

Лабораторная работа №5

Расширенная настройка HTTP-сервера Apache

Беличева Дарья Михайловна

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
4	Выводы	16
5	Контрольные вопросы	17

Список иллюстраций

3.1	Создание каталога private	6
3.2	Генерация ключа и сертификата	6
3.3	Наличие ключа в каталоге	7
3.4	Наличие сертификата в каталоге	8
3.5	Содержимое сертификата	8
3.6	Редактирование файла	10
3.7	Настройка межсетевого экрана на сервере	10
3.8	Сообщение о незащищенности на сайте	11
3.9	Добавление адреса сервера в исключения	11
3.10	Содержание сертификата	12
3.11	Установка пакетов для работы с php	12
3.12	Редактирование файла index.php	13
3.13	Права доступа и контекст безопасности в SELinux	13
3.14	Содержание сайта	14
3.15	Внесения изменений в настройки внутреннего окружения	14
3.16	Редактирование скрипта	15

1 Цель работы

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

2 Задание

1. Сгенерировать криптографический ключ и самоподписанный сертификат безопасности для возможности перехода веб-сервера от работы через протокол HTTP к работе через протокол HTTPS;
2. Настроить веб-сервер для работы с PHP;
3. Написать скрипт для Vagrant, фиксирующий действия по расширенной настройке HTTP-сервера во внутреннем окружении виртуальной машины server.

3 Выполнение лабораторной работы

Конфигурирование HTTP-сервера для работы через протокол HTTPS

Загрузим вашу операционную систему и перейдем в рабочий каталог с проектом: `cd C:\Users\dasha\work\study\dmbelicheva\vagrant`

Запустим виртуальную машину `server: make server-up`

На виртуальной машине `server` войдем под своим пользователем и откроем терминал. Перейдем в режим суперпользователя: `sudo -i`

В каталоге `/etc/ssl` создадим каталог `private`.

```
[dmbelicheva@server.dmbelicheva.net ~]$ sudo -i
[sudo] password for dmbelicheva:
[root@server.dmbelicheva.net ~]# mkdir -p /etc/pki/tls/private
[root@server.dmbelicheva.net ~]# ln -s /etc/pki/tls/private /etc/ssl/private
ln: target '/etc/ssl/private': No such file or directory
[root@server.dmbelicheva.net ~]# ln -s /etc/pki/tls/private /etc/ssl/private
[root@server.dmbelicheva.net ~]# cd /etc/pki/tls/private
```

Рис. 3.1: Создание каталога `private`

Сгенерируем ключ и сертификат:

```
[root@server.dmbelicheva.net private]# openssl req -x509 -nodes -newkey rsa:2048 -keyout www.dmbelicheva.net.key -out www.dmbelicheva.net.crt
.....
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
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]:dmBelicheva
Organizational Unit Name (eg, section) []:dmBelicheva
Common Name (eg, your name or your server's hostname) []:dmBelicheva.net
Email Address []:dmBelicheva@dmBelicheva.net
[root@server.dmbelicheva.net private]#
```

Рис. 3.2: Генерация ключа и сертификата

- `req -x509` означает, что используется запрос подписи сертификата x509 (CSR);
- параметр `-nodes` указывает OpenSSL, что нужно пропустить шифрование сертификата SSL с использованием парольной фразы, т.е. позволить Apache читать файл без какого-либо вмешательства пользователя (без ввода пароля при попытке доступа к странице, в частности);
- параметр `-newkey rsa: 2048` указывает, что одновременно создаются новый ключ и новый сертификат, причём используется 2048-битный ключ RSA;
- параметр `-keyout` указывает, где хранить сгенерированный файл закрытого ключа при создании;
- параметр `-out` указывает, где разместить созданный сертификат SSL. Далее требуется заполнить сертификат:

Сгенерированные ключ и сертификат появились в соответствующем каталогах `/etc/ssl/private` и `/etc/ssl/certs`.

