Отчёт по лабораторной работе 5

Расширенная настройка HTTP-сервера Apache

Заур Мустафаев

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

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

2 Выполнение работы

2.1 Конфигурирование HTTP-сервера для работы через протокол HTTPS

1. Выполнен вход на виртуальную машину **server**, осуществлён переход в режим суперпользователя и рабочий каталог для хранения SSL-ключей и сертификатов.

2. В каталоге `/etc/pki/tls` создан подкаталог `private`, а также сформирована символьная ссылка `/etc/ssl/private`, необходимая для корректной работы веб-сервера с SSL-ключами.

3. С использованием утилиты **OpenSSL** сгенерированы самоподписанный SSL-сертификат и закрытый ключ RSA длиной 2048 бит.
   В процессе генерации заполнены обязательные поля сертификата: код страны, страна, город, организация, подразделение, доменное имя сервера и адрес электронной почты.

Рис. 2.1: Генерация SSL-ключа и сертификата OpenSSL

4. Сформированный сертификат был скопирован в каталог `/etc/ssl/certs`, предназначенный для хранения SSL-сертификатов.

5. Выполнена настройка виртуального хоста веб-сервера Apache.

   Для порта **80** настроено автоматическое перенаправление запросов на HTTPS, для порта **443** включена поддержка SSL и указаны пути к сертификату и закрытому ключу.

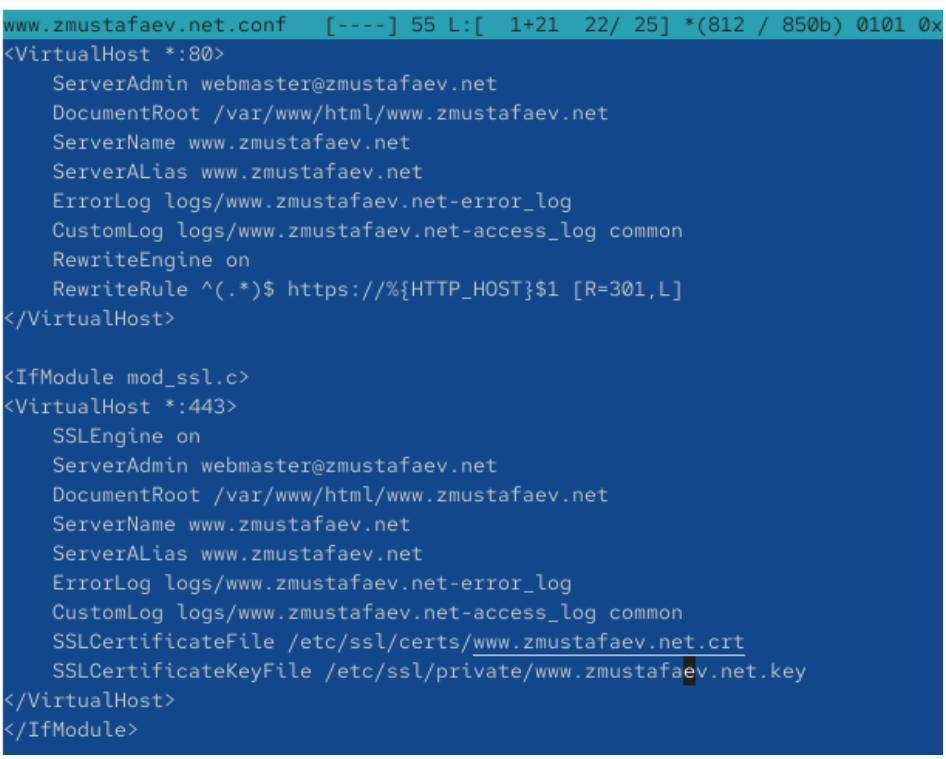

Рис. 2.2: Конфигурация виртуального хоста Apache для HTTPS

6. На клиентской виртуальной машине выполнено обращение к веб-серверу по протоколу HTTPS.

   В связи с использованием самоподписанного сертификата браузер вывел предупреждение о потенциальной угрозе безопасности.

