

# **Отчёт по лабораторной работе №5**

**Дисциплина: Администрирование сетевых подсистем**

Ибрахим Мухсейн Алькамаль

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Выполнение лабораторной работы</b>	<b>6</b>
2.1 Конфигурирование HTTP-сервера для работы через протокол HTTPS	6
2.2 Конфигурирование HTTP-сервера для работы с PHP . . . . .	10
2.3 Внесение изменений в настройки внутреннего окружения виртуальной машины . . . . .	12
<b>3 Выводы</b>	<b>14</b>
<b>4 Ответы на контрольные вопросы</b>	<b>15</b>

# Список иллюстраций

2.1	Генерация самоподписанного SSL-сертификата и закрытого ключа OpenSSL . . . . .	7
2.2	Конфигурация виртуального хоста Apache для HTTPS . . . . .	8
2.3	Настройка firewalld и перезапуск службы httpd . . . . .	9
2.4	Доступ к веб-серверу по HTTPS в браузере клиента . . . . .	9
2.5	Установка пакета PHP с помощью dnf . . . . .	10
2.6	Создание index.php, настройка прав доступа и перезапуск httpd . . . . .	11
2.7	Отображение страницы phpinfo() в браузере клиента . . . . .	11
2.8	Копирование конфигурации HTTP и SSL в каталог provision . . . . .	12
2.9	Изменённый скрипт http.sh с установкой PHP и настройкой HTTPS . . . . .	13

# **Список таблиц**

# **1 Цель работы**

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

## **2 Выполнение лабораторной работы**

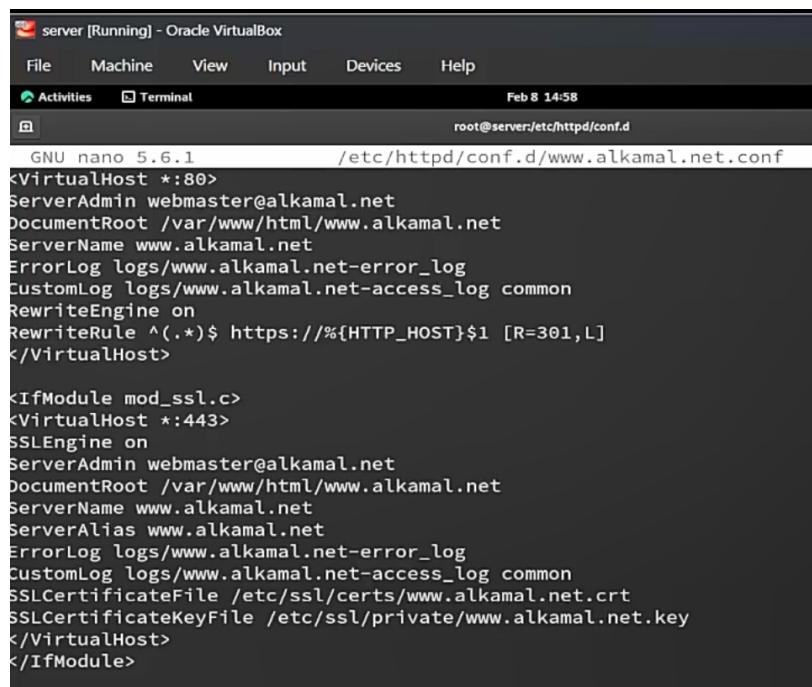
### **2.1 Конфигурирование HTTP-сервера для работы через протокол HTTPS**

В каталоге `/etc/pki/tls/private` был создан закрытый ключ и самоподписанный сертификат с использованием команды `openssl req -x509 -nodes -newkey rsa:2048`. В процессе генерации были заданы параметры DN: код страны — RU, страна — Russia, город — Moscow, организация и подразделение — alkamal, общее имя (CN) — alkamal.net, e-mail — alkamal@alkamal.net. После генерации сертификат был скопирован в каталог `/etc/ssl/certs` (рис. 2.1).

```
[root@server.alkamal.net ~]# mkdir -p /etc/pki/tls/private
[root@server.alkamal.net ~]# ln -s /etc/pki/tls/private /etc/ssl/private
[root@server.alkamal.net ~]# cd /etc/pki/tls/private
[root@server.alkamal.net private]# openssl req -x509 -nodes -newkey rsa:2048 -keyout www.alkamal.net.key -out www.alkamal.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]: Russia
String too long, must be at most 2 bytes long
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]:alkamal
Organizational Unit Name (eg, section) []:alkamal.net
Common Name (eg, your name or your server's hostname) []:www.alkamal.net
Email Address []:alkamal@alkamal.net
[root@server.alkamal.net private]# cp /etc/ssl/private/www.alkamal.net.crt /etc/ssl/certs
[root@server.alkamal.net private]#
```

