Отчёт по лабораторной работе 5

Расширенная настройкаHTTP-сервера Apache

Суннатилло Махмудов

Содержание

# 1 Цель работы

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

# 2 Теоретические сведения

**HTTPS (HyperText Transfer Protocol Secure)** — расширение протокола HTTP, обеспечивающее безопасную передачу данных между клиентом (браузером) и сервером за счет использования шифрования. Работает поверх TLS/SSL, защищая соединение от перехвата и подмены информации.

* **Клиент** — программа (обычно веб-браузер), отправляющая запросы к серверу.
* **Сервер** — веб-сервер, обрабатывающий запросы и отправляющий ответы.
* **SSL/TLS-сертификат** — цифровой сертификат, подтверждающий подлинность сайта и содержащий ключи для шифрования.
* **Шифрование** — обеспечивает конфиденциальность данных, передаваемых по сети.

## 2.1 Основные свойства HTTPS

* **Аутентификация** — подтверждает, что клиент общается именно с тем сервером, к которому обращался.
* **Конфиденциальность** — данные передаются в зашифрованном виде и недоступны для перехвата.
* **Целостность** — защищает информацию от подмены или изменения при передаче.

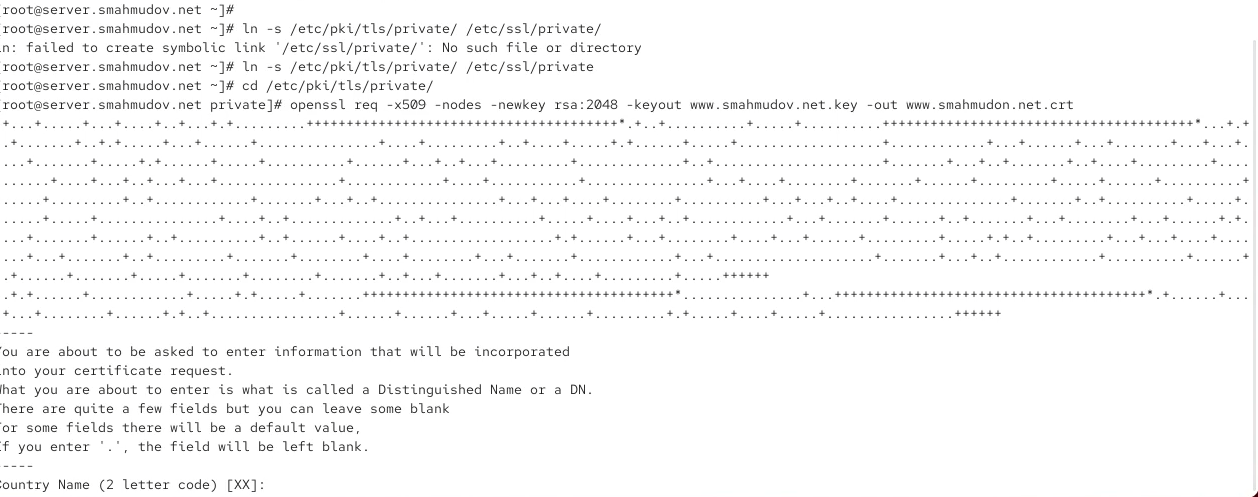
## 2.2 Этапы установления HTTPS-соединения

1. **Client Hello** — клиент отправляет список поддерживаемых шифров и параметров.
2. **Server Hello** — сервер выбирает параметры и передает свой SSL/TLS-сертификат.
3. **Обмен ключами** — клиент и сервер договариваются о сессионных ключах для шифрования.
4. **Secure Connection** — все дальнейшие данные передаются по зашифрованному каналу.