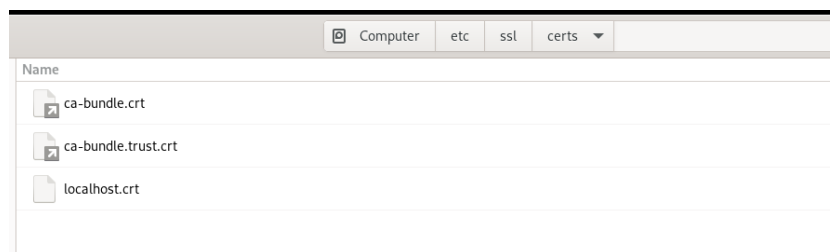


Рис. 3.3: Наличие ключа в каталоге

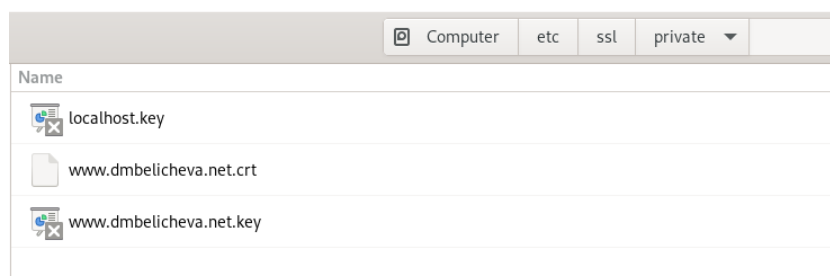


Рис. 3.4: Наличие сертификата в каталоге

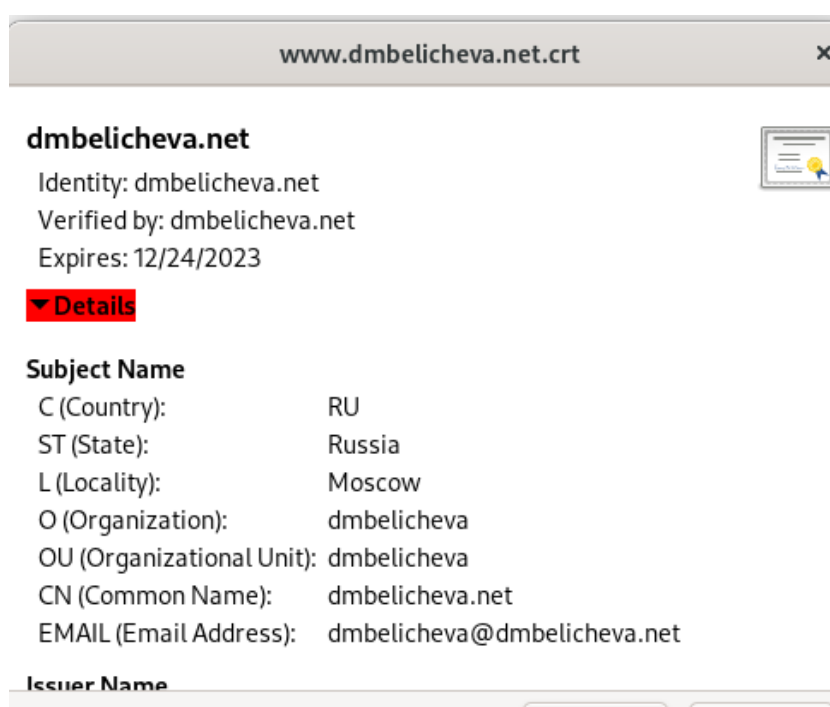


Рис. 3.5: Содержимое сертификата

Для перехода веб-сервера `www.dmbelicheva.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.dmbelicheva.net.conf` и заменим его содержимое на следующее:

```
<VirtualHost *:80>
```



```

ServerAdmin webmaster@dmbelicheva.net
DocumentRoot /var/www/html/www.dmbelicheva.net
ServerName www.dmbelicheva.net
ServerAlias www.dmbelicheva.net
ErrorLog logs/www.dmbelicheva.net-error_log
CustomLog logs/www.dmbelicheva.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@dmbelicheva.net
    DocumentRoot /var/www/html/www.dmbelicheva.net
    ServerName www.dmbelicheva.net
    ServerAlias www.dmbelicheva.net
    ErrorLog logs/www.dmbelicheva.net-error_log
    CustomLog logs/www.dmbelicheva.net-access_log common
    SSLCertificateFile /etc/ssl/certs/www.dmbelicheva.net.crt
    SSLCertificateKeyFile /etc/ssl/private/www.dmbelicheva.net.key
</VirtualHost>
</IfModule>

