

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

Расширенная настройка

HTTP-сервера Apache

Дисциплина: Администрирование сетевых подсистем

Студент: Карташова А.С.

Группа: НФИбд-03-18

МОСКВА

2020 г.

Оглавление

Цель работы	2
Задачи	2
Ход работы	3
Конфигурирование HTTP-сервера для работы через протокол	3
HTTPS	3
Конфигурирование HTTP-сервера для работы с PHP	5
Внесение изменений в настройки внутреннего окружения виртуальной машины	7
Вывод	8
Контрольные вопросы	8

Цель работы

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

Задачи

1. Сгенерируем криптографический ключ и само подписанный сертификат безопасности для возможности перехода веб-сервера от работы через протокол HTTP к работе через протокол HTTPS
2. Настроим веб-сервер для работы с PHP
3. Настроим виртуальный хостинг.
4. Напишем скрипт для Vagrant, фиксирующий действия по расширенной настройке HTTP-сервера во внутреннем окружении виртуальной машины server

Ход работы

Конфигурирование HTTP-сервера для работы через протокол HTTPS

Запустим виртуальную машину, перейдем в режим суперпользователя и сформируем ssh-ключ

```
root@server.askartashova.net ~]# mkdir -p /etc/ssl/private
root@server.askartashova.net ~]# cd /etc/ssl/private
root@server.askartashova.net private]# openssl req -x509 -nodes -newkey rsa:2048 -keyout askartashova.net.key
out askartashova.net.crt
Invalid command 'red'; type "help" for a list.
root@server.askartashova.net private]# openssl req -x509 -nodes -newkey rsa:2048 -keyout askartashova.net.key
out askartashova.net.crt
Generating a RSA private key
.....+++++
.....+++++
Writing new private key to 'askartashova.net.key'
-----
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]:askartashova
Organizational Unit Name (eg, section) []:askartashova.net
Common Name (eg, your name or your server's hostname) []:askartashova.net
Email Address []:askartashova@askartashova.net
root@server.askartashova.net private]#
```

Для перехода веб-сервера `www.user.net` на функционирование через протокол HTTPS изменим его конфигурационный файл. Перейдем в каталог с конфигурационными файлами:

```
cd /etc/httpd/conf.d
```

Откроем на редактирование файл `/etc/httpd/conf.d/www.askartashova.net.conf` и заменим его содержимое на следующее:

<VirtualHost *:80> - адрес хоста для прослушивания и номер порта(80), * вместо адреса означает, что веб сервер будет слушать соединения на всех адрес(на localhost и на внешнем)

ServerAdmin webmaster@user.net -email администратора сервера

DocumentRoot /var/www/html/www.user.net

ServerName www.user.net – имя домена

ServerAlias www.user.net – псевдоним домена

ErrorLog logs/www.user.net-error_log – место, куда сохраняются логи ошибок

CustomLog logs/www.user.net-access_log common - место, куда сохраняются логи доступа