Рисунок 2.1: Генерация самоподписанного SSL-сертификата и закрытого ключа OpenSSL

Далее был отредактирован конфигурационный файл виртуального хоста `/etc/httpd/conf.d/www.alkamal.net.conf`. В блоке `<VirtualHost * :80>` задана переадресация HTTP-запросов на HTTPS с использованием `RewriteRule`. В блоке `<VirtualHost * :443>` включён модуль SSL (`SSLEngine on`), указаны пути к файлам сертификата (`SSLCertificateFile`) и закрытого ключа (`SSLCertificateKeyFile`), а также параметры журнала и корневой каталог сайта (рис. 2.2).



```
server [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal Feb 8 14:58
root@server:/etc/httpd/conf.d
GNU nano 5.6.1 /etc/httpd/conf.d/www.alkamal.net.conf
<VirtualHost *:80>
ServerAdmin webmaster@alkamal.net
DocumentRoot /var/www/html/www.alkamal.net
ServerName www.alkamal.net
ErrorLog logs/www.alkamal.net-error_log
CustomLog logs/www.alkamal.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@alkamal.net
DocumentRoot /var/www/html/www.alkamal.net
ServerName www.alkamal.net
ServerAlias www.alkamal.net
ErrorLog logs/www.alkamal.net-error_log
CustomLog logs/www.alkamal.net-access_log common
SSLCertificateFile /etc/ssl/certs/www.alkamal.net.crt
SSLCertificateKeyFile /etc/ssl/private/www.alkamal.net.key
</VirtualHost>
</IfModule>
```

Рисунок 2.2: Конфигурация виртуального хоста Apache для HTTPS

После изменения конфигурации были внесены корректировки в настройки межсетевого экрана: выполнена проверка активных и доступных сервисов, добавлен сервис `https`, изменения сохранены как постоянные и применены с помощью перезагрузки правил. Затем выполнен перезапуск службы `httpd` для применения новой конфигурации (рис. 2.3).

```
[root@server.alkamal.net conf.d]# firewall-cmd --list-services
cockpit dhcp dhcpcv6-client dns http ssh
[root@server.alkamal.net conf.d]# firewall-cmd --get-services
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqp amqps apcup
sd audit ausweisapp2 bacula bacula-client bareos-director bareos-filedaemon bareos-storag
e bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin-testnet-rpc bittorrent-lsd ceph ceph
-exporter ceph-mon cfengine checkmk-agent cockpit collectd condor-collector cratedb ctdb
dds dds-multicast dds-unicast dhcp dhcpcv6 dhcpcv6-client distcc dns dns-over-tls docker-
gistry docker-swarm dropbox-lansync elasticsearch etcd-client etcd-server finger foreman
foreman-proxy freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp
galera ganglia-client ganglia-master git gpd grafana gre high-availability http http3 ht
tps ident imap ipfs ipp ipp-client ipsec irc ircs iscsi-target isns jenkins kadmin
kdeconnect kerberos kibana klogin kpasswd kprop kshell kube-api kube-apiserver kube-contr
ol-plane kube-control-plane-secure kube-controller-manager kube-controller-manager-secure
kube-nodeport-services kube-scheduler kube-scheduler-secure kube-worker kubelet kubelet-
readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-network llmnr llmnr-clie
nt llmnr-tcp llmnr-udp manageservices matrix mdns memcache minidlna mongodb mosh mountd mqtt
mqqt-tls ms-wbt msqql murmur mysql nbd nebula netbios-ns netdata-dashboard nfs nfs3 ntmea
-0183 nrpe_ntp_ntp_nut opentelemetry openvpn ovirt-imageio ovirt-storageconsole ovirt-vmconsol
e plex pmcd pmproxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy prometheus promethe
us-node-exporter proxy-dhcp ps2link ps3netsrvr ptp pulseaudio puppetmaster quassel radius
rdp redis redis-sentinel rootd rpc-bind rquotad rsh rsyncd rtsp salt-master samba samba-c
lient samba-dc sane sip sipsl slp smtp smtp-submission smtts snmp snmpnts-snmpnts-trap snm
ptrap spideroak-lansync spotify-sync squid ssdp ssh steam-streaming svdrp svn syncthing s
yncthing-gui syncthing-relay synergy syslog syslog-tls telnet tentacle tftp tile38 tinc t
or-socks transmission-client upnp-client vdsm vnc-server warpinator wbem-http wbem-https
wireguard ws-discovery ws-discovery-client ws-discovery-tcp ws-discovery-udp wsman wsmans
xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-server zerotier
[root@server.alkamal.net conf.d]# firewall-cmd --add-service=https
success
[root@server.alkamal.net conf.d]# firewall-cmd --add-service=https --permanent
success
[root@server.alkamal.net conf.d]# firewall-cmd --reload
success
[root@server.alkamal.net conf.d]# systemctl restart httpd
[root@server.alkamal.net conf.d]#
```