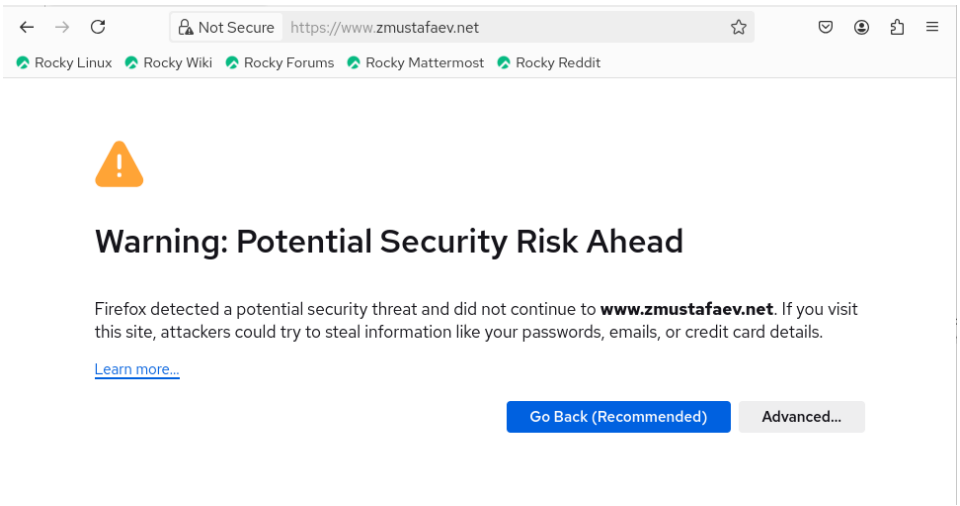

Рис. 2.3: Предупреждение браузера о самоподписанном сертификате

7. Адрес сервера был добавлен в постоянные исключения безопасности браузера, после чего обеспечен доступ к веб-ресурсу по защищённому соединению.

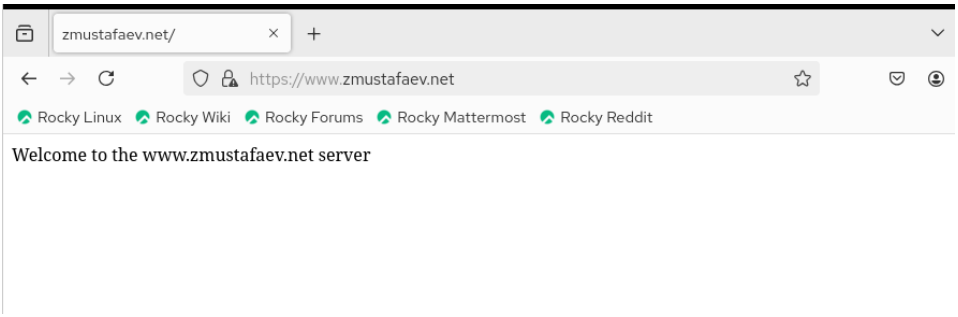

Рис. 2.4: Открытие сайта по HTTPS после добавления исключения

8. Выполнен просмотр сведений SSL-сертификата через интерфейс браузера. Подтверждено соответствие данных сертификата параметрам, указанным при его создании.

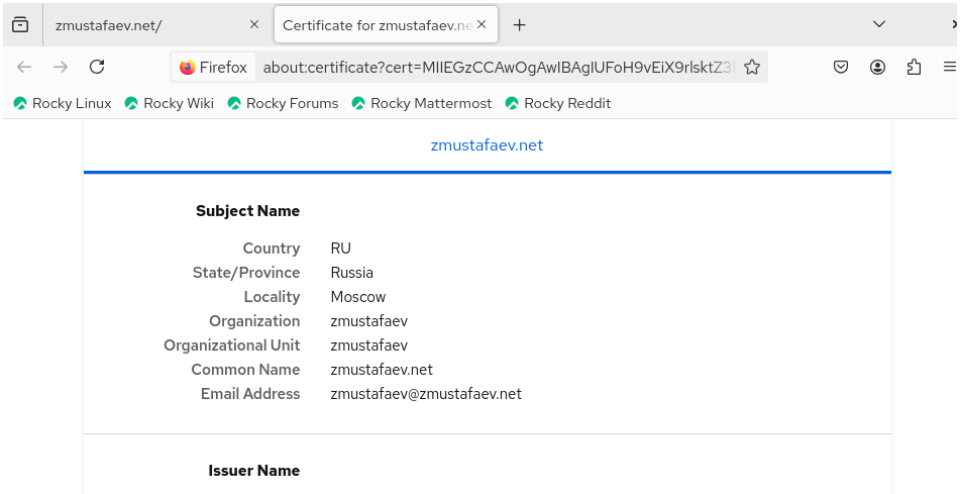

Рис. 2.5: Просмотр информации SSL-сертификата

2.2 Конфигурирование HTTP-сервера для работы с PHP

1. На сервере установлены пакеты **PHP** и дополнительные модули, необходимые для обработки PHP-скриптов веб-сервером Apache.