Автоматическое подключение сервера к https

RewriteEngine on – включает работу механизма преобразований

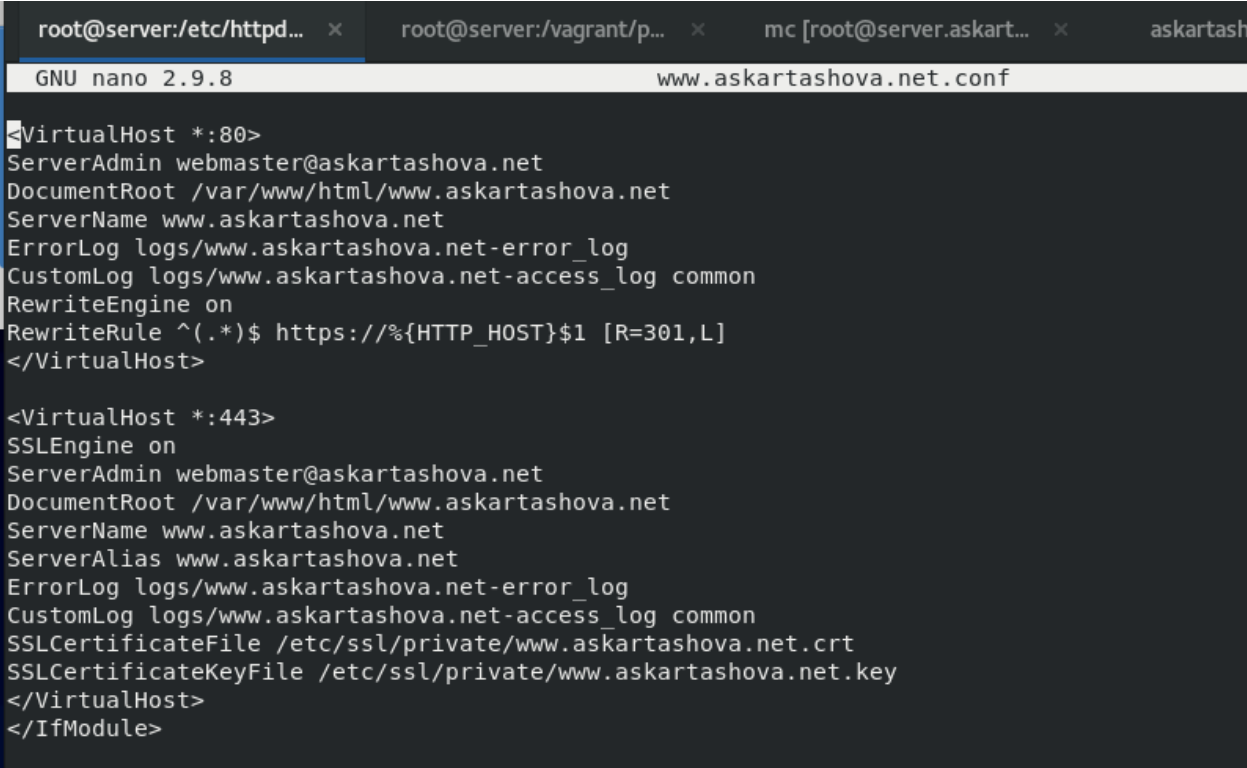
RewriteCond %{SERVER_NAME} = www.user.net – условие преобразования

RewriteRule ^(.*)\$ https://%{HTTP_HOST}%1 [R=301,L] – определяет правила работы механизма преобразований

</VirtualHost>

При использовании безопасного протокола https для работы виртуального хоста, нужно создать хост на порту 443. Весь код секции `<IfModule mod_ssl.c>` будет выполнен лишь в том случае, если активирован модуль `mod_ssl.c`. Это нужно для того, чтобы, если модуль не активирован, код не выживал ошибок:

```
<IfModule mod_ssl.c>
<VirtualHost *:443>
    SSLEngine on – включает поддержку SSL
    ServerAdmin webmaster@user.net
    DocumentRoot /var/www/html/www.user.net
    ServerName www.user.net
    ServerAlias www.user.net
    ErrorLog logs/www.user.net-error_log
    CustomLog logs/www.user.net-access_log common
    SSLCertificateFile /etc/ssl/private/www.user.net.crt - путь к файлам сертификата
    SSLCertificateKeyFile /etc/ssl/private/www.user.net.key - пути к файлам приватного
    ключа
</VirtualHost>
</IfModule>
```



```
root@server:/etc/httpd... x root@server:/vagrant/p... x mc [root@server.askart... x askartash
GNU nano 2.9.8 www.askartashova.net.conf
<VirtualHost *:80>
ServerAdmin webmaster@askartashova.net
DocumentRoot /var/www/html/www.askartashova.net
ServerName www.askartashova.net
ErrorLog logs/www.askartashova.net-error_log
CustomLog logs/www.askartashova.net-access_log common
RewriteEngine on
RewriteRule ^(.*)$ https://%{HTTP_HOST}$1 [R=301,L]
</VirtualHost>

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

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

