РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

**Факультет физико-математических и естественных наук**

**Кафедра прикладной информатики и теории вероятностей**

ОТЧЕТ

по лабораторной работе № 5

**Расширенная настройка**

**HTTP-сервера Apache**

Дисциплина: Администрирование сетевых подсистем

Студент: Карташова А.С.

Группа: НФИбд-03-18

**МОСКВА**

2020 г.

Оглавление

[Цель работы 2](#_Toc57611611)

[Задачи 2](#_Toc57611612)

[Ход работы 3](#_Toc57611613)

[Конфигурирование HTTP-сервера для работы через протокол 3](#_Toc57611614)

[HTTPS 3](#_Toc57611615)

[Конфигурирование HTTP-сервера для работы с PHP 5](#_Toc57611616)

[Внесение изменений в настройки внутреннего окружения виртуальной машины 7](#_Toc57611617)

[Вывод 8](#_Toc57611618)

[Контрольные вопросы 8](#_Toc57611619)

# Цель работы

Приобретение практических навыков по расширенному конфигурированию HTTP сервера Apache в части безопасности и возможности использования PHP.

# Задачи

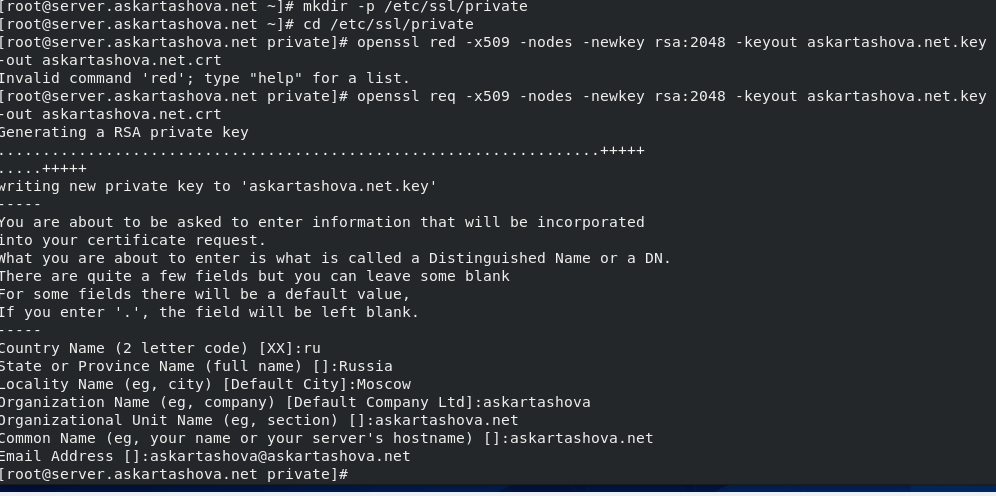
1. Сгенерируем криптографический ключ и само подписанный сертификат безопасности для возможности перехода веб-сервера от работы через протокол HTTP к работе через протокол HTTPS
2. Настроим веб-сервер для работы с PHP
3. Настроим виртуальный хостинг.
4. Напишем скрипт для Vagrant, фиксирующий действия по расширенной настройке HTTP-сервера во внутреннем окружении виртуальной машины server

# Ход работы

## Конфигурирование HTTP-сервера для работы через протокол

## HTTPS

Запустим виртуальную машину, перейдем в режим суперпользователя и сформируем ssh-ключ



Для перехода веб-сервера www.user.net на функционирование через протокол

HTTPS изменим его конфигурационный файл. Перейдем в каталог с конфигурационными файлами:

*cd /etc/httpd/conf.d*

Откроим на редактирование файл /etc/httpd/conf.d/www.askartashova.net.conf

и замените его содержимое на следующее:

**<*VirtualHost \*:80>*** *- адресс хоста для прослушивания и номер порта(80), \* вместо адреса означает, что веб сервер будет слушать соединения на всех адрес(на localhost и на внешнем)*