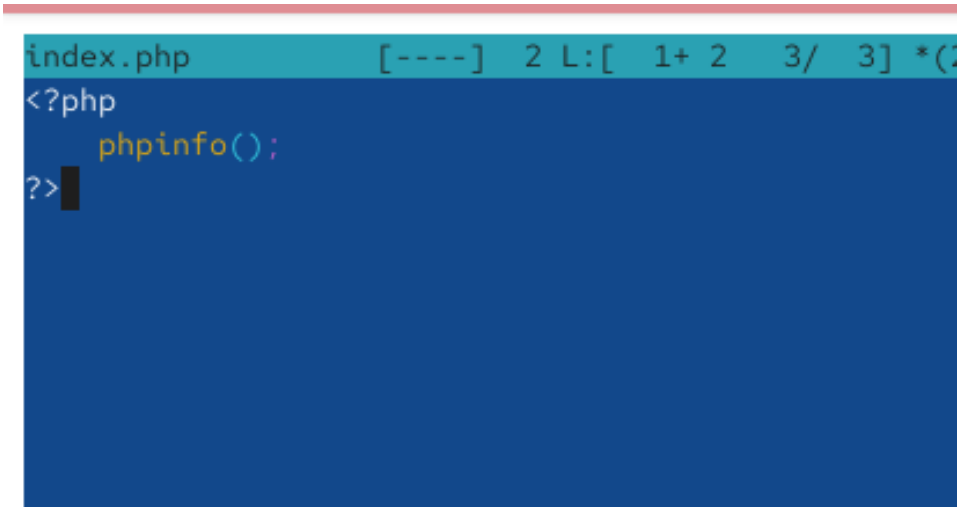

Рис. 2.6: Установка пакетов PHP

2. В каталоге `/var/www/html/www.zmustafaev.net` файл `index.html` был заменён на `index.php`, содержащий вызов функции `phpinfo()` для отображения параметров среды PHP.

Рис. 2.7: Создание файла index.php с phpinfo

3. Для каталога с веб-контентом заданы корректные права доступа, владель-

цем назначены пользователь и группа **apache**.

4. Восстановлены контексты безопасности **SELinux** для системных каталогов и каталога веб-контента.

5. Веб-сервер Apache был перезапущен для применения внесённых изменений.

6. На клиентской виртуальной машине в браузере выполнено обращение к веб-серверу по HTTPS.
   Подтверждено, что PHP-скрипты корректно обрабатываются и отображается страница с информацией о версии PHP и параметрах конфигурации.

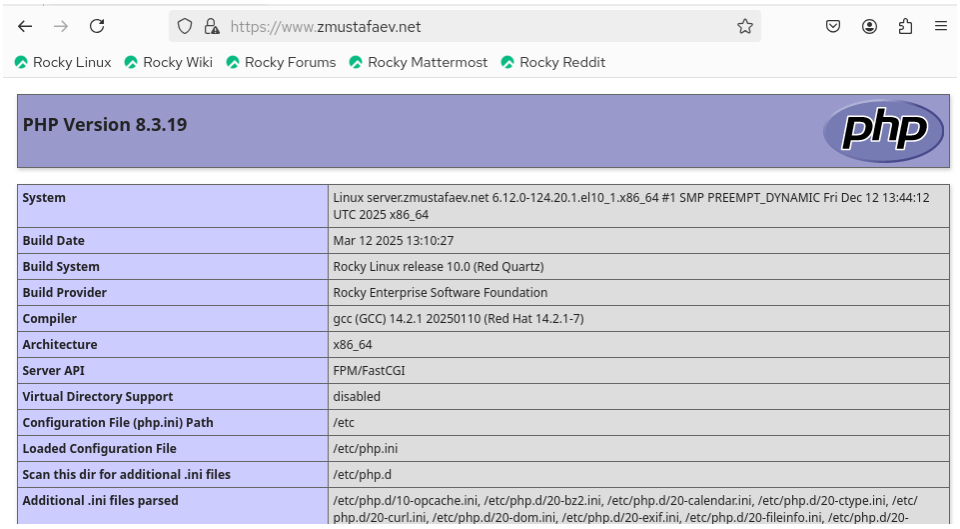

Рис. 2.8: Отображение страницы phpinfo в браузере

2.3 Внесение изменений в настройки внутреннего окружения виртуальной машины

1. Конфигурационные файлы веб-сервера, файлы веб-контента, а также SSL-ключ и сертификат были скопированы в каталог `/vagrant/provision/server/http` для обеспечения автоматического развёртывания окружения.

