

Отчет по лабораторной работе №5

Дисциплина: Администрирование сетевых подсистем

Иванов Сергей Владимирович

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
3.1	Конфигурирование HTTP-сервера для работы через протокол HTTPS	6
3.2	Конфигурирование HTTP-сервера для работы с РНР	10
3.3	Внесение изменений в настройки внутреннего окружения виртуальной машины	11
4	Ответы на контрольные вопросы	14
5	Выводы	16

Список иллюстраций

3.1	Запуск server	6
3.2	Создание каталога private	6
3.3	Создание ключа и сертификата	7
3.4	Копирование сертификата	7
3.5	Редактирование конф. файла	8
3.6	Меняем настройки firewall	8
3.7	Перезапуск сервера	8
3.8	Доступ к веб-серверу по HTTPS	9
3.9	Добавление в исключение	9
3.10	Содержание сертификата	10
3.11	Установка PHP	10
3.12	Файл index.php	11
3.13	Корректирование прав и перезапуск сервера	11
3.14	Страница с PHP	12
3.15	Создание каталогов и копирование файлов	12
3.16	Изменение скрипта http.sh	13

1 Цель работы

Целью этой работы является приобретение практических навыков по расширенному конфигурированию HTTPсервера Apache в части безопасности и возможности использования PHP.

2 Задание

1. Сгенерируйте криптографический ключ и самоподписанный сертификат безопасности для возможности перехода веб-сервера от работы через протокол HTTP к работе через протокол HTTPS (см. раздел 5.4.1).
2. Настройте веб-сервер для работы с PHP (см. раздел 5.4.2).
3. Напишите (или скорректируйте) скрипт для Vagrant, фиксирующий действия по расширенной настройке HTTP-сервера во внутреннем окружении виртуальной машины `server` (см. раздел 5.4.3).

3 Выполнение лабораторной работы

3.1 Конфигурирование HTTP-сервера для работы через протокол HTTPS

Загрузим операционную систему и перейдем в рабочий каталог с проектом: `cd /var/tmp/user_name/vagrant` . Запустим виртуальную машину `server: vagrant up server` . (рис. 1).

```
C:\Users\lserg>cd C:\work_asp\svivanov\vagrant
C:\work_asp\svivanov\vagrant>vagrant up server
Bringing machine 'server' up with 'virtualbox' provider...
==> server: You assigned a static IP ending in ".1" or ":1" to this machi
==> server: This is very often used by the router and can cause the
```

Рис. 3.1: Запуск server

На виртуальной машине `server` войдем под пользователем и откроем терминал. Перейдем в режим суперпользователя. 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. 2).
```

```
[svivanov@server ~]$ sudo -i
[sudo] пароль для svivanov:
[root@server.svivanov.net ~]# mkdir -p /etc/pki/tls/private
[root@server.svivanov.net ~]# ln -s /etc/pki/tls/private /etc/ssl/private
[root@server.svivanov.net ~]# cd /etc/pki/tls/private
[root@server.svivanov.net private]#
```

Рис. 3.2: Создание каталога private

Сгенерируем ключ и сертификат, используя следующую команду:

```
openssl req -x509 -nodes -newkey rsa:2048 -keyout www.svivanov.net.key -out  
www.svivanov.net.crt  
mv www.svivanov.net.crt /etc/pki/tls/certs (рис. 3)
```

```
-----  
Country Name (2 letter code) [XX]:RU  
State or Province Name (full name) []:Russia  
Locality Name (eg, city) [Default City]:Moscow  
Organization Name (eg, company) [Default Company Ltd]:svivanov  
Organizational Unit Name (eg, section) []:svivanov  
Common Name (eg, your name or your server's hostname) []:svivanov.net  
Email Address []:svivanov@svivanov.net  
[root@server.svivanov.net private]#
```

Рис. 3.3: Создание ключа и сертификата

Сгенерированные ключ и сертификат появятся в соответствующем каталоге /etc/ssl/private. Скопируем сертификат в каталог /etc/ssl/certs: `cp /etc/ssl/private/www.svivanov.net.crt /etc/ssl/cert/` (рис. 4)