```
[root@server.askartashova.net conf.d]# firewall-cmd --list-service
cockpit dhcp dhcpv6-client dns http ssh
[root@server.askartashova.net conf.d]# firewall-cmd --get-service
usage: see firewall-cmd man page
firewall-cmd: error: ambiguous option: --get-service could match --get-services, --get-service-helpers
[root@server.askartashova.net conf.d]# firewall-cmd --get-services
RH-Satellite-6 amanda-client amanda-k5-client amqp amqps apcupsd audit bacula bacula-client bb bgp bitcoin bitc
oin-rpc bitcoin-testnet bitcoin-testnet-rpc bittorrent-lsd ceph ceph-mon cfengine cockpit condor-collector ctdb
dhcp dhcpv6 dhcpv6-client distcc dns dns-over-tls docker-registry docker-swarm dropbox-lansync elasticsearch e
tcd-client etcd-server finger freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp gangli
a-client ganglia-master git grafana gre high-availability http https imap imaps ipp ipp-client ipsec irc ircs i
scsi-target isns jenkins kadmin kdeconnect kerberos kibana klogin kpasswd kprop kshell kube-apiserver ldap ldap
s libvirt libvirt-tls lightning-network llmnr managesieve matrix mdns memcache minidlna mongodb mosh mountd mqt
t mqtt-tls ms-wbt mssql murmur mysql nfs nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ovirt-storageconsole
ovirt-vmconsole plex pmcd pmpoxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy prometheus proxy-dhcp ptp p
ulseaudio puppetmaster quassel radius rdp redis redis-sentinel rpc-bind rsh rsyncd rtsp salt-master samba samba
-client samba-dc sane sip sips slp smtp smtp-submission smtps snmp snmptrap spideroak-lansync spotify-sync squi
d sssd ssh steam-streaming svdrp svn syncthing syncthing-gui synergy syslog syslog-tls telnet tentacle tftp tft
p-client tile38 tinc tor-socks transmission-client upnp-client vdsms vnc-server wbem-http wbem-https wsman wsman
s xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-server
[root@server.askartashova.net conf.d]# firewall-cmd --add-service=https
success
[root@server.askartashova.net conf.d]# firewall-cmd --add-service=https -permanent
usage: see firewall-cmd man page
firewall-cmd: error: unrecognized arguments: -permanent
[root@server.askartashova.net conf.d]# firewall-cmd --add-service=https-permanent
Error: INVALID SERVICE: https-permanent
[root@server.askartashova.net conf.d]# firewall-cmd --add-service=https --permanent
success
[root@server.askartashova.net conf.d]# firewall-cmd --reload
success
```

Перезапустим веб-сервер:

systemctl restart httpd

```
[root@server.askartashova.net conf.d]# systemctl restart httpd
Job for httpd.service failed because the control process exited with error code.
See "systemctl status httpd.service" and "journalctl -xe" for details.
[root@server.askartashova.net conf.d]# nano www.askartashova.net.conf
[root@server.askartashova.net conf.d]# nano www.askartashova.net.conf
[root@server.askartashova.net conf.d]# systemctl restart httpd
Job for httpd.service failed because the control process exited with error code.
See "systemctl status httpd.service" and "journalctl -xe" for details.
[root@server.askartashova.net conf.d]# nano www.askartashova.net.conf
[root@server.askartashova.net conf.d]# nano www.askartashova.net.conf
[root@server.askartashova.net conf.d]# systemctl restart httpd
[root@server.askartashova.net conf.d]#
```

При перезапуске веб-сервера у меня возникло 2 ошибки:

- В файле `www.askartashova.net.conf` потеряла строку `<IfModule mod_ssl.c>`
- Неправильный путь к файлу ключа: при формировании файлы называются `user.net.crt` и `user.net.key`, а в `www.askartashova.net.conf` пути к несуществующим файлам
`SSLCertificateFile /etc/ssl/private/www.askartashova.net.crt`
`SSLCertificateKeyFile /etc/ssl/private/www.askartashova.net.key`

На виртуальной машине client в строке браузера введем название веб-сервера `www.askartashova.net` и убедимся, что произойдёт автоматическое переключение на работу по протоколу HTTPS. Опять не работает, как и с http

Конфигурирование HTTP-сервера для работы с PHP

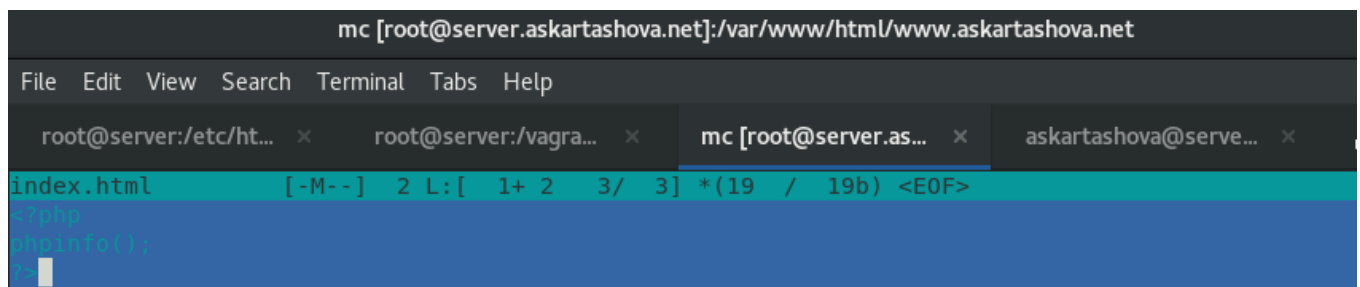
Установим пакеты для работы с PHP:

dnf -y install php