2. В сценарий автоматической настройки `http.sh` добавлены команды установки PHP, настройки межсетевого экрана для разрешения работы по протоколам HTTP и HTTPS, а также запуска и добавления сервиса Apache в автозагрузку.

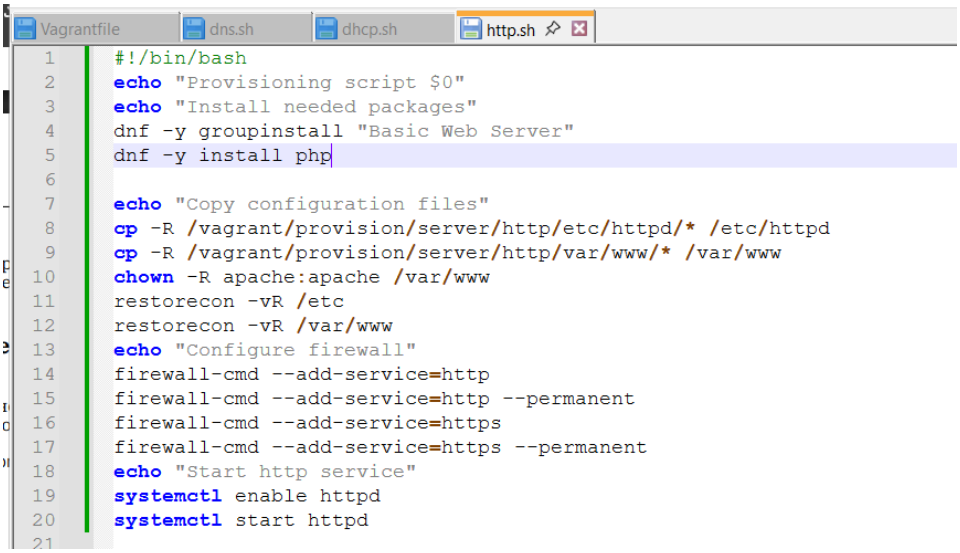

Рис. 2.9: Изменение сценария автоматической настройки http.sh

3 Вывод

В ходе работы был сконфигурирован HTTP-сервер Apache для работы по защищённому протоколу HTTPS. Сгенерированы самоподписанные SSL-ключ и сертификат, выполнена настройка виртуальных хостов с перенаправлением HTTP-запросов на HTTPS. Обеспечен доступ к веб-ресурсу по зашифрованному соединению. Дополнительно настроена поддержка PHP, проверена корректная обработка PHP-скриптов и автоматизация развёртывания конфигурации в среде Vagrant.

4 Контрольные вопросы

1. **В чём отличие HTTP от HTTPS?**
   HTTP передаёт данные в открытом виде, без шифрования, что делает их уязвимыми для перехвата. HTTPS использует SSL/TLS для шифрования соединения, обеспечивая конфиденциальность и целостность передаваемых данных.

2. **Каким образом обеспечивается безопасность контента веб-сервера при работе через HTTPS?**
   Безопасность обеспечивается за счёт шифрования трафика с использованием SSL/TLS, проверки подлинности сервера с помощью цифрового сертификата и защиты данных от изменения при передаче.

3. **Что такое сертификационный центр? Приведите пример.**
   Сертификационный центр — это организация, которая выпускает и подписывает цифровые сертификаты, подтверждающие подлинность веб-серверов. Примером сертификационного центра является Let’s Encrypt.

5 Список литературы

1. Apache HTTP Server Version 2.4 Documentation. — URL: http://httpd.apache.org/docs/current/ (дата обр. 13.09.2021).
2. Httpd — Apache Hypertext Transfer Protocol Server. — URL: https://httpd.apache.org/docs/2.4/pr (дата обр. 13.09.2021).