

Отчет по лабораторной работе №5

Расширенная настройка HTTP-сервера Apache

Галацан Николай, НПИбд-01-22

Содержание

| | | |
|----------|---|-----------|
| 1 | Цель работы | 4 |
| 2 | Выполнение лабораторной работы | 5 |
| 2.1 | Конфигурирование HTTP-сервера для работы через протокол HTTPS | 5 |
| 2.2 | Конфигурирование HTTP-сервера для работы с РНР | 9 |
| 2.3 | Внесение изменений в настройки внутреннего окружения виртуальной машины | 10 |
| 3 | Выводы | 12 |
| 4 | Ответы на контрольные вопросы | 13 |

Список иллюстраций

| | | |
|------|---|----|
| 2.1 | Заполнение сертификата | 6 |
| 2.2 | Содержимое каталогов /etc/ssl/private и /etc/ssl/certs | 6 |
| 2.3 | Редактирование файла /etc/httpd/conf.d/www.ngalacan.net | 7 |
| 2.4 | Внесение изменений в настройки межсетевого экрана, перезапуск веб-сервера | 7 |
| 2.5 | Сообщение о незащищенности соединения | 8 |
| 2.6 | Информация о веб-странице | 8 |
| 2.7 | Содержимое сертификата | 9 |
| 2.8 | Замена файла /var/www/html/www.ngalacan.net/index.html на index.php | 9 |
| 2.9 | Редактирование index.php | 9 |
| 2.10 | Веб-страница с информацией об используемой версии PHP | 10 |
| 2.11 | Внесение изменений в скрипт http.sh | 11 |

1 Цель работы

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

2 Выполнение лабораторной работы

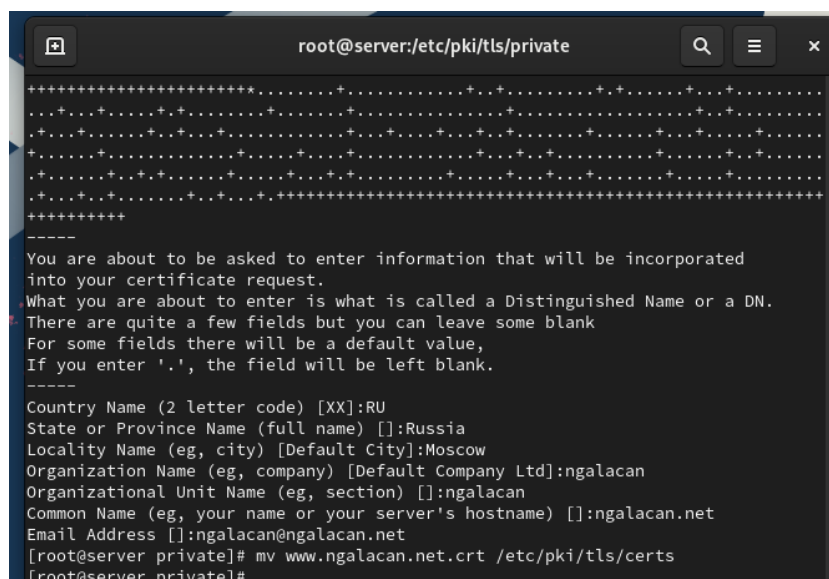
2.1 Конфигурирование HTTP-сервера для работы через протокол HTTPS

Запускаю VM через рабочий каталог. На VM server вхожу под собственным пользователем и перехожу в режим суперпользователя. В каталоге /etc/ssl создаю каталог private:

```
mkdir -p /etc/pki/tls/private  
ln -s /etc/pki/tls/private /etc/ssl/private  
cd /etc/pki/tls/private
```

Генерирую ключ и сертификат (рис. 2.1), введя следующую команду:

```
openssl req -x509 -nodes -newkey rsa:2048 -keyout www.ngalacan.net.key  
↪ -out www.ngalacan.net.crt
```



Сгенерированные ключ и сертификат появляются в соответствующем каталоге `/etc/ssl/private`. Копирую сертификат в каталог `/etc/ssl/certs` (рис. 2.2)

Редактирую конфигурационный файл `/etc/httpd/conf.d/www.ngalacan.net`
(рис. 2.3)



Рис. 2.3: Редактирование файла /etc/httpd/conf.d/www.ngalacan.net

Вношу изменения в настройки межсетевого экрана на сервере, перезапускаю веб-сервер (рис. 2.4)

