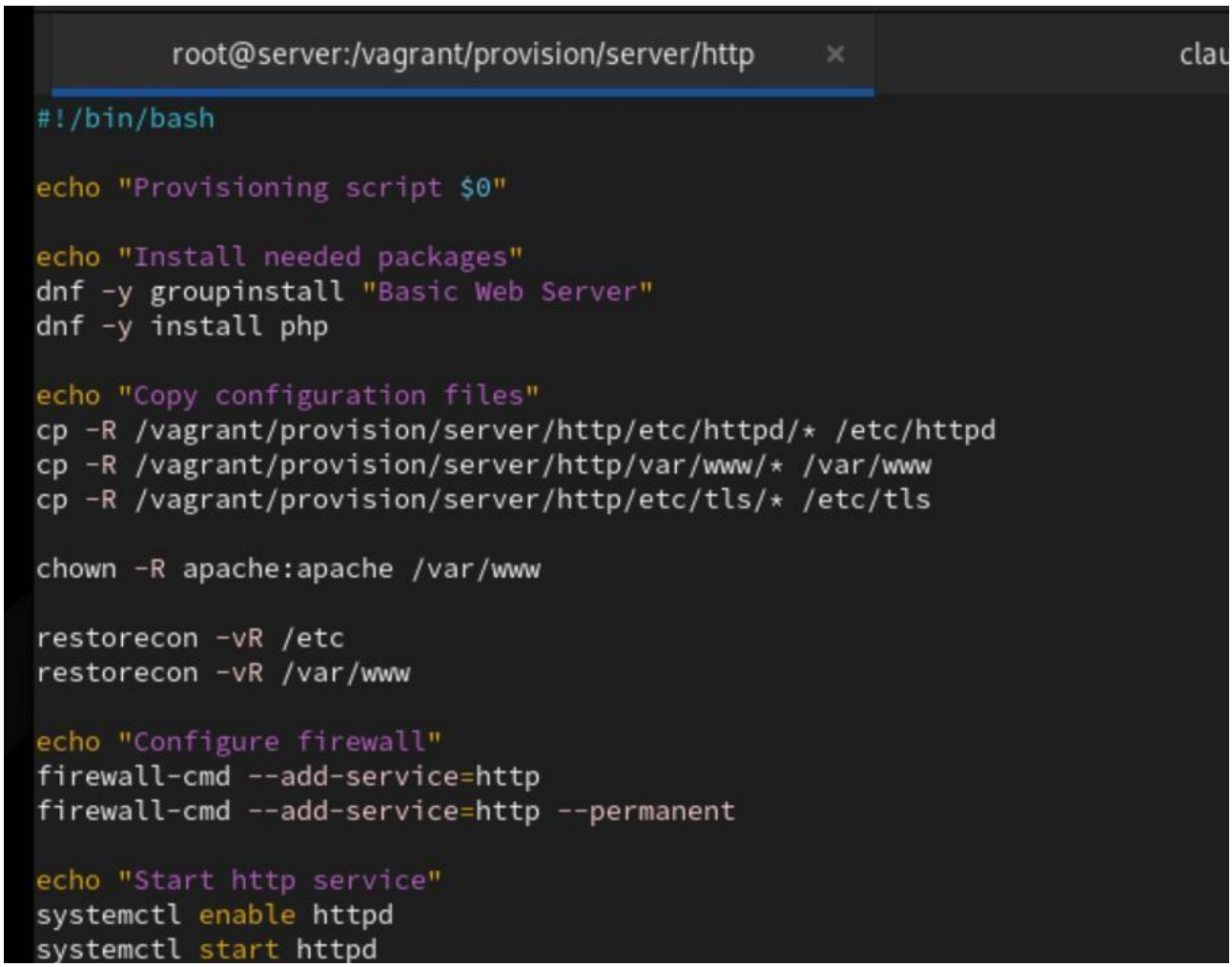

Рис. 3.2. Внесение изменений в скрипт /vagrant/provision/server/http.sh, добавив установку PHP и настройку межсетевого экрана, разрешающую работать с https.

Вывод:

В ходе выполнения лабораторной работы были приобретены практические навыки по расширенному конфигурированию HTTP-сервера Apache в части безопасности и возможности использования PHP.

Ответы на контрольные вопросы:

1. В чём отличие HTTP от HTTPS? – **Отличие HTTP от HTTPS:**

   HTTP (HyperText Transfer Protocol) – это протокол передачи данных, который используется для передачи информации между клиентом (например, веббраузером) и сервером. Однако он не обеспечивает шифрование данных, что делает их уязвимыми к перехвату злоумышленниками.

   HTTPS (HyperText Transfer Protocol Secure) - это расширение протокола HTTP с добавлением шифрования, обеспечивающее безопасную передачу данных между клиентом и сервером. Протокол HTTPS использует SSL (Secure Sockets Layer) или более современный TLS (Transport Layer Security) для шифрования данных.

2. Каким образом обеспечивается безопасность контента веб-сервера при работе через HTTPS? – **Обеспечение безопасности контента веб-сервера при работе через HTTPS:**

   Шифрование данных: при использовании HTTPS данные, передаваемые между клиентом и сервером, шифруются, что делает их невозможными для прочтения злоумышленниками, перехватывающими трафик.

   Идентификация сервера: сервер предоставляет цифровой сертификат, подтверждающий его легитимность. Этот сертификат выдается сертификационным центром и содержит информацию о владельце сертификата, публичный ключ для шифрования и подпись, подтверждающую подлинность сертификата.

3. Что такое сертификационный центр? Приведите пример. - **Сертификационный центр:**

   Определение: сертификационный центр (Центр сертификации) - это доверенная сторона, которая выдает цифровые сертификаты, подтверждающие подлинность владельца сертификата.

Пример: Одним из известных сертификационных центров является "Let's Encrypt". Он предоставляет бесплатные SSL-сертификаты, которые используются для обеспечения безопасного соединения на множестве вебсайтов. Владельцы веб-сайтов могут получить сертификат от Let's Encrypt, чтобы обеспечить шифрование и подтвердить свою легитимность в онлайнсреде.