```

```
root@server:/etc/httpd/conf.d
GNU nano 5.6.1 www.dmbelicheva.net.conf
<VirtualHost *:80>
    ServerAdmin webmaster@dmbelicheva.net
    DocumentRoot /var/www/html/www.dmbelicheva.net
    ServerName www.dmbelicheva.net
    ServerAlias www.dmbelicheva.net
    ErrorLog logs/www.dmbelicheva.net-error_log
    CustomLog logs/www.dmbelicheva.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@dmbelicheva.net
    DocumentRoot /var/www/html/www.dmbelicheva.net
    ServerName www.dmbelicheva.net
    ServerAlias www.dmbelicheva.net
    ErrorLog logs/www.dmbelicheva.net-error_log
    CustomLog logs/www.dmbelicheva.net-access_log common
    SSLCertificateFile /etc/ssl/certs/www.dmbelicheva.net.crt
    SSLCertificateKeyFile /etc/ssl/private/www.dmbelicheva.net.key
</VirtualHost>
</IfModule>
```

Рис. 3.6: Редактирование файла

Внесем изменения в настройки межсетевого экрана на сервере, разрешив работу с https. Перезапустим веб-сервер: `systemctl restart httpd`.

```
[root@server.dmbelicheva.net conf.d]# firewall-cmd --list-services
cockpit dhcp dhcpv6-client dns http ssh
[root@server.dmbelicheva.net conf.d]# firewall-cmd --get-services
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqp amqps apcupsd audit ausweisapp2 ba
k-agent cockpit collectd condor-collector cratedb ctddb dhcp dhcpv6 dhcpv6-client distcc dns dns-over-tls docke
p freeipa-ldaps freeipa-replication freeipa-trust ftp galera ganglia-client ganglia-master git gpsd grafana gre
in kdeconnect kerberos kibana klogon kpasswd kprop kshell kube-api kube-apiserver kube-control-plane kube-contro
ure kube-worker kubelet kubelet-readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-network llmnr l
etbios-ns netdata-dashboard nfs nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ovirt-storageconsole ovirt-vmc
tsrv ptp pulseaudio puppetmaster quassel radius rdp redis redis-sentinel rpc-bind rquotad rsh rsyncd rtsp salt-m
nsync spotify-sync squid sssd ssh steam-streaming svdrp svn syncthing syncthing-gui synergy syslog syslog-tls te
covery ws-discovery-client ws-discovery-tcp ws-discovery-udp wsman wsmans xdmcp xmpp-bosh xmpp-client xmpp-local
[root@server.dmbelicheva.net conf.d]# firewall-cmd --add-service=https
success
[root@server.dmbelicheva.net conf.d]# firewall-cmd --add-service=https --permanent
success
[root@server.dmbelicheva.net conf.d]# firewall-cmd --reload
success
[root@server.dmbelicheva.net conf.d]# systemctl restart httpd
```

Рис. 3.7: Настройка межсетевого экрана на сервере

На виртуальной машине client в строке браузера введем название веб-сервера `www.user.net` и убедимся, что произойдёт автоматическое переключение на работу по протоколу HTTPS. На открывшейся странице с сообщением о незащищённости соединения нажмем кнопку «Дополнительно», затем добавим адрес сервера в постоянные исключения. Затем посмотрим содержание сертификата.