Рисунок 2.3: Настройка firewalld и перезапуск службы httpd

На виртуальной машине client в браузере выполнен переход по адресу <https://www.alkamal.net>. Произошло автоматическое перенаправление с HTTP на HTTPS. После добавления исключения безопасности отображается содержимое веб-страницы сервера, что подтверждает корректную работу HTTPS-соединения (рис. 2.4).

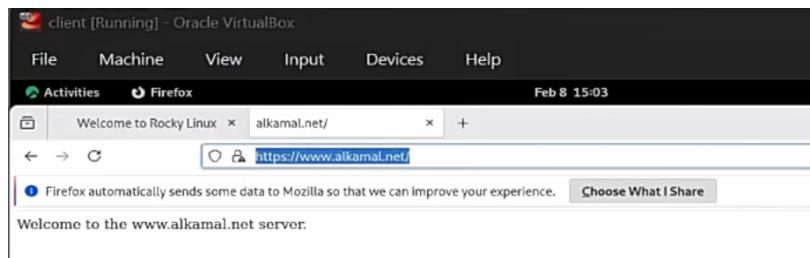
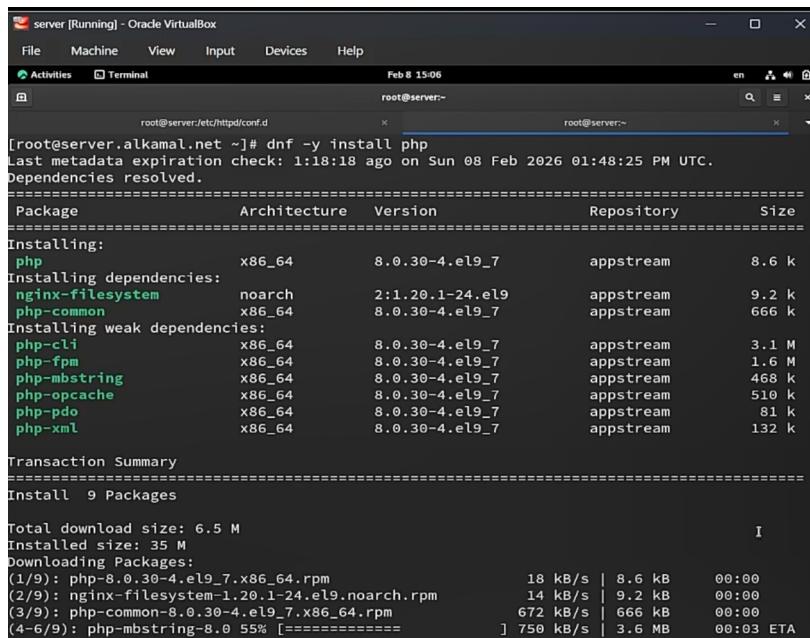