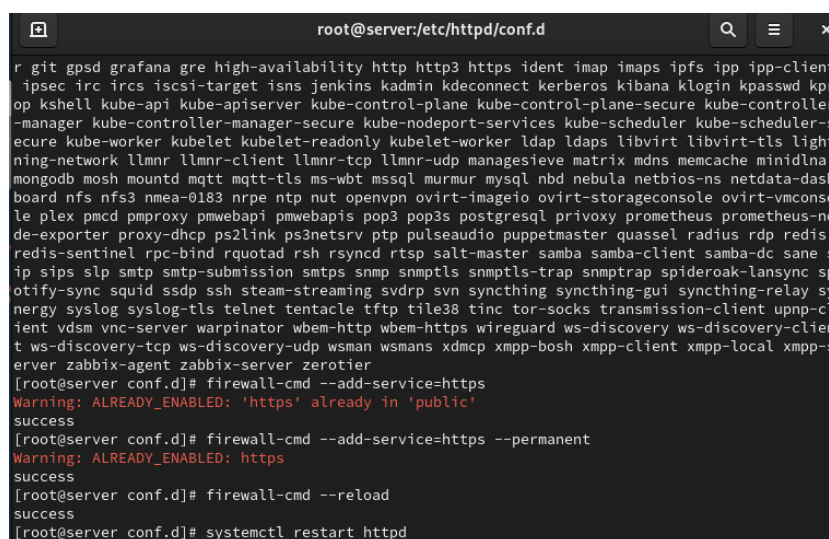


Рис. 2.4: Внесение изменений в настройки межсетевого экрана, перезапуск веб-сервера

На VM client открываю в браузере страницу www.ngalacan.net с сообщением о незащищенности соединения (рис. 2.5). Добавив страницу в исключения, просматри-
ваю информацию о сертификате (рис. 2.6), (рис. 2.7).