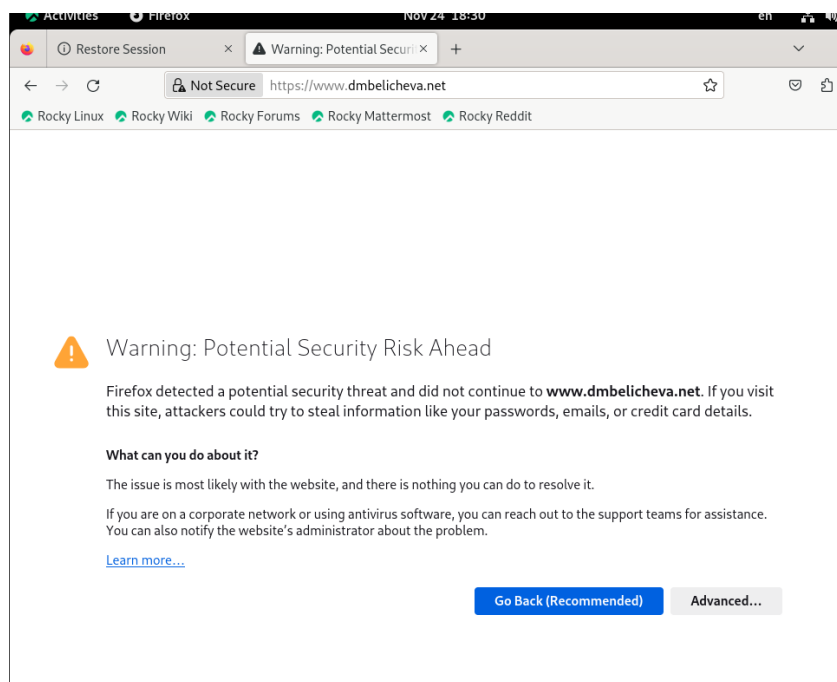


Рис. 3.8: Сообщение о незащищенности на сайте

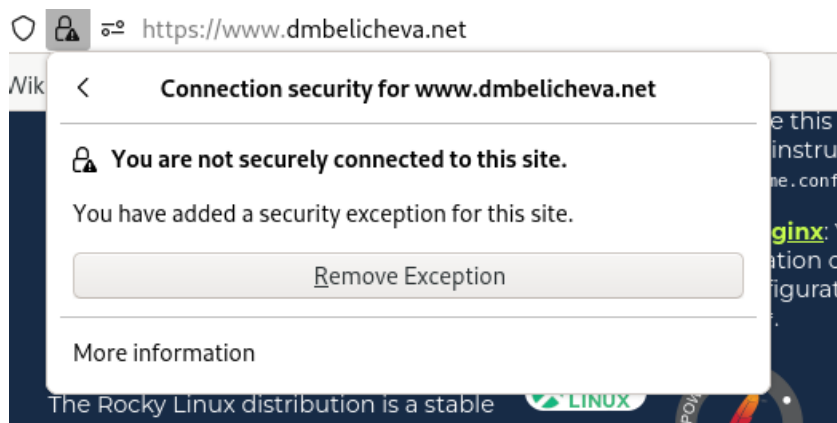


Рис. 3.9: Добавление адреса сервера в исключения

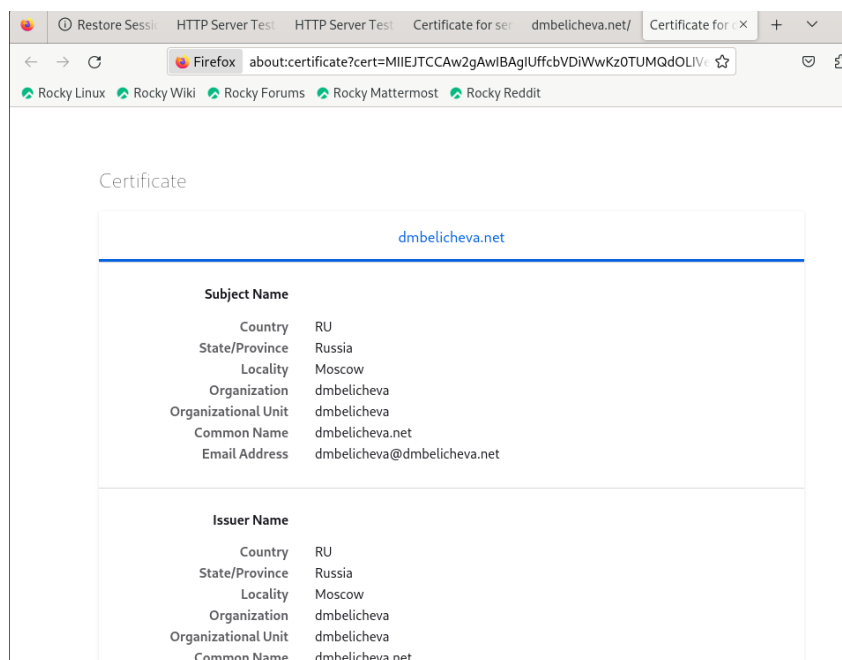


Рис. 3.10: Содержание сертификата

Конфигурирование HTTP-сервера для работы с PHP

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

```
[root@server.dmbelicheva.net conf.d]# dnf -y install php
Last metadata expiration check: 1:28:05 ago on Fri 24 Nov 2023 05:16:06 PM UTC.
Dependencies resolved.
=====
Package                                Architecture      Version           Repository
=====
Installing:
php                                    x86_64            8.0.30-1.el9_2    appstream
Installing dependencies:
nginxfilesystem                       noarch            1:1.20.1-14.el9_2.1 appstream
php-common                             x86_64            8.0.30-1.el9_2    appstream
Installing weak dependencies:
php-cli                               x86_64            8.0.30-1.el9_2    appstream
php-fpm                               x86_64            8.0.30-1.el9_2    appstream
php-mbstring                          x86_64            8.0.30-1.el9_2    appstream
php-opcache                           x86_64            8.0.30-1.el9_2    appstream
php-pdo                               x86_64            8.0.30-1.el9_2    appstream
