

Отчёт о лабораторной работе

Лабораторная работа 5

Мошаров Денис Максимович

Содержание

Цель работы

Приобретение практических навыков по расширенному конфигурированию HTTP-сервера Apache в части безопасности и возможности использования PHP.

Выполнение лабораторной работы

Для начала запустим наш сервер через vagrant (рис. @fig:001).

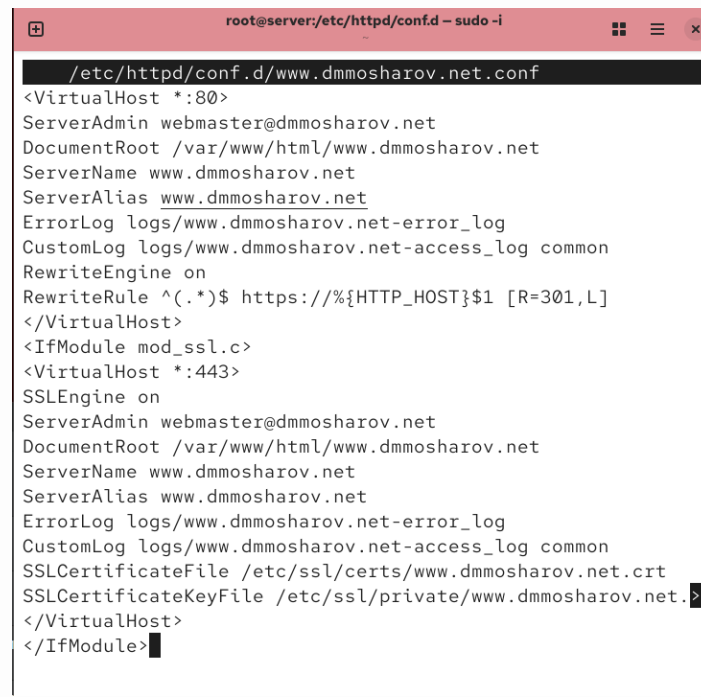
```
C:\Users\denis>cd C:\work_asp\dmmosharov\vagrant

C:\work_asp\dmmosharov\vagrant>vagrant up server
Bringing machine 'server' up with 'virtualbox' provider...
==> server: You assigned a static IP ending in ".1" or ":1" to this machine.
==> server: This is very often used by the router and can cause the
==> server: network to not work properly. If the network doesn't work
==> server: properly, try changing this IP.
==> server: You assigned a static IP ending in ".1" or ":1" to this machine.
==> server: This is very often used by the router and can cause the
==> server: network to not work properly. If the network doesn't work
==> server: properly, try changing this IP.
==> server: Clearing any previously set forwarded ports...
==> server: Clearing any previously set network interfaces...
==> server: Preparing network interfaces based on configuration...
server: Adapter 1: nat
server: Adapter 2: intnet
==> server: Forwarding ports...
server: 22 (guest) => 2222 (host) (adapter 1)
==> server: Running 'pre-boot' VM customizations...
==> server: Booting VM...
==> server: Waiting for machine to boot. This may take a few minutes...
server: SSH address: 127.0.0.1:2222
server: SSH username: vagrant
server: SSH auth method: password
```

Запуск сервера

Далее нам необходимо создать сертификат. Создадим его в созданной нами папке /etc/pki/tls/private, который сделаем симлинком на папку /etc/ssl/private. После создания сертификата, скопируем его в /etc/ssl/certs (рис. [-@fig:002]).

- 10: Завершение блока конфигурации виртуального хоста для порта 80.
- 11: Начало условного блока: конфигурация внутри будет применяться только если загружен модуль mod_ssl.
- 12: Начало блока конфигурации виртуального хоста для обработки запросов по порту 443 (стандартный порт для HTTPS).
- 13: Включение поддержки SSL/TLS шифрования для данного виртуального хоста.
- 14: Указание email-адреса администратора для этого хоста.
- 15: Определение корневой директории с файлами сайта для HTTPS-версии.
- 16: Установка основного доменного имени для HTTPS-хоста.
- 17: Задание альтернативных имен для HTTPS-хоста.
- 18: Указание пути к файлу для логов ошибок HTTPS-хоста.
- 19: Указание пути к файлу для логов доступа HTTPS-хоста.
- 20: Указание пути к файлу публичного SSL-сертификата (.crt).
- 21: Указание пути к файлу приватного (секретного) ключа SSL (.key).
- 22: Завершение блока конфигурации виртуального хоста для порта 443.
- 23: Завершение условного блока IfModule.