```
[root@server.svivanov.net certs]# cp /etc/ssl/private/www.svivanov.net.crt /etc/ssl/certs/  
[root@server.svivanov.net certs]#
```

Рис. 3.4: Копирование сертификата

Для перехода веб-сервера `www.svivanov.net` на функционирование через протокол HTTPS требуется изменить его конфигурационный файл. Перейдем в каталог с конфигурационными файлами: `cd /etc/httpd/conf.d`. Откроем на редактирование файл `/etc/httpd/conf.d/www.svivanov.net.conf` и заменим его содержимое на следующее: (рис. 5)

```

<VirtualHost *:80>
    ServerAdmin webmaster@svivanov.net
    DocumentRoot /var/www/html/www.svivanov.net
    ServerName www.svivanov.net
    ServerAlias www.svivanov.net
    ErrorLog logs/www.svivanov.net-error_log
    CustomLog logs/www.svivanov.net-access_log common
    RewriteEngine on
    RewriteRule ^(.*)$ https://%{HTTP_HOST}$1 [R=301,L]
</VirtualHost>

<IfModule mod_ssl.c>
<VirtualHost *:443>
    SSLEngine on
    ServerAdmin webmaster@svivanov.net
    DocumentRoot /var/www/html/www.svivanov.net
    ServerName www.svivanov.net
    ServerAlias www.svivanov.net
    ErrorLog logs/www.svivanov.net-error_log
    CustomLog logs/www.svivanov.net-access_log common
    SSLCertificateFile /etc/ssl/certs/www.svivanov.net.crt
    SSLCertificateKeyFile /etc/ssl/private/www.svivanov.net.key
</VirtualHost>
</IfModule>

```

Рис. 3.5: Редактирование конф. файла

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 (рис. 6)

```

```

[root@server.svivanov.net conf.d]# firewall-cmd --add-service=https
success
[root@server.svivanov.net conf.d]# firewall-cmd --add-service=https --permanent
success
[root@server.svivanov.net conf.d]# firewall-cmd --reload
success
[root@server.svivanov.net conf.d]#

```

Рис. 3.6: Меняем настройки firewall

Перезапустим веб-сервер: `systemctl restart httpd` (рис. 7)

```

[root@server.svivanov.net certs]# systemctl restart httpd
[root@server.svivanov.net certs]#

```

Рис. 3.7: Перезапуск сервера

На виртуальной машине client в строке браузера введем название веб-сервера

www.svivanov.net и убедимся, что произойдёт автоматическое переключение на работу по протоколу HTTPS. (рис. 8)

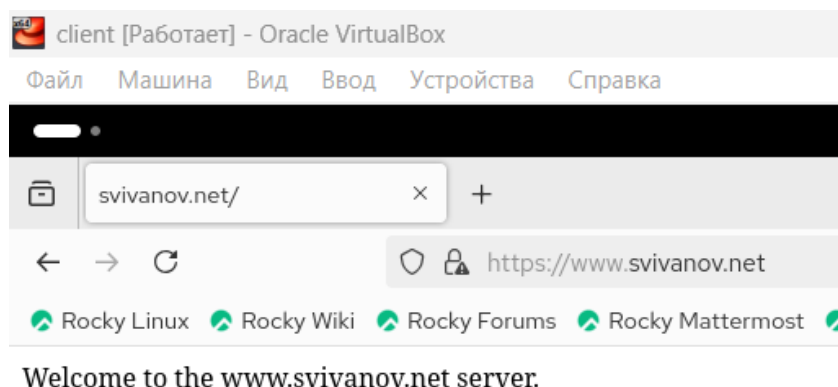


Рис. 3.8: Доступ к веб-серверу по HTTPS

На открывшейся странице с сообщением о незащищённости соединения нажмем кнопку «Дополнительно», затем добавим адрес вашего сервера в постоянные исключения. (рис. 9)