Рисунок 2.4: Доступ к веб-серверу по HTTPS в браузере клиента

## 2.2 Конфигурирование HTTP-сервера для работы с PHP

На сервере выполнена установка пакета php с использованием менеджера пакетов dnf. В результате были установлены основной пакет PHP версии 8.0.30 и связанные зависимости, необходимые для функционирования PHP-интерпретатора в составе веб-сервера (рис. 2.5).



```
[root@server ~]# dnf -y install php
Last metadata expiration check: 1:18:18 ago on Sun 08 Feb 2026 01:48:25 PM UTC.
Dependencies resolved.
=====
 Package           Architecture   Version        Repository      Size
=====
 Installing:
  php              x86_64        8.0.30-4.el9_7    appstream     8.6 k
 Installing dependencies:
  nginx-filesystem          noarch      2:1.20.1-24.el9    appstream     9.2 k
  php-common               x86_64        8.0.30-4.el9_7    appstream    666 k
 Installing weak dependencies:
  php-cli                 x86_64        8.0.30-4.el9_7    appstream     3.1 M
  php-fpm                 x86_64        8.0.30-4.el9_7    appstream     1.6 M
  php-mbstring             x86_64        8.0.30-4.el9_7    appstream     468 k
  php-opcache              x86_64        8.0.30-4.el9_7    appstream     510 k
  php-pdo                  x86_64        8.0.30-4.el9_7    appstream     81 k
  php-xml                 x86_64        8.0.30-4.el9_7    appstream     132 k
 Transaction Summary
=====
 Install 9 Packages

 Total download size: 6.5 M
 Installed size: 35 M
 Downloading Packages:
 (1/9): php-8.0.30-4.el9_7.x86_64.rpm           18 kB/s | 8.6 kB   00:00
 (2/9): nginx-filesystem-1.20.1-24.el9.noarch.rpm 14 kB/s | 9.2 kB   00:00
 (3/9): php-common-8.0.30-4.el9_7.x86_64.rpm      672 kB/s | 666 kB  00:00
 (4-6/9): php-mbstring-8.0.55% [=====]          750 kB/s | 3.6 MB  00:03 ETA
```

Рисунок 2.5: Установка пакета PHP с помощью dnf

В каталоге /var/www/html/www.alkamal.net файл index.html был удалён и создан файл index.php с содержимым `phpinfo()`, предназначенным для вывода конфигурационной информации PHP. После этого изменён владелец каталога /var/www на пользователя и группу apache, восстановлены контексты безопасности SELinux для каталогов /etc и /var/www, затем выполнен перезапуск службы httpd для применения изменений (рис. 2.6).

```
[root@server.alkamal.net ~]# cd /var/www/html/www.alkamal.net
[root@server.alkamal.net www.alkamal.net]# ls
index.html
[root@server.alkamal.net www.alkamal.net]# rm index.html
rm: remove regular file 'index.html'? y
[root@server.alkamal.net www.alkamal.net]# nano index.php
[root@server.alkamal.net www.alkamal.net]# chown -R apache:apache /var/www
[root@server.alkamal.net www.alkamal.net]# restorecon -vR /etc
Relabeled '/etc/sysconfig/network-scripts/ifcfg-eth1' from unconfined_u:object_r:net_conf_t:s0
to unconfined_u:object_r:net_conf_t:s0
[root@server.alkamal.net www.alkamal.net]# restorecon -vR /var/www
[root@server.alkamal.net www.alkamal.net]# systemctl restart httpd
[root@server.alkamal.net www.alkamal.net]#
```

Рисунок 2.6: Создание index.php, настройка прав доступа и перезапуск httpd

На виртуальной машине client в браузере открыт адрес [www.alkamal.net](http://www.alkamal.net). Отображена страница `phpinfo()`, содержащая сведения о версии PHP (8.0.30), параметрах сборки и загруженных модулях, что подтверждает корректную интеграцию PHP с HTTP-сервером (рис. 2.7).