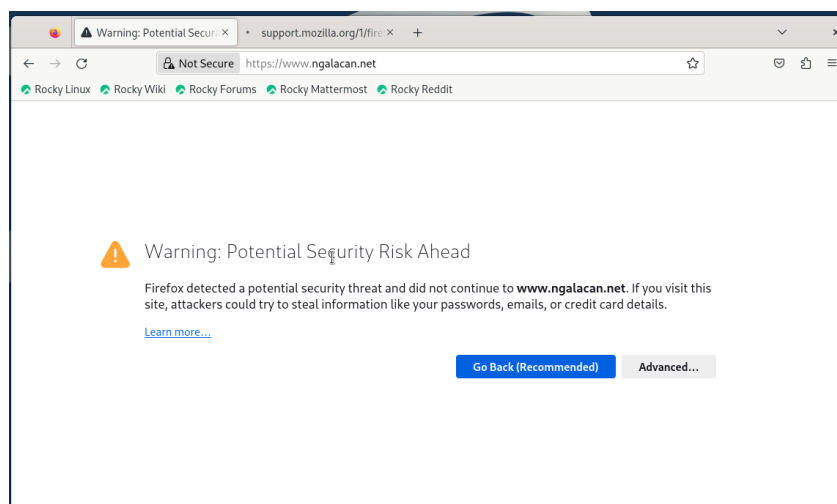


Рис. 2.5: Сообщение о незащищенности соединения

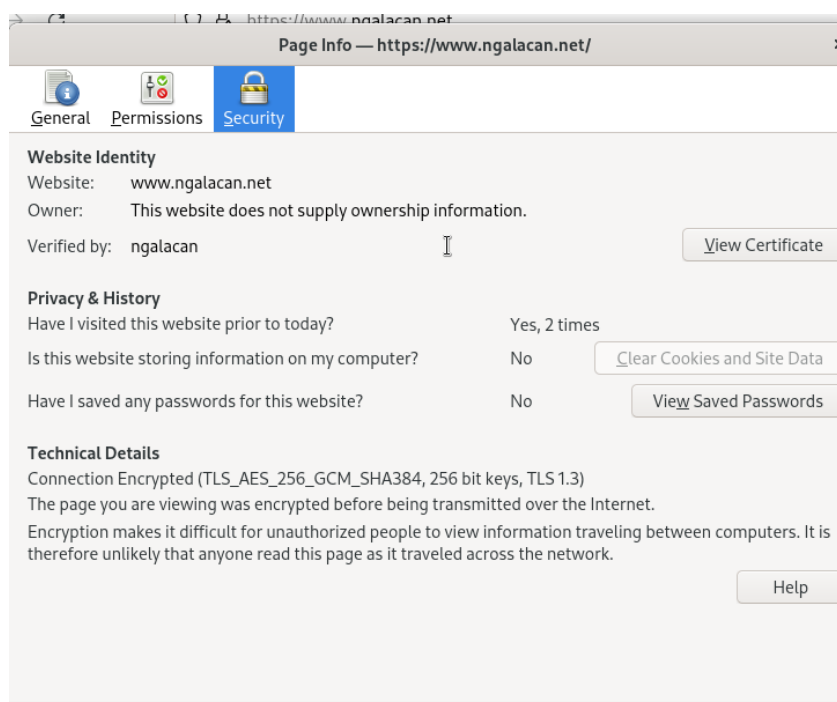


Рис. 2.6: Информация о веб-странице

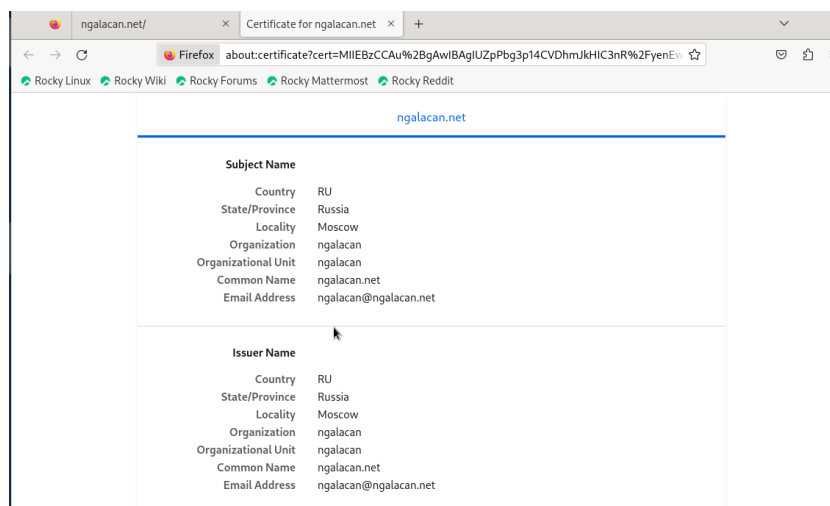


Рис. 2.7: Содержимое сертификата

2.2 Конфигурирование HTTP-сервера для работы с PHP

Устанавливаю пакеты для работы с PHP: `dnf -y install php`.

В каталоге `/var/www/html/www.ngalacan.net` заменяю `index.html` на `index.php` (рис. 2.8).

```
[root@server ~]# cd /var/www/html/www.ngalacan.net
[root@server www.ngalacan.net]# ls
index.html
[root@server www.ngalacan.net]# rm index.html
rm: remove regular file 'index.html'? y
[root@server www.ngalacan.net]# touch index.php
[root@server www.ngalacan.net]# gedit index.php
```

Рис. 2.8: Замена файла `/var/www/html/www.ngalacan.net/index.html` на `index.php`

Редактирую `index.php` (рис. 2.9).

```
*index.php
/var/www/html/www.ngalacan.net
1 ?php
2 phpinfo();
3 ?
```

Рис. 2.9: Редактирование `index.php`

Корректирую права доступа в каталог с веб-контентом, восстанавливаю контекст безопасности в SELinux, перезагружаю HTTP-сервер:

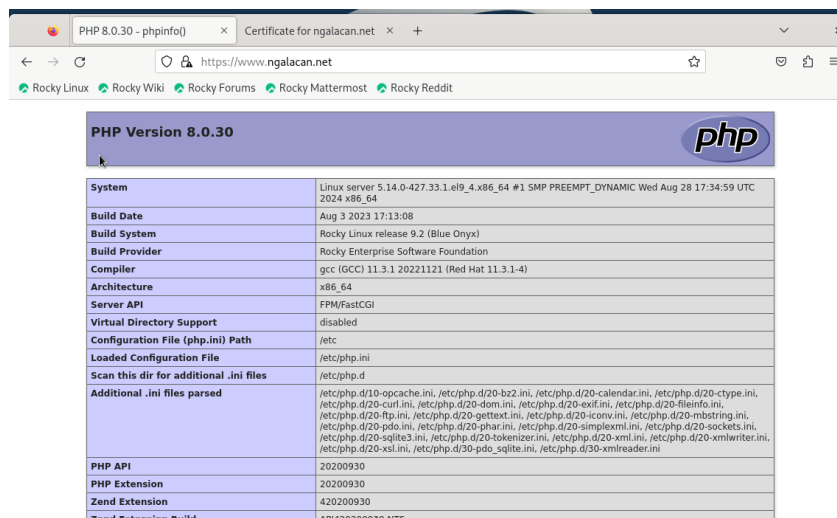
```
chown -R apache:apache /var/www
```

```
restorecon -vR /etc
```

```
restorecon -vR /var/www
```

```
systemctl restart httpd
```

На VM client ввожу в адресную строку браузера `www.ngalacan.net` и вижу веб-страницу с информацией об используемой версии PHP (рис. 2.10).