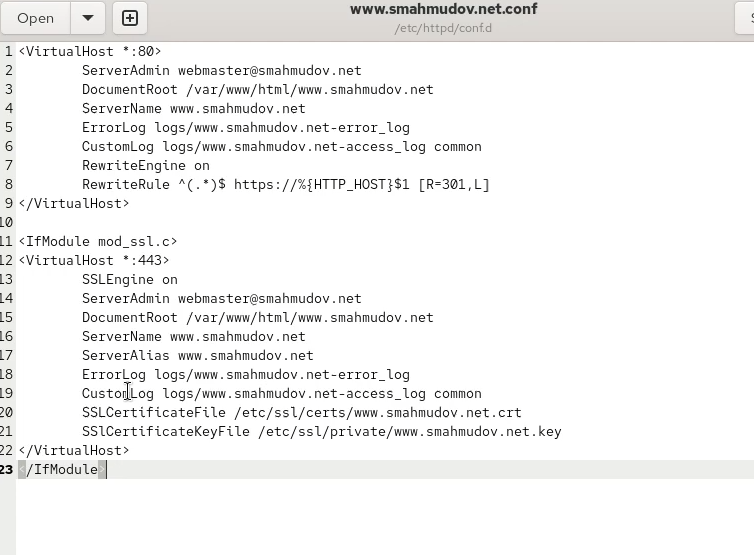
# 3 Выполнение лабораторной работы

## 3.1 Конфигурирование HTTP-сервера для работы через протокол HTTPS

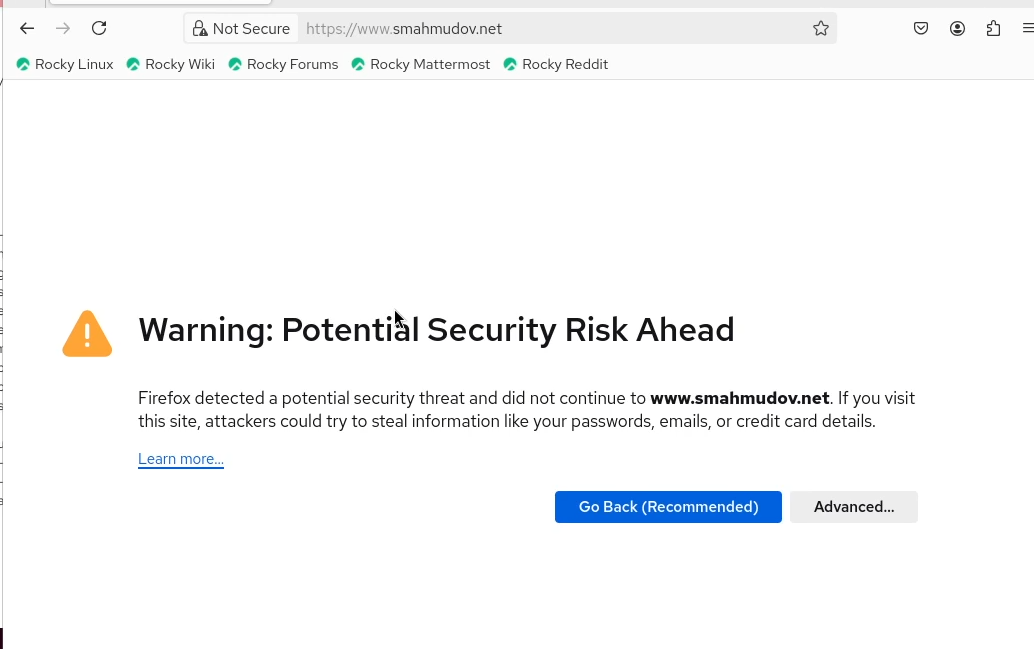
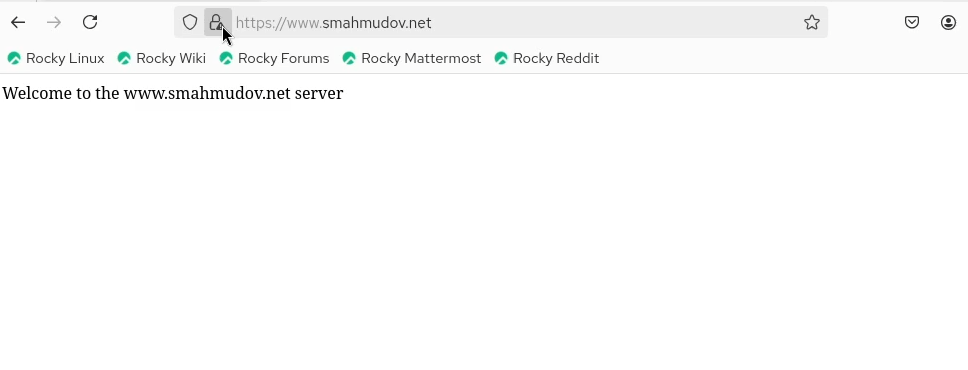
1. На виртуальной машине **server** был создан каталог для хранения ключей и сертификатов.  
   После этого сгенерированы приватный ключ и самоподписанный сертификат при помощи утилиты OpenSSL.

