

Организовать API с 1С

Apache

Качаем [Apache](#), копируем на диск C

httpd.conf

```
# === Основные настройки ===
Define SRVROOT "C:/Apache24"
ServerRoot ${SRVROOT}
Listen 4453
ServerAdmin admin@example.org
ServerName terminal.example.org

# === Размер тела запроса (100MB) ===
LimitRequestBody 104857600

# === Поддержка Keep-Alive для долгих запросов 1С ===
KeepAlive On
MaxKeepAliveRequests 1000
KeepAliveTimeout 300
Timeout 600

# === Настройки ядра ===
AcceptFilter https none
AcceptFilter http none
EnableSendfile Off
EnableMMAP Off

# === Загрузка необходимых модулей ===
LoadModule authn_core_module modules/mod_authn_core.so
LoadModule authn_file_module modules/mod_authn_file.so
```

```
LoadModule authz_core_module modules/mod_authz_core.so
LoadModule authz_host_module modules/mod_authz_host.so
LoadModule dir_module modules/mod_dir.so
LoadModule env_module modules/mod_env.so
LoadModule include_module modules/mod_include.so
LoadModule isapi_module modules/mod_isapi.so
LoadModule log_config_module modules/mod_log_config.so
LoadModule mime_module modules/mod_mime.so
LoadModule negotiation_module modules/mod_negotiation.so
LoadModule setenvif_module modules/mod_setenvif.so
LoadModule socache_shmcb_module modules/mod_socache_shmcb.so
LoadModule ssl_module modules/mod_ssl.so
LoadModule _1cws_module "C:/Program Files/1cv8/8.3.27.1859/bin/wsap24.dll"
LoadModule headers_module modules/mod_headers.so
LoadModule alias_module modules/mod_alias.so
```

```
# === Корневая директория ===
DocumentRoot "${SRVROOT}/htdocs"
<Directory "${SRVROOT}/htdocs">
    Options -Indexes +FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```

```
# === Запрет доступа к системным файлам ===
<Files ".ht*">
    Require all denied
</Files>
<DirectoryMatch "^.*\..+">
    Require all denied
</DirectoryMatch>
```

```
# === Логи ===
ErrorLog "logs/error.log"
LogLevel warn
CustomLog "logs/access.log" common
```

```
# === SSL ===
<IfModule ssl_module>
```

```
SSLEngine On
SSLCertificateFile conf/ssl/cert.pem
SSLCertificateKeyFile conf/ssl/key.pem
SSLCACertificateFile conf/ssl/ca.crt

SSLProtocol -all +TLSv1.2
SSLCipherSuite HIGH:!aNULL:!MD5
SSLHonorCipherOrder On
#SSLOpenSSLConfCmd CertificateVerification require

SSLSessionCache "shmcb:${SRVROOT}/logs/ssl_scache(2048000)"
SSLSessionCacheTimeout 300

AddType application/x-x509-ca-cert .crt
AddType application/x-pkcs7-crl .crl

CustomLog "${SRVROOT}/logs/ssl_request.log" \
    "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b" env=HTTPS
</IfModule>

# ==== Публикация базы 1C ====
Alias "/post4egor" "bases/post4egor"
<Directory "bases/post4egor">
    AllowOverride None
    Options None
    Require all granted

    SetHandler 1c-application
    ManagedApplicationDescriptor "bases/post4egor/default.vrd"

    #Header set Cache-Control "no-store, no-cache, must-revalidate"
    #Header set Pragma "no-cache"

    SSLVerifyClient require
    SSLRequire %{SSL_CLIENT_S_DN_CN} eq "Apache1CPub"

</Directory>
```

```
# === Индексный файл ===  
<IfModule dir_module>  
    DirectoryIndex index.html  
</IfModule>
```

default.vrd

```
<?xml version="1.0" encoding="UTF-8"?>  
<point xmlns="http://v8.1c.ru/8.2/virtual-resource-system"  
  xmlns:xs="http://www.w3.org/2001/XMLSchema"  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  base="/post4egor"  
  ib="Srvr=&quot;onec&quot;;Ref=&quot;post4egor&quot;;usr=&quot;1CAPIUser&quot;;pwd=&quot;keef9yooy  
3Ze1itiexie&quot;;">  
  <httpServices publishByDefault="false">  
    <service name="Api"  
      rootUrl="api"  
      enable="true"  
      reuseSessions="autouse"  
      sessionMaxAge="60"  
      poolSize="20"  
      poolTimeout="10">  
    </service>  
  </httpServices>  
</point>
```

Сертификат для веб-сервера

```
openssl req -x509 -nodes -days 3650 -newkey rsa:3072 \  
-keyout key.pem \  
-out cert.pem \  
-subj "/C=RU/ST=Moscow/L=Moscow/O=Company/OU=IT/CN=localhost"
```

CA

```
openssl req -new -newkey rsa:3072 -nodes -keyout ca.key -x509 -days 3650 \  
-subj /C=RU/ST=Msk/L=Msk/O=MyOrg/OU=MyUnit/CN=IT/emailAddress=usr@localhost \  

```

```
-out ca.crt
```

О том, как выпустить клиентские сертификаты в Apache написано тут:

https://www.opennet.ru/base/sec/ssl_cert.txt.html

Проверка

POST

```
curl -X POST \  
  --cert-type P12 \  
  --cert ./Apache1CPub.p12 \  
  --pass "" \  
  -H "Content-Type: application/json" \  
  -d '{"key": "value"}' \  
  -k \  
  "https://apachehost:4453/post4egor/hs/api/v1/TakeJSON"
```

GET

```
curl -X GET \  
  --cert-type P12 \  
  --cert ./Apache1CPub.p12 \  
  --pass "" \  
  -k \  
  "https://apachehost:4453/post4egor/hs/api/test/"
```

Revision #8

Created 22 January 2026 02:54:58 by Admin

Updated 23 January 2026 05:01:34 by Admin