php-xml                               x86_64            8.0.30-1.el9_2    appstream
=====
Transaction Summary
=====
Install 9 Packages
```

Рис. 3.11: Установка пакетов для работы с php

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

```
<?php
```

```
phpinfo();  
?>
```

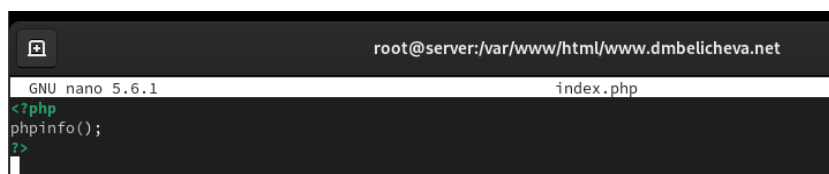


Рис. 3.12: Редактирование файла index.php

Скорректируем права доступа в каталог с веб-контентом: `chown -R apache:apache /var/www`

Восстановим контекст безопасности в SELinux:

```
restorecon -vR /etc
```

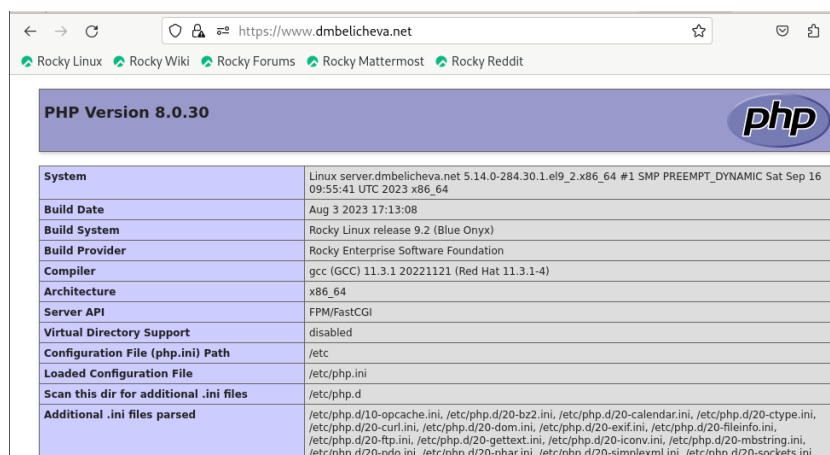
```
restorecon -vR /var/www
```

Перезапустим HTTP-сервер: `systemctl restart httpd`

```
[root@server.dmbelicheva.net www.dmbelicheva.net]# chown -R apache:apache /var/www  
[root@server.dmbelicheva.net www.dmbelicheva.net]# restorecon -vR /etc  
[root@server.dmbelicheva.net www.dmbelicheva.net]# restorecon -vR /var/www  
[root@server.dmbelicheva.net www.dmbelicheva.net]# systemctl restart httpd
```

Рис. 3.13: Права доступа и контекст безопасности в SELinux

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



PHP Version 8.0.30	
System	Linux server.dmbelicheva.net 5.14.0-284.30.1.el9_2.x86_64 #1 SMP PREEMPT_DYNAMIC Sat Sep 16 09:55:41 UTC 2023 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,