***ServerAdmin*** [***webmaster@user.net***](mailto:webmaster@user.net) ***-****email администратора сервера*

***DocumentRoot /var/www/html/www.user.net***

***ServerName*** [***www.user.net***](http://www.user.net) ***–*** *имя домена*

***ServerAlias*** [***www.user.net***](http://www.user.net) ***–*** *псевдоним домена*

***ErrorLog logs/www.user.net-error\_log*** *– место, куда сохраняются логи ошибок*

***CustomLog logs/www.user.net-access\_log common -*** *место, куда сохраняются логи доступа*

*Автоматическое подключение сервера к https*

***RewriteEngine on –*** *включает работу механизма преобразований*

***RewriteCond %{SERVER\_NAME} =*** [***www.user.net***](http://www.user.net) ***–*** *условие преобразования*

***RewriteRule ^(.\*)$ https://%{HTTP\_HOST}$1 [R=301,L]*** *– определяет правила работы механизма преобразований*

***</VirtualHost>***

При использовании безопасного протокола https для работы виртуального хоста, нужно создать хост на порту 433. Весь код секции *<IfModule mod\_ssl.c>* будет выполнен лишь в том случае, если активирован модуль mod\_ssl.c. Это нужно для того, чтобы, если модуль не активирован, код не выживал ошибок:

***<IfModule mod\_ssl.c>***

***<VirtualHost \*:443>***

***SSLEngine on*** *– включает поддержку SSL*

***ServerAdmin webmaster@user.net***

***DocumentRoot /var/www/html/www.user.net***

***ServerName www.user.net***

***ServerAlias www.user.net***

***ErrorLog logs/www.user.net-error\_log***

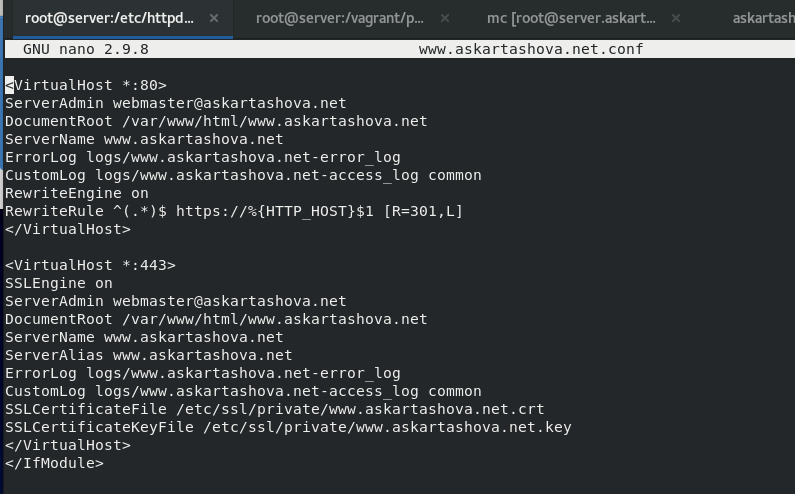
***CustomLog logs/www.user.net-access\_log common***

***SSLCertificateFile*** ***/etc/ssl/private/www.user.net.crt - пути*** *к файлам сертификата*

***SSLCertificateKeyFile /etc/ssl/private/www.user.net.key*** *- пути к файлам приватного ключа*

***</VirtualHost>***

***</IfModule>***

******

Внесем изменения в настройки межсетевого экрана на сервере, разрешив работу с https:

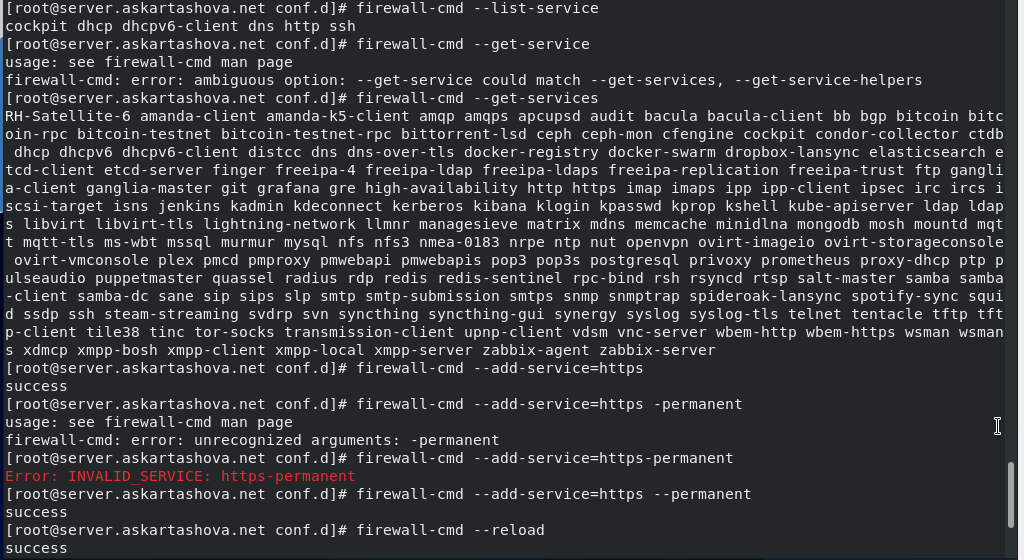
*firewall-cmd --list-services*

*firewall-cmd --get-services*

*firewall-cmd --add-service=https*

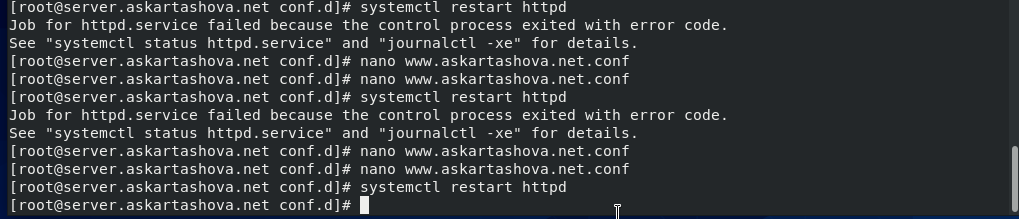
*firewall-cmd --add-service=https –permanent*

*firewall-cmd -reload*



Перезапустим веб-сервер:

*systemctl restart httpd*

**

При перезапуске веб-сервера у меня возникло 2 ошибки:

* В файле www.askartashova.net.conf потеряла строку <IfModule mod\_ssl.c>
* Неправильный путь к файлу ключа:при формировании файлы называются user.net.crt и user.net.key, а в www.askartashova.net.conf пути к несуществующим файлам

SSLCertificateFile /etc/ssl/private/www.askartashova.net.crt

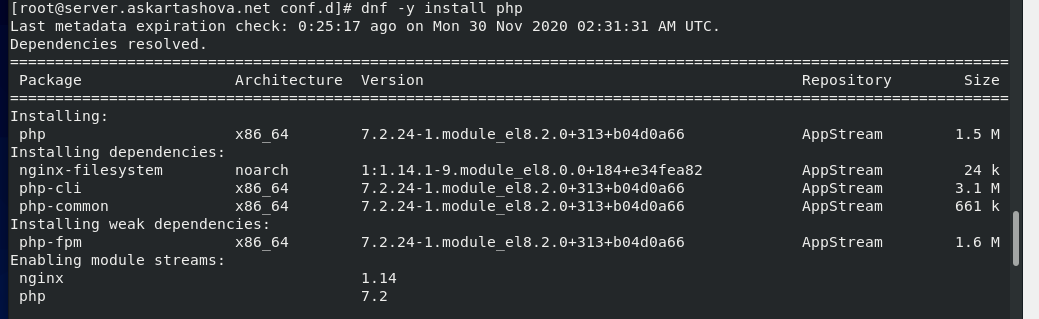
SSLCertificateKeyFile /etc/ssl/private/www.askartashova.net.key

На виртуальной машине client в строке браузера введем название веб-сервера www.askartashova.net и убедимся, что произойдёт автоматическое переключение на работу по протоколу HTTPS. Опять не работает, как и c http

## Конфигурирование HTTP-сервера для работы с PHP

Установим пакеты для работы с PHP:

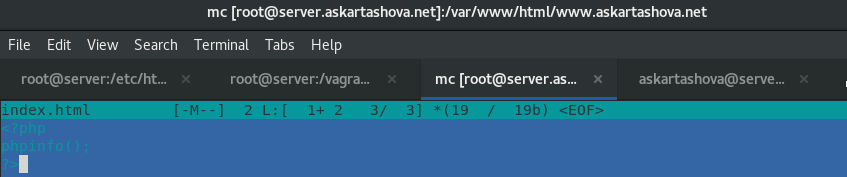
*dnf -y install php*



В каталоге /var/www/html/www.user.net заменим файл index.html на index.php следующего содержания:

*<?php*

*phpinfo();*

*?>*

Скорректируйем права доступа в каталог с веб-контентом:

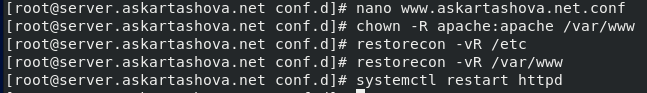
*chown -R apache:apache /var/www*

Восстановим контекст безопасности в SELinux:

*restorecon -vR /etc*

*restorecon -vR /var/www*

Перезапустим HTTP-сервер:

*systemctl restart httpd*

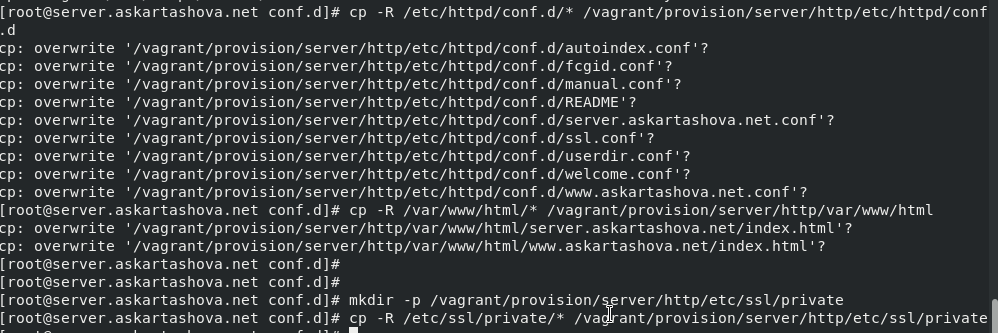
На виртуальной машине client в строке браузера введем название веб-сервера www.user.net и убедимся, что будет выведена страница с информацией об используемой на веб-сервере версии PHP.

## Внесение изменений в настройки внутреннего окружения виртуальной машины

На виртуальной машине server перейдем в каталог для внесения изменений

в настройки внутреннего окружения /vagrant/provision/server/http и в со-

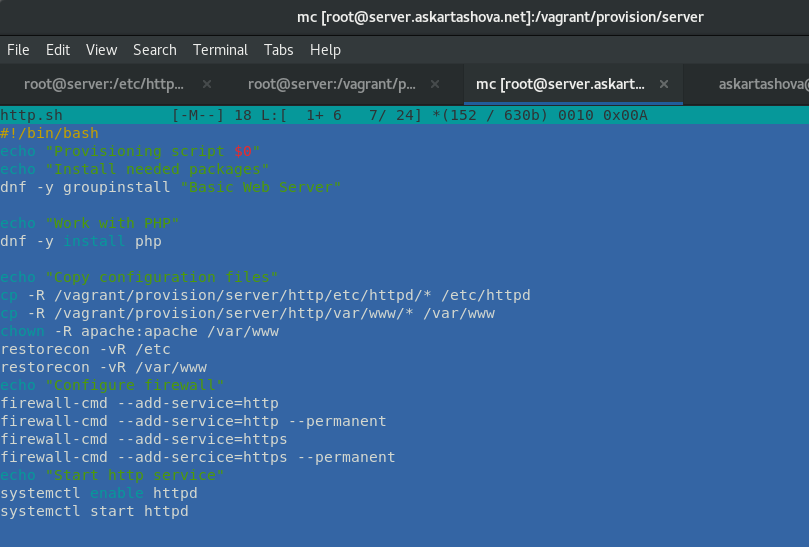
ответствующие каталоги скопируйте конфигурационные файлы:



В имеющийся скрипт /vagrant/provision/server/http.sh внесем измене-

ния, добавив установку PHP и настройку межсетевого экрана, разрешающую

работать с https.



# Вывод

Мы приобрели практические навыки по расширенному конфигурированию HTTP сервера Apache в части безопасности и возможности использования PHP.

# Контрольные вопросы

1. В чём отличие HTTP от HTTPS?

*HTTPS* — это расширение протокола HTTP для

поддержки шифрования в целях повышения безопасности.

2. Каким образом обеспечивается безопасность контента веб-сервера при работе через HTTPS?

Безопасность контента веб сервера достигается с помощью использования криптографических протоколов. Для шифрования может применяться протокол SSL (Secure Sockets Layer) или протокол TLC (Transport Layer Security). Оба протокола используют асимметричное шифрование для аутентификации, симметричное шифрование для конфиденциальности и коды аутентичности сообщений для сохранения целостности сообщений.

3. Что такое сертификационный центр? Приведите пример.

Сертификационный центр (Certification authority, CA) представляет

собой компонент глобальной службы каталогов, отвечающий за управление криптографическими ключами пользователей.

Цетр сертификации на сервере находится в каталоге /etc/ssl/private/ и состоит из 2х остальных компонентов :

* askartashova.net.crt - файл публичного сертификата ЦС. Пользователи, серверы и клиенты используют его для подтверждения для подтверждения подлинности системы и предотвращения атак через посредника.
* askartashova.net.key — закрытый ключ, который ЦС использует для подписания сертификатов серверов и клиентов.Если злоумышленик обнаружил ключ, его следует уничтожить