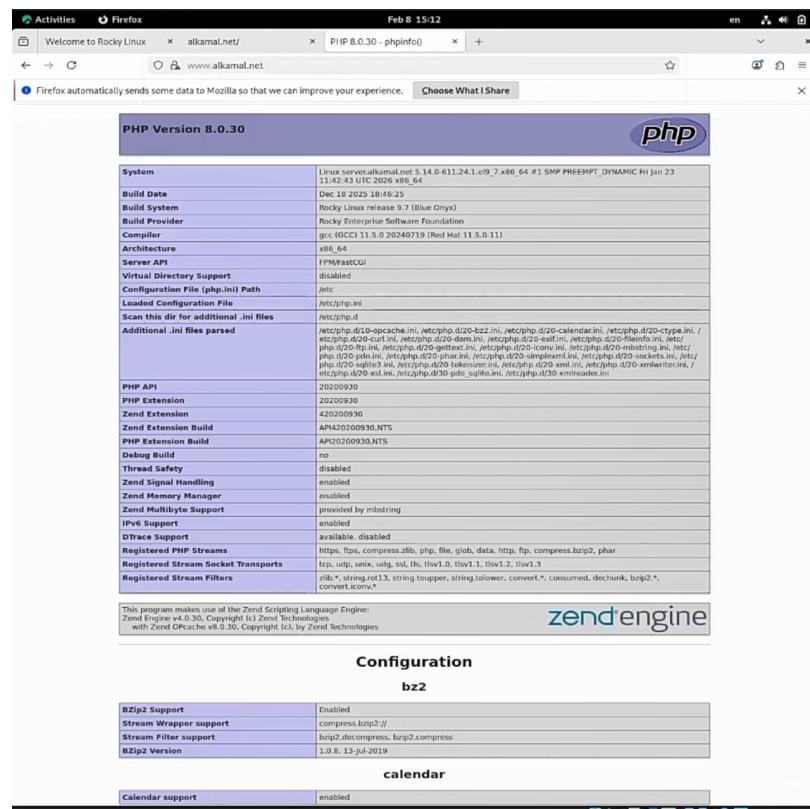


Рисунок 2.7: Отображение страницы `phpinfo()` в браузере клиента

## 2.3 Внесение изменений в настройки внутреннего окружения виртуальной машины

На виртуальной машине server выполнено копирование конфигурационных файлов веб-сервера и веб-контента в каталог внутреннего окружения `/vagrant/provision/server`. Скопированы файлы из `/etc/httpd/conf.d` и `/var/www/html`, созданы каталоги для хранения ключей и сертификатов `/etc/pki/tls/private` и `/etc/pki/tls/certs` внутри структуры provision, после чего в них помещены файлы `www.alkamal.net.key` и `www.alkamal.net.crt` (рис. 2.8).

```
[alkamal@server.alkamal.net ~]$ sudo -i
[sudo] password for alkamal:
[root@server.alkamal.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/rocky-snipolicy.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/server.alkamal.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.alkamal.net.conf'? y
[root@server.alkamal.net ~]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www
/html
cp: overwrite '/vagrant/provision/server/http/var/www/html/server.alkamal.net/index.html'? y
[root@server.alkamal.net ~]# mkdir -p /vagrant/provision/server/http/etc/pki/tls/private
[root@server.alkamal.net ~]# mkdir -p /vagrant/provision/server/http/etc/pki/tls/certs
[root@server.alkamal.net ~]# cp -R /etc/pki/tls/private/www.alkamal.net.key /vagrant/provis
ion/server/http/etc/pki/tls/private
[root@server.alkamal.net ~]# cp -R /etc/pki/tls/certs/www.alkamal.net.crt /vagrant/provis
ion/server/http/etc/pki/tls/certs
[root@server.alkamal.net ~]#
```

Рисунок 2.8: Копирование конфигурации HTTP и SSL в каталог provision

Далее в скрипт `/vagrant/provision/server/http.sh` внесены изменения: добавлена установка пакета `php`, реализовано копирование конфигурационных файлов в системные каталоги `/etc/httpd` и `/var/www`, выполнена корректировка владельца каталога `/var/www`, восстановление контекстов SELinux и добавлены правила межсетевого экрана для сервисов `http` и `https` с постоянным сохранением. В завершение настроено включение и запуск службы `httpd` (рис. 2.9).

```
GNU nano 5.6.1          /vagrant/provision/server/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
```

Рисунок 2.9: Изменённый скрипт http.sh с установкой PHP и настройкой HTTPS

## **3 Выводы**

В ходе выполнения работы была реализована настройка HTTP-сервера Apache для функционирования по протоколу HTTPS с использованием самоподписанного SSL-сертификата. Выполнена генерация закрытого ключа и сертификата, их интеграция в конфигурацию виртуального хоста и настройка автоматической переадресации HTTP-запросов на HTTPS.

Дополнительно выполнена установка и интеграция интерпретатора PHP с веб-сервером. Создание файла `index.php` с вызовом `phpinfo()` подтвердило корректность обработки PHP-скриптов сервером.

Внесённые изменения были перенесены во внутреннее окружение виртуальной машины посредством копирования конфигурационных файлов и модификации provisioning-скрипта `http.sh`, включающего установку PHP, настройку SELinux и правил межсетевого экрана для сервисов `http` и `https`.

Работоспособность веб-сервера подтверждена успешным доступом к ресурсу по протоколу HTTPS и отображением страницы конфигурации PHP в браузере клиента.

## **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. Владельцы веб-сайтов могут получить сертификат от Let's Encrypt, чтобы обеспечить шифрование и подтвердить свою легитимность в онлайн-среде.