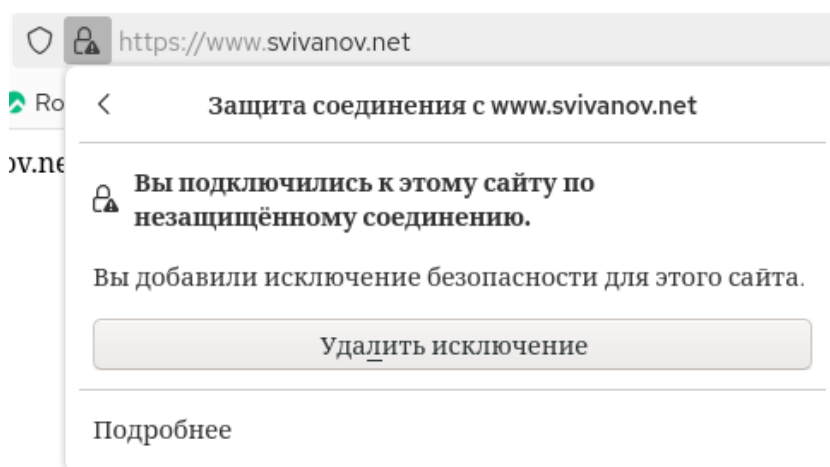


Рис. 3.9: Добавление в исключение

Затем посмотрим содержание сертификата. (рис. 10)

Сертификат

svivanov.net	
Субъект	
Страна	RU
Область/Регион	Russia
Населённый пункт	Moscow
Организация	svivanov
Подразделение	svivanov
Общее имя	svivanov.net
Адрес электронной почты	svivanov@svivanov.net
Издатель	
Страна	RU
Область/Регион	Russia
Населённый пункт	Moscow
Организация	svivanov
Подразделение	svivanov
Общее имя	svivanov.net
Адрес электронной почты	svivanov@svivanov.net
Срок действия	
Действителен с	Mon, 29 Sep 2025 09:54:11 GMT
Действителен по	Wed, 29 Oct 2025 09:54:11 GMT

Рис. 3.10: Содержание сертификата

3.2 Конфигурирование HTTP-сервера для работы с PHP

Установим пакеты для работы с PHP: `dnf -y install php`. (рис. 11)

```
[root@server.svivanov.net conf.d]# dnf -y install php
Rocky Linux 10 - BaseOS                               263 B/s | 3.9 kB  00:15
Rocky Linux 10 - BaseOS                               1.2 MB/s | 19 MB  00:16
Rocky Linux 10 - AppStream                             6.0 kB/s | 3.9 kB  00:00
Rocky Linux 10 - AppStream                             1.0 MB/s | 2.1 MB  00:02
Rocky Linux 10 - Extras                                4.8 kB/s | 3.1 kB  00:00
Rocky Linux 10 - Extras                                5.8 kB/s | 5.4 kB  00:00
Dependencies resolved.
```

Рис. 3.11: Установка PHP

В каталоге `/var/www/html/www.svivanov.net` заменим файл `index.html` на `index.php` следующего содержания: (рис. 12)

```
<?php
phpinfo();
?>
```

Рис. 3.12: Файл index.php

Скорректируем 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 (рис. 13)
```

```
[root@server.svivanov.net www.svivanov.net]# chown -R apache:apache /var/www
[root@server.svivanov.net www.svivanov.net]# restorecon -vR /etc
Relabeled /etc/NetworkManager/system-connections/eth1.nmconnection from unconfined_u:object_r
confined_u:object_r:NetworkManager_etc_rw_t:s0
[root@server.svivanov.net www.svivanov.net]# restorecon -vR /var/www
[root@server.svivanov.net www.svivanov.net]# systemctl restart httpd
[root@server.svivanov.net www.svivanov.net]#
```

Рис. 3.13: Корректирование прав и перезапуск сервера

3.3 Внесение изменений в настройки внутреннего окружения виртуальной машины

На виртуальной машине client в строке браузера введем название веб-сервера `www.svivanov.net` и убедимся, что будет выведена страница с информацией об используемой на веб-сервере версии PHP. (рис. 14)

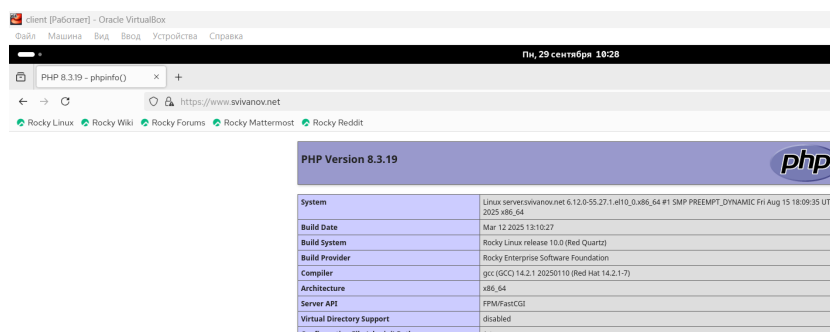


Рис. 3.14: Страница с PHP

На виртуальной машине server перейдем в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/http` и в соответствующие каталоги скопируем 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.svivanov.net.key /vagrant/provision/server/http/etc/pki/tls/private
```

```
cp -R /etc/pki/tls/certs/www.svivanov.net.crt /vagrant/provision/server/http/etc/pki/tls/certs
```