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

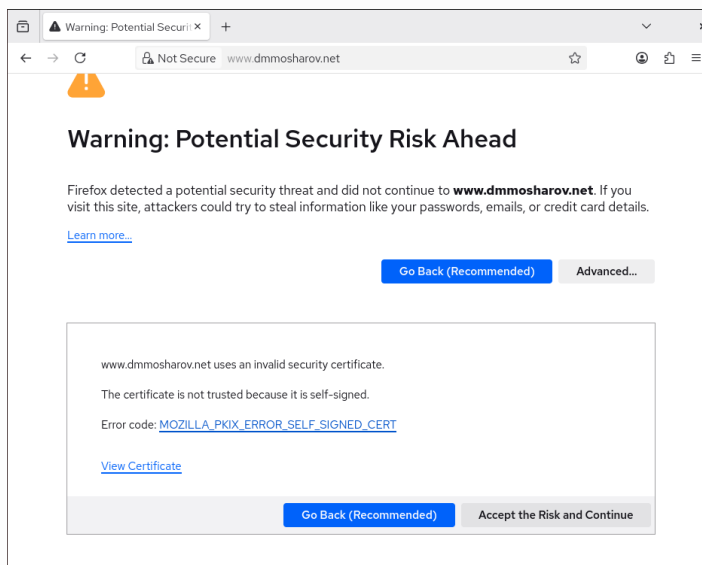
www.dmmosharov.net.conf
```

Теперь с помощью фаервола разрешим работу с https и перезапустим службу httpd (рис. [-@fig:005]).