| | |
|---|--|
| System | Linux server 5.14.0-427.33.1.el9_4.x86_64 #1 SMP PREEMPT_DYNAMIC Wed Aug 28 17:34:59 UTC 2024 x86_64 |
| Build Date | Aug 3 2023 17:13:08 |
| Build System | Rocky Linux release 9.2 (Blue Onyx) |
| Build Provider | Rocky Enterprise Software Foundation |
| Compiler | gcc (GCC) 11.3.1 20221121 (Red Hat 11.3.1-4) |
| Architecture | x86_64 |
| Server API | FPM/FastCGI |
| Virtual Directory Support | disabled |
| Configuration File (php.ini) Path | /etc |
| Loaded Configuration File | /etc/php.ini |
| Scan this dir for additional .ini files | /etc/php.d |
| Additional .ini files parsed | /etc/php.d/10-opcache.ini, /etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-dom.ini, /etc/php.d/20-exif.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-mbstring.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-simplexml.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/20-xml.ini, /etc/php.d/20-xmlwriter.ini, /etc/php.d/20-xsl.ini, /etc/php.d/30-pdo_sqlite.ini, /etc/php.d/30-xmldrader.ini |
| PHP API | 20200930 |
| PHP Extension | 20200930 |
| Zend Extension | 420200930 |
| Zend Extension Build | ARM/32/20200930 |

Рис. 2.10: Веб-страница с информацией об используемой версии PHP

2.3 Внесение изменений в настройки внутреннего окружения виртуальной машины

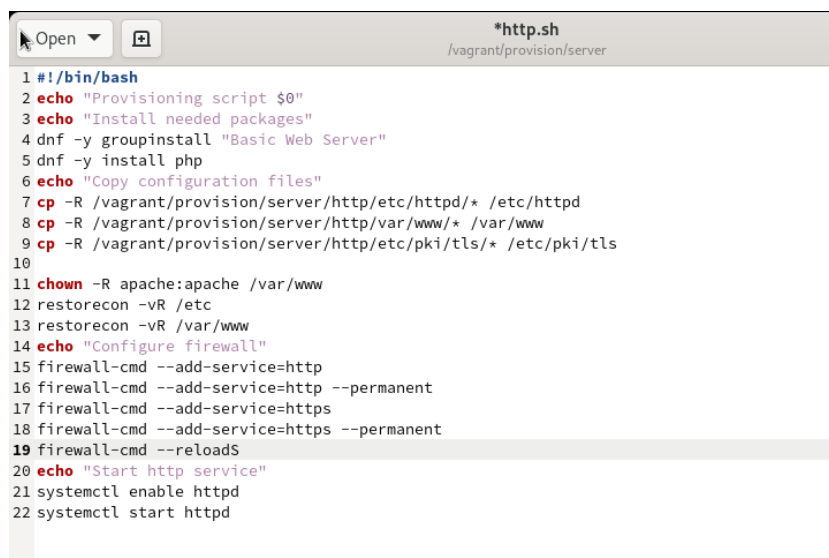
На VM server перехожу в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/` и копирую в соответствующие каталоги конфигурационные файлы:

```
cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/etc/httpd/conf.d
```

```
cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
```

```
mkdir -p /vagrant/provision/server/http/etc/pki/tls/private
mkdir -p /vagrant/provision/server/http/etc/pki/tls/certs
cp -R /etc/pki/tls/private/www.user.net.key
    ↪ /vagrant/provision/server/http/etc/pki/tls/private
cp -R /etc/pki/tls/certs/www.user.net.crt
    ↪ /vagrant/provision/server/http/etc/pki/tls/certs
```

В скрипт `/vagrant/provision/server/http.sh` вношу изменения, добавив установку PHP и настройку межсетевого экрана для работы с `https` (рис. 2.11).



```
*http.sh
/vagrant/provision/server

1 #!/bin/bash
2 echo "Provisioning script $0"
3 echo "Install needed packages"
4 dnf -y groupinstall "Basic Web Server"
5 dnf -y install php
6 echo "Copy configuration files"
7 cp -R /vagrant/provision/server/http/etc/httpd/* /etc/httpd
8 cp -R /vagrant/provision/server/http/var/www/* /var/www
9 cp -R /vagrant/provision/server/http/etc/pki/tls/* /etc/pki/tls
10
11 chown -R apache:apache /var/www
12 restorecon -vR /etc
13 restorecon -vR /var/www
14 echo "Configure firewall"
15 firewall-cmd --add-service=http
16 firewall-cmd --add-service=http --permanent
17 firewall-cmd --add-service=https
18 firewall-cmd --add-service=https --permanent
19 firewall-cmd --reload
20 echo "Start http service"
21 systemctl enable httpd
22 systemctl start httpd
```

Рис. 2.11: Внесение изменений в скрипт `http.sh`

3 Выводы

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

4 Ответы на контрольные вопросы

1. В чём отличие HTTP от HTTPS?

- **HTTP** (HyperText Transfer Protocol) – это протокол передачи данных, который используется для передачи информации между клиентом (например, веб-браузером) 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?

- 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?

- 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 веб-сайтов. Владелец веб-сайтов могут получить сертификат от Let’s Encrypt, чтобы обеспечить шифрование и подтвердить свою легитимность в онлайн-среде.