(рис. 15)

```
[root@server.svivanov.net ~]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
cp: overwrite '/vagrant/provision/server/http/var/www/html/server.svivanov.net/index.html'? y
[root@server.svivanov.net ~]# mkdir -p /vagrant/provision/server/http/etc/pki/tls/private
[root@server.svivanov.net ~]# mkdir -p /vagrant/provision/server/http/etc/pki/tls/certs
[root@server.svivanov.net ~]# cp -R /etc/pki/tls/private/www.svivanov.net.key
cp: missing destination file operand after '/etc/pki/tls/private/www.svivanov.net.key'
Try 'cp --help' for more information.
[root@server.svivanov.net ~]# cp -R /etc/pki/tls/private/www.svivanov.net.key /vagrant/provision/server/http/etc/pki/tls/private
[root@server.svivanov.net ~]# cp -R /etc/pki/tls/certs/www.user.net.crt /vagrant/provision/server/http/etc/pki/tls/certs
cp: cannot stat '/etc/pki/tls/certs/www.user.net.crt': Нет такого файла или каталога
[root@server.svivanov.net ~]# cp -R /etc/pki/tls/certs/www.svivanov.net.crt /vagrant/provision/server/http/etc/pki/tls/certs
[root@server.svivanov.net ~]#
```

Рис. 3.15: Создание каталогов и копирование файлов

В имеющийся скрипт `/vagrant/provision/server/http.sh` внесем изменения, добавив установку PHP и настройку межсетевого экрана, разрешающую работать с https. (рис. 16)

```
#!/bin/bash
echo "Provisioning script $0"
echo "Install needed packages"
dnf -y groupinstall "Basic Web Server"
dnf -y install php
echo "Copy configuration files"
cp -R /vagrant/provision/server/http/etc/httpd/* /etc/httpd
cp -R /vagrant/provision/server/http/var/www/* /var/www
chown -R apache:apache /var/www
restorecon -vR /etc
restorecon -vR /var/www
echo "Configure firewall"
firewall-cmd --add-service=http
firewall-cmd --add-service=http --permanent
firewall-cmd --add-service=https
firewall-cmd --add-service=https --permanent
echo "Start http service"
systemctl enable httpd
systemctl start httpd
```

Рис. 3.16: Изменение скрипта http.sh

4 Ответы на контрольные вопросы

1. В чём отличие HTTP от HTTPS?

HTTP - это стандартный протокол для передачи данных, который не использует шифрование. Все данные (логины, пароли) передаются в открытом виде.

HTTPS - это безопасная версия HTTP, которая шифрует весь трафик между браузером и сервером с помощью протокола SSL/TLS.

Отличие в том, что HTTPS обеспечивает конфиденциальность и целостность данных, а HTTP - нет.

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

Безопасность обеспечивается протоколом SSL/TLS по трем направлениям:

1. Шифрование: Все данные передаются в зашифрованном виде, что защищает их от перехвата.
2. Аутентификация: Сервер предъявляет браузеру цифровой сертификат, подтверждающий его подлинность.
3. Целостность: Специальные механизмы гарантируют, что данные не были изменены при передаче.

3. Что такое сертификационный центр? Приведите пример.

Сертификационный центр (Certificate Authority, CA) - это доверенная организация, которая выпускает цифровые сертификаты для веб-сайтов.

Он проверяет владельца домена и digitally подписывает сертификат. Браузеры доверяют сертификатам от известных ЦС, чьи корневые сертификаты в них предустановлены.

Примеры: Let's Encrypt (бесплатный), DigiCert, Sectigo.

5 Выводы

В ходе выполнения лабораторной работы мы приобрели практические навыки по расширенному конфигурированию HTTPсервера Apache в части безопасности и возможности использования PHP.