```
[root@server.askartashova.net conf.d]# dnf -y install php
Last metadata expiration check: 0:25:17 ago on Mon 30 Nov 2020 02:31:31 AM UTC.
Dependencies resolved.
=====
Package                Architecture Version                                Repository      Size
=====
Installing:
php                    x86_64      7.2.24-1.module_el8.2.0+313+b04d0a66 AppStream       1.5 M
Installing dependencies:
nginx-filesystem      noarch      1:1.14.1-9.module_el8.0.0+184+e34fea82 AppStream        24 k
php-cli                x86_64      7.2.24-1.module_el8.2.0+313+b04d0a66 AppStream       3.1 M
php-common             x86_64      7.2.24-1.module_el8.2.0+313+b04d0a66 AppStream       661 k
Installing weak dependencies:
php-fpm               x86_64      7.2.24-1.module_el8.2.0+313+b04d0a66 AppStream       1.6 M
Enabling module streams:
nginx                 1.14
php                   7.2
=====
```

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

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



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

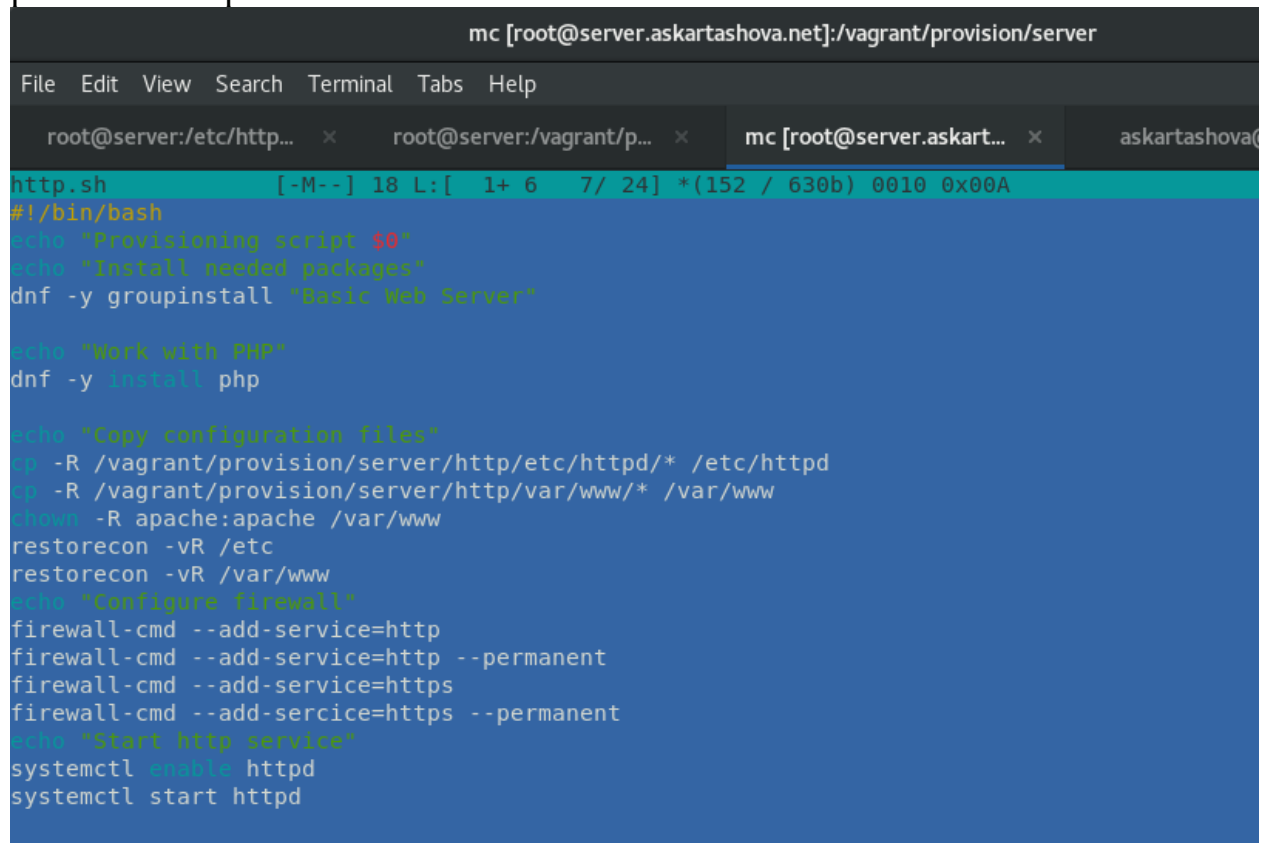
На виртуальной машине client в строке браузера введем название веб-сервера `www.user.net` и убедимся, что будет выведена страница с информацией об используемой на веб-сервере версии PHP.

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.askartashova.net conf.d]# 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'?
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/fcgid.conf'?
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/manual.conf'?
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/README'?
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/server.askartashova.net.conf'?
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/ssl.conf'?
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/userdir.conf'?
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/welcome.conf'?
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/www.askartashova.net.conf'?