* 
* Рис. 1: Генерация ключа и сертификата
* В процессе генерации были указаны значения: страна — **RU**, город — **Moscow**, организация и подразделение — *smahmudov*, доменное имя — *smahmudov.net*, e-mail — *smahmudov@smahmudov.net*.

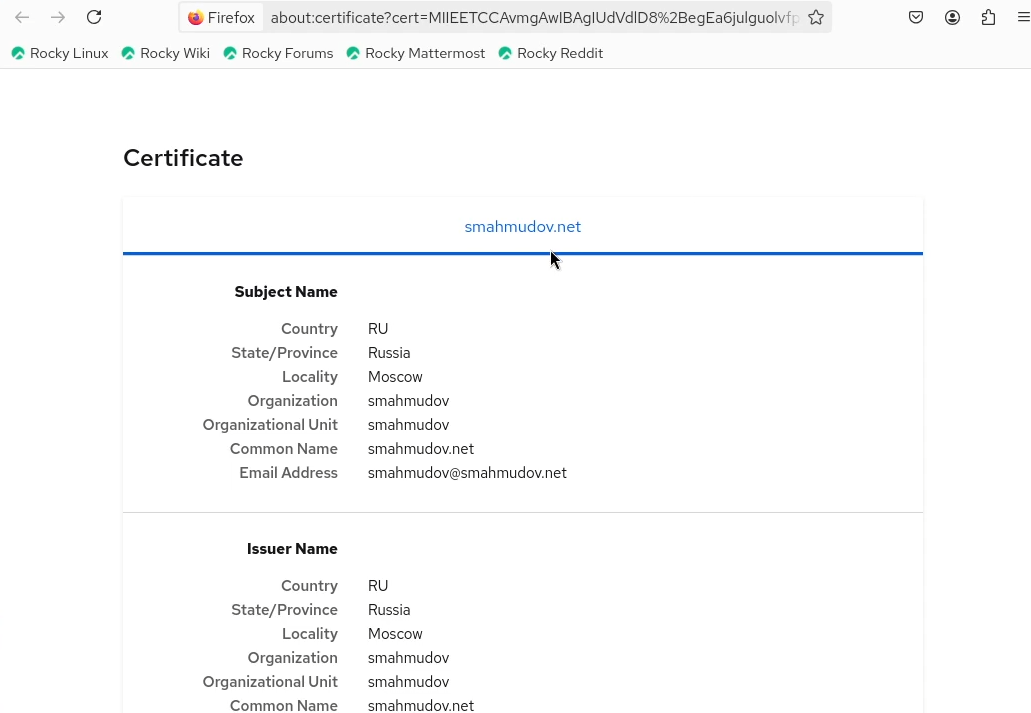
1. Был отредактирован конфигурационный файл **/etc/httpd/conf.d/www.smahmudov.net.conf**.  
   В него внесены следующие изменения:
   * для порта **80** настроено перенаправление всех HTTP-запросов на HTTPS;
   * определены директории для корня сайта и логов ошибок/доступа;
   * для порта **443** включён SSL и указаны пути к сертификату и приватному ключу;
   * заданы доменное имя сервера и алиас.