```
[root@server.dmmosharov.net conf.d]# firewall-cmd --list-services
cockpit dhcp dhcpv6-client dns http ssh
[root@server.dmmosharov.net conf.d]# firewall-cmd --get-services
0-AD RH-Satellite-6 RH-Satellite-6-capsule afp alvr amanda-client amanda-k5-client amqp a
mqps anno-1602 anno-1800 apcupsd aseqnet audit ausweisapp2 bacula bacula-client bareos-di
rector bareos-filedaemon bareos-storage bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoi
n-testnet-rpc bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent civilizat
ion-iv civilization-v cockpit collectd condor-collector cratedb ctddb dds dds-multicast dd
s-unicast dhcp dhcpv6 dhcpv6-client distcc dns dns-over-quic dns-over-tls docker-registry
docker-swarm dropbox-lansync elasticsearch etcd-client etcd-server factorio finger forem
an foreman-proxy freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust f
tp galera ganglia-client ganglia-master git gpsd grafana gre high-availability http http3
https ident imap imaps iperf2 iperf3 ipfs ipp ipp-client ipsec irc ircs iscsi-target isn
s jenkins kadmin kdeconnect kerberos kibana klogin kpasswd kprop kshell kube-api kube-api
server kube-control-plane kube-control-plane-secure kube-controller-manager kube-controll
er-manager-secure kube-nodeport-services kube-scheduler kube-scheduler-secure kube-work
er kubelet kubelet-readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-netwo
rk llmnr llmnr-client llmnr-tcp llmnr-udp managesieve matrix mdns memcache minecraft minidl
na mmdp mongod mosh mountd mpd mqtt mqtt-tls ms-wbt mssql murmur mysql nbd nebula need-f
or-speed-most-wanted netbios-ns netdata-dashboard nfs nfs3 nmea-0183 nrpe ntp nut opentel
emetry openvpn ovirt-imageio ovirt-storageconsole ovirt-vmconsole plex pmcd pmproxy pmweb
api pmwebapis pop3 pop3s postgresql privoxy prometheus prometheus-node-exporter proxy-dhc
p ps2link ps3netstv ptp pulseaudio puppetmaster quassel radius radsec rdp redis redis-sen
tinel rootd rpc-bind rquotad rsh rsyncd rtsp salt-master samba samba-client samba-dc sane
settlers-history-collection sip sipd slimevr slp smtp smtp-submission smtps snmp snmptls
snmptls-trap snmptrap spideroak-lansync spotify-sync squid ssdp ssh statsrv steam-lan-tr
ansfer steam-streaming stellaris stronghold-crusader stun stuns submission supertuxkart s
vdrp svn syncthing syncthing-gui syncthing-relay synergy syscomlan syslog syslog-tls teln
et tentacle terraria tftp tile38 tinc tor-socks transmission-client turn turns upnp-clien
t vdsd vnc-server vrip warpinator wbem-http wbem-https wireguard ws-discovery ws-discover
y-client ws-discovery-host ws-discovery-tcp ws-discovery-udp wsdd wsdd-http wsman wsmans
xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-java-gateway zabbix
x-server zabbix-trapper zabbix-web-service zero-k zerotier
[root@server.dmmosharov.net conf.d]# firewall-cmd --add-service=https
success
[root@server.dmmosharov.net conf.d]# firewall-cmd --add-service=https --permanent
success
[root@server.dmmosharov.net conf.d]# firewall-cmd --reload
success
[root@server.dmmosharov.net conf.d]#
```

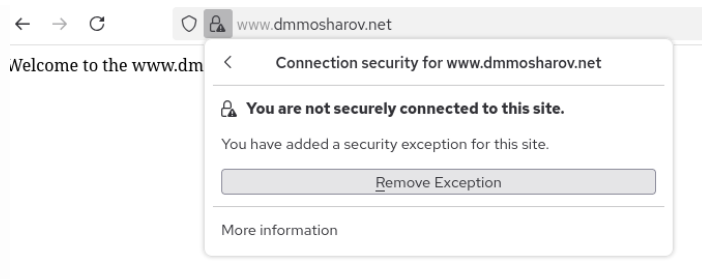
Настройка фаервола

Теперь перейдём на машину клиента. Попробуем войти на наш сайт www.dmmosharov.net и увидим, что браузер предупреждает о том, что соединение незащищено. Тем не менее, подтверждаем переход на сайт (рис. [-@fig:006]).



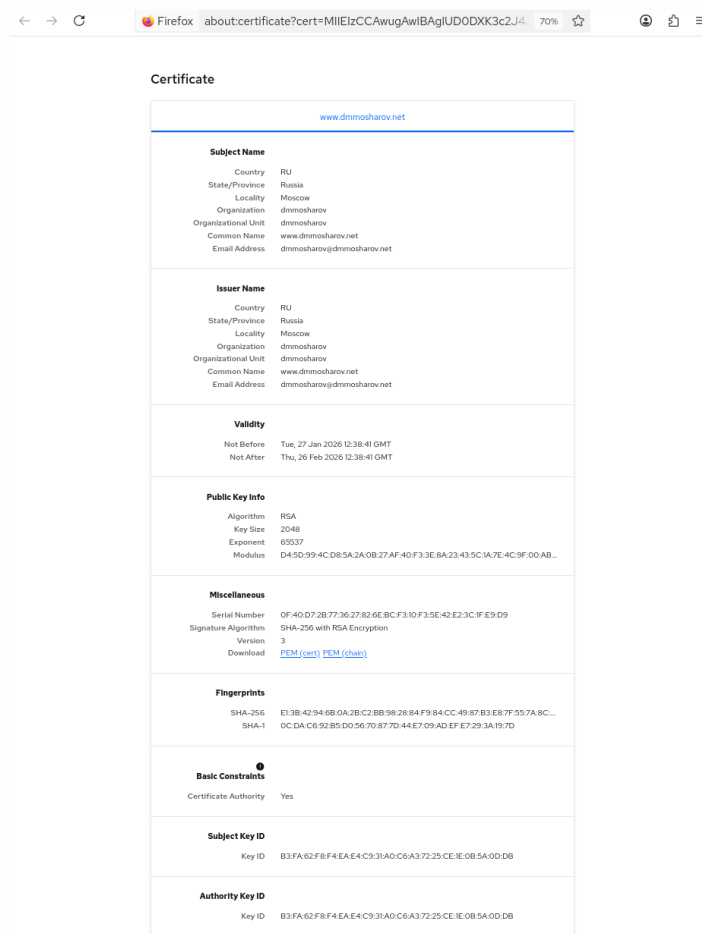
Предупреждение

Как видим, соединение к сайту происходит по https (рис. [-@fig:007]).



Подключение к сайту

Посмотрим на сертификат, и увидим, что там те же данные, что мы вводили при создании (рис. [-@fig:008]).



Сертификат

Установим на сервере пакет php (рис. [-@fig:009]).