[root@server.askartashova.net conf.d]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
cp: overwrite '/vagrant/provision/server/http/var/www/html/server.askartashova.net/index.html'?
cp: overwrite '/vagrant/provision/server/http/var/www/html/www.askartashova.net/index.html'?
[root@server.askartashova.net conf.d]#
[root@server.askartashova.net conf.d]#
[root@server.askartashova.net conf.d]# mkdir -p /vagrant/provision/server/http/etc/ssl/private
[root@server.askartashova.net conf.d]# cp -R /etc/ssl/private/* /vagrant/provision/server/http/etc/ssl/private
```

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



```
mc [root@server.askartashova.net]:/vagrant/provision/server
File Edit View Search Terminal Tabs Help
root@server:/etc/http... x root@server:/vagrant/p... x mc [root@server.askart... x askartashova(
http.sh [-M--] 18 L:[ 1+ 6 7/ 24] *(152 / 630b) 0010 0x00A
#!/bin/bash
echo "Provisioning script $0"
echo "Install needed packages"
dnf -y groupinstall "Basic Web Server"

echo "Work with PHP"
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
```

Вывод

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

Контрольные вопросы

1. В чём отличие HTTP от HTTPS?

HTTPS — это расширение протокола HTTP для поддержки шифрования в целях повышения безопасности.

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

Безопасность контента веб сервера достигается с помощью использования криптографических протоколов. 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.

Центр сертификации на сервере находится в каталоге `/etc/ssl/private/` и состоит из 2х остальных компонентов :

- `askartashova.net.crt` - файл публичного сертификата ЦС. Пользователи, серверы и клиенты используют его для подтверждения для подтверждения подлинности системы и предотвращения атак через посредника.
- `askartashova.net.key` — закрытый ключ, который ЦС использует для подписания сертификатов серверов и клиентов. Если злоумышленник обнаружил ключ, его следует уничтожить