* 
* Рис. 2: Редактирование конфигурационного файла

1. После перезапуска веб-сервера при обращении к сайту по HTTPS браузер выдал предупреждение о небезопасном соединении, так как использован самоподписанный сертификат.

* 
* Рис. 3: Предупреждение браузера о небезопасном соединении
* После добавления исключения сайт открылся корректно по защищённому протоколу:
* 
* Рис. 4: Подключение к сайту по HTTPS

1. В настройках браузера был просмотрен сертификат. Он содержит следующие данные:
   * Страна: **RU**
   * Город: **Moscow**
   * Организация: **smahmudov**
   * Домен: **smahmudov.net**
   * E-mail: **smahmudov@smahmudov.net**

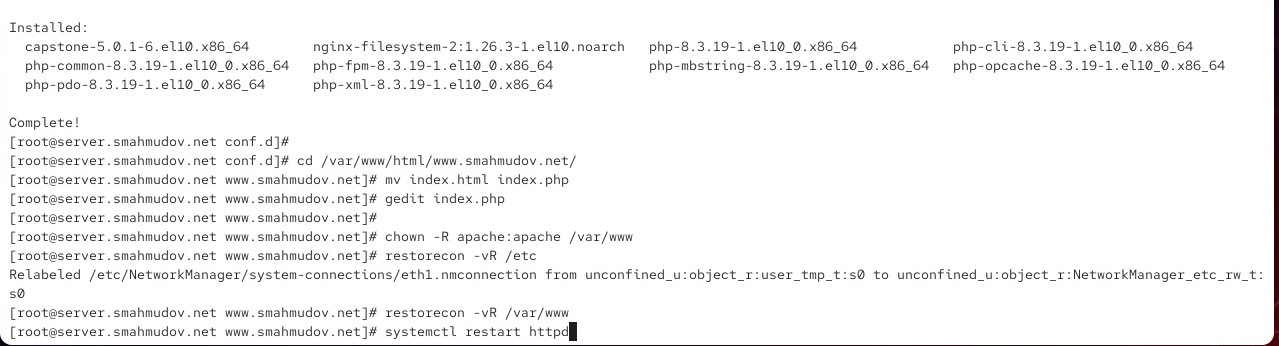
* 
* Рис. 5: Просмотр сертификата в браузере

## 3.2 Конфигурирование HTTP-сервера для работы с PHP

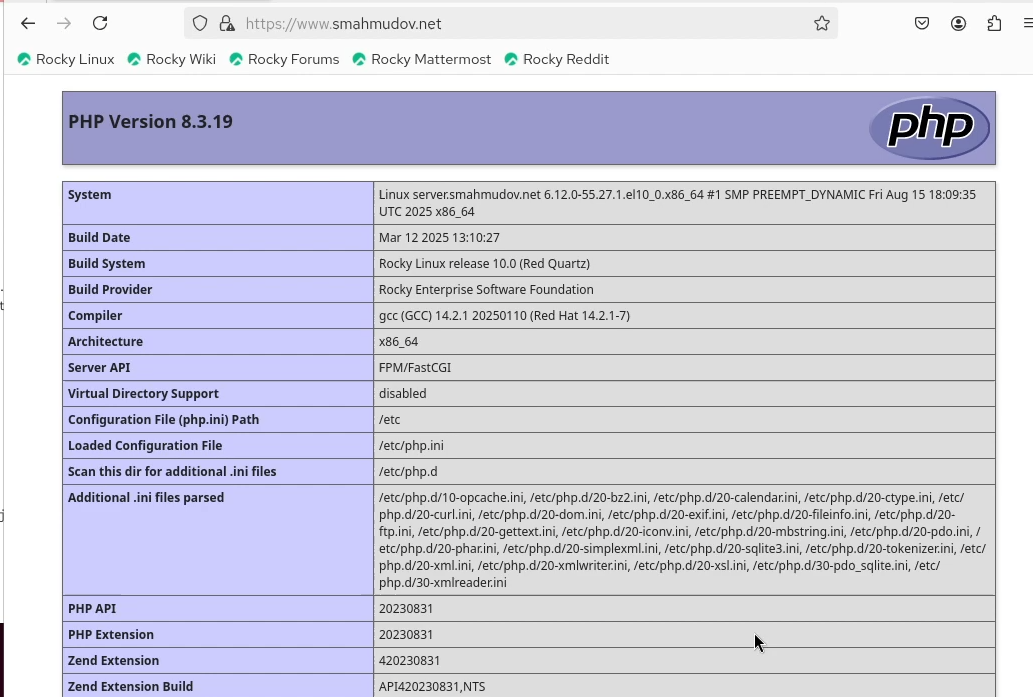
1. На сервер были установлены необходимые пакеты для поддержки PHP.