```
[root@server.dmmosharov.net conf.d]# dnf -y install php
Extra Packages for Enterprise Linux 10 - x86_64      2.3 kB/s | 12 kB    00:05
Extra Packages for Enterprise Linux 10 - x86_64      1.2 MB/s | 5.6 MB    00:04
Rocky Linux 10 - BaseOS                             288 B/s | 4.3 kB    00:15
Rocky Linux 10 - BaseOS                             18 MB/s | 8.7 MB    00:00
Rocky Linux 10 - AppStream                          573 B/s | 4.3 kB    00:07
Rocky Linux 10 - AppStream                          2.4 MB/s | 2.1 MB    00:00
Rocky Linux 10 - CRB                                14 kB/s | 4.3 kB    00:00
Rocky Linux 10 - CRB                                1.0 MB/s | 498 kB    00:00
Rocky Linux 10 - Extras                             6.5 kB/s | 3.1 kB    00:00
Rocky Linux 10 - Extras                             8.8 kB/s | 5.9 kB    00:00
Dependencies resolved.
=====
Package                Architecture Version      Repository    Size
=====
Installing:
php                    x86_64      8.3.26-1.el10_1 appstream     9.1 k
Installing dependencies:
capstone              x86_64      5.0.1-6.el10 appstream     1.0 M
nginx-filessystem      noarch      2:1.26.3-1.el10 appstream     11 k
php-common            x86_64      8.3.26-1.el10_1 appstream     716 k
Installing weak dependencies:
php-cli               x86_64      8.3.26-1.el10_1 appstream     3.6 M
php-fpm               x86_64      8.3.26-1.el10_1 appstream     1.9 M
php-mbstring          x86_64      8.3.26-1.el10_1 appstream     519 k
php-opcache           x86_64      8.3.26-1.el10_1 appstream     364 k
php-pdo               x86_64      8.3.26-1.el10_1 appstream     86 k
php-xml               x86_64      8.3.26-1.el10_1 appstream     147 k

Transaction Summary
=====
Install 10 Packages
```

Установка php

Создадим на замену старого index.html файл index.php в папке /var/www/html/www.dmmosharov.net (рис. [-@fig:010]).

```
[root@server.dmmosharov.net conf.d]# touch /var/www/html/www.dmmosharov.net/index.php
[root@server.dmmosharov.net conf.d]# nano /var/www/html/www.dmmosharov.net/index.php
[root@server.dmmosharov.net conf.d]# rm /var/www/html/www.dmmosharov.net/index.html
rm: remove regular file '/var/www/html/www.dmmosharov.net/index.html'? y
[root@server.dmmosharov.net conf.d]#
```

Замена файла index.html на index.php

В index.php запишем следующее (рис. [-@fig:011]).

```
GNU nano 8.1 /var/www/html/www.dmmosharov.net/index.php
<?php
phpinfo();
?>
```

Содержимое index.php

Теперь восстановим метки selinux и поменяем владельца нашей папки /var/www на apache, после чего перезапустим службу (рис. [-@fig:012]).

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


Исправление настроек доступа

Теперь посмотрим, как на машине клиента отображается наш сайт (рис. [-@fig:013]).

PHP 8.3.26 - php Xdmmosharov.net Xserver.dmmoshaXHTTP Server TeXProblem loadX+X


www.dmmosharov.net

PHP Version 8.3.26



System	Linux serverdmmosharov.net 6.12.0-55.41.1.el10_0.x86_64 #1 SMP PREEMPT_DYNAMIC Fri Oct 31 14:20:11 UTC 2025 x86_64
Build Date	Sep 23 2025 17:57:26
Build System	Rocky Linux release 10.1 (Red Quartz)
Build Provider	Rocky Enterprise Software Foundation
Compiler	gcc (GCC) 14.3.1 20250617 (Red Hat 14.3.1-2)
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-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-xmlreader.ini
PHP API	20230831
PHP Extension	20230831
Zend Extension	420230831
Zend Extension Build	API420230831,NTS
PHP Extension Build	API20230831,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
Zend Max Execution Timers	disabled
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	compress.zlib, php, file, glob, data, http, ftp, https, ftps, compress.bzip2, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, bzip2.*, convertLiconv.*

This program makes use of the Zend Scripting Language Engine:
Zend Engine v4.3.26, Copyright (c) Zend Technologies with Zend OPcache v8.3.26, Copyright (c), by Zend Technologies



Configuration

bz2

BZip2 Support	Enabled
Stream Wrapper support	compress.bzip2://
Stream Filter support	bzip2.decompress, bzip2.compress

Отображение сайта

Теперь сохраним нашу конфигурацию для vagrant (рис. [-@fig:014]).

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

Сохранение конфигурации

И модифицируем наш файл `/vagrant/provision/server/http.sh` таким образом, что он будет устанавливать пакет `php` и настраивать `firewall` для работы с `https` (рис. [-@fig:015]).

```
GNU nano 8.1 /vagrant/provision/server/http.sh
#!/bin/bash
echo "Provisioning script $0"
echo "Install needed packages"
dnf -y groupinstall "Basic Web Server"
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
echo "Start http service"
systemctl enable httpd
systemctl start httpd

echo "install php"
dnf -y install php
echo "configure firewall for https"
firewall-cmd --list-services
firewall-cmd --get-services
firewall-cmd --add-services=https
firewall-cmd --add-service=https --permanent
firewall-cmd --reload
echo "restart httpd"
systemctl restart httpd
```

`http.sh`

Выводы

В результате выполнения лабораторной работы были получены навыки по конфигурированию HTTP-сервера Apache и https