Рис. 3.14: Содержание сайта

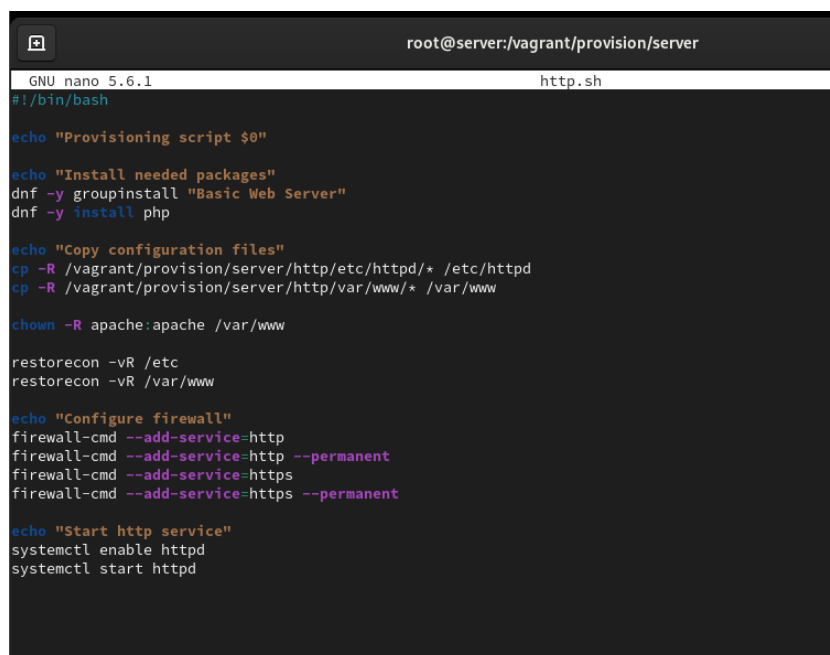
Внесение изменений в настройки внутреннего окружения виртуальной машины

На виртуальной машине server перейдем в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/http` и в соответствующие каталоги скопируйте конфигурационные файлы:

```
[root@server.dmbelicheva.net www.dmbelicheva.net]# cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/etc/httpd/conf.d
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/autoindex.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/fcgid.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/manual.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/README'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/server.dmbelicheva.net.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/ssl.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/userdir.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/welcome.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/www.dmbelicheva.net.conf'? y
[root@server.dmbelicheva.net www.dmbelicheva.net]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
cp: overwrite '/vagrant/provision/server/http/var/www/html/server.dmbelicheva.net/index.html'? y
[root@server.dmbelicheva.net www.dmbelicheva.net]#
bash: y: command not found...
[root@server.dmbelicheva.net www.dmbelicheva.net]# mkdir -p /vagrant/provision/server/http/etc/pki/tls/private
[root@server.dmbelicheva.net www.dmbelicheva.net]# cd /vagrant/provision/server/http
[root@server.dmbelicheva.net http]# mkdir -p /vagrant/provision/server/http/etc/pki/tls/certs
[root@server.dmbelicheva.net http]# cp -R /etc/pki/tls/private/www.dmbelicheva.net.key /vagrant/provision/server/http/etc/pki/tls/private
[root@server.dmbelicheva.net http]# cp -R /etc/pki/tls/certs/www.dmbelicheva.net.crt /vagrant/provision/server/http/etc/pki/tls/certs
[root@server.dmbelicheva.net http]#
```

Рис. 3.15: Внесения изменений в настройки внутреннего окружения

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



```
root@server:/vagrant/provision/server
GNU nano 5.6.1 http.sh
#!/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: Редактирование скрипта

4 Выводы

в процессе выполнения данной лабораторной работы я приобрела практические навыки по расширенному конфигурированию HTTP-сервера Apache в части безопасности и возможности использования PHP.

5 Контрольные вопросы

1. В чём отличие HTTP от HTTPS?

Отличие состоит в том, что HTTPS — расширение протокола HTTP для поддержки шифрования в целях повышения безопасности.

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

Улучшение безопасности при использовании HTTPS вместо HTTP достигается за счёт использования криптографических протоколов при организации HTTP-соединения и передачи по нему данных. Для шифрования может применяться протокол SSL (Secure Sockets Layer) или протокол TLS (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.

Пример: IdenTrust, DigiCert.