* 
* Рис. 6: Установка пакетов PHP

1. В каталоге **/var/www/html/www.smahmudov.net** файл *index.html* был заменён на *index.php* с минимальным содержимым для вывода информации о конфигурации PHP.

* 
* Рис. 7: Создание index.php с вызовом phpinfo()

1. Для корректной работы были скорректированы права доступа к каталогу с веб-контентом и восстановлен контекст SELinux. После этого веб-сервер был перезапущен.
2. При обращении к сайту **https://www.smahmudov.net** в браузере отобразилась служебная страница PHP с подробной информацией о версии и параметрах окружения.

* 
* Рис. 8: Вывод phpinfo() в браузере

## 3.3 Внесение изменений в настройки внутреннего окружения

1. Конфигурационные файлы веб-сервера и каталоги сайта были скопированы в структуру **/vagrant/provision/server/http** для сохранения и последующего использования. Также были добавлены сертификат и приватный ключ.

* 
* Рис. 9: Копирование конфигурационных файлов и сертификатов в окружение Vagrant

1. В скрипт **/vagrant/provision/server/http.sh** были внесены изменения:
   * добавлена установка PHP;
   * настроен межсетевой экран для разрешения работы с HTTPS.

# 4 Вывод

В ходе лабораторной работы был настроен веб-сервер **Apache** для работы через протокол **HTTPS** с использованием самоподписанного SSL-сертификата. Реализовано автоматическое перенаправление с HTTP на HTTPS, проверена корректность сертификата и установлено защищённое соединение. Дополнительно была выполнена установка и настройка **PHP**, что позволило протестировать работоспособность динамических страниц с помощью встроенной функции **phpinfo()**. Результаты подтвердили правильную конфигурацию веб-сервера и поддержку защищённого доступа.

# 5 Контрольные вопросы

1. **В чём отличие HTTP от HTTPS?**  
   HTTP передаёт данные в открытом виде и не обеспечивает защиты от перехвата или подмены.  
   HTTPS работает поверх протокола **TLS/SSL**, что позволяет шифровать весь трафик и гарантировать безопасное соединение между клиентом и сервером.
2. **Каким образом обеспечивается безопасность контента веб-сервера при работе через HTTPS?**  
   Безопасность достигается за счёт использования **шифрования**, **аутентификации** и **контроля целостности данных**. Для этого применяется SSL/TLS-сертификат, который подтверждает подлинность сервера и позволяет установить защищённый канал связи.
3. **Что такое сертификационный центр? Приведите пример.**  
   Сертификационный центр (Certificate Authority, CA) — это организация, которая выпускает и подписывает цифровые сертификаты, подтверждающие подлинность владельца ресурса.  
   Примеры: **DigiCert**, **GlobalSign**, **Let’s Encrypt**.

# 6 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/programs/httpd.html (дата обр. 13.09.2021).