Отчет. Задание № 2. Работа с веб-сервисами.

Левченко Антон, 43506/3

1. **Установить в качестве web-серверов Apache, Nginx и IIS.**
2. **Для каждого из них построить конфигурационные файлы, которые**
   1. **устанавливают в качестве основной кодировки файлов UTF-8**
   2. **определяет единую точку входа в приложение**
   3. **запрещает доступ к определенной директории (на свое усмотрение) при любых запросах**
   4. **организуют доступ к admin дирректории по средствам base-аутентификации**
   5. **устанавливают правила для кросс доменных запросов**
   6. **настроить кэширование на уровне сервера**

**Репозиторий:** <https://github.com/antonylevchenko/ui_task_2_web_services>

**Apache**

В репозитории конфигурационный файл apache – apache/httpd.conf. Должен быть помещён по адресу apache/conf/httpd.conf при установленном apache. Например, C:\xampp\apache\conf\httpd.conf.

* **Установка в качестве основной кодировки файлов UTF-8**

# Default charset

AddDefaultCharset UTF-8

* **Определение единой точки входа в приложение**

DocumentRoot "C:\Users\anton.levchenko\Projects\University\ui\UI\_task\_2\_web\_services\app"

Header set Access-Control-Allow-Origin "\*"

<Directory "C:\Users\anton.levchenko\Projects\University\ui\UI\_task\_2\_web\_services\app">

#

# Possible values for the Options directive are "None", "All",

# or any combination of:

# Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews

#

# Note that "MultiViews" must be named \*explicitly\* --- "Options All"

# doesn't give it to you.

#

# The Options directive is both complicated and important. Please see

# http://httpd.apache.org/docs/2.4/mod/core.html#options

# for more information.

#

Options Indexes FollowSymLinks Includes ExecCGI

#

# AllowOverride controls what directives may be placed in .htaccess files.

# It can be "All", "None", or any combination of the keywords:

# AllowOverride FileInfo AuthConfig Limit

#

AllowOverride All

#

# Controls who can get stuff from this server.

#

Require all granted

# To enable entry point

<IfModule mod\_rewrite.c>

RewriteEngine on

RewriteCond %{REQUEST\_FILENAME} /.\*

RewriteRule ^ entry.php [PT]

RewriteCond %{REQUEST\_FILENAME} !-f

RewriteCond %{REQUEST\_FILENAME} !-d

RewriteRule ^ entry.php [PT,L]

</IfModule>

</Directory>

* **Запрет доступа к определённой директории**

# Forbidden directory

<Directory "C:\Users\anton.levchenko\Projects\University\ui\UI\_task\_2\_web\_services\app\forbidden">

Deny from all

</Directory>

* **Организация доступа к admin директории посредством base-аутентификации**

# Admin directory, which requires authentication (login: admin, password: admin)

<Directory "C:\Users\anton.levchenko\Projects\University\ui\UI\_task\_2\_web\_services\app\admin">

AuthType Basic

AuthName "Authentication Required"

AuthUserFile "C:\Users\anton.levchenko\Projects\University\ui\UI\_task\_2\_web\_services\app\forbidden\.htpasswd"

Require valid-user

</Directory>

* **Установка правила для кросс-доменных запросов**

Header set Access-Control-Allow-Origin "\*"

* **Кэширование на уровне сервера**

LoadModule cache\_disk\_module modules/mod\_cache\_disk.so

**+**

# Cache

<IfModule mod\_cache.c>

<IfModule mod\_cache\_disk.c>

CacheRoot "C:/xampp/apache/cache"

CacheDirLevels 5

CacheDirLength 3

</IfModule>

</IfModule>

Результаты работы тестовых запросов, отправленных с помощью Postman приведены в директории apache/screenshots.

**Nginx**

В репозитории конфигурационный файл nginx – nginx/nginx.conf. Должен быть помещён по адресу conf/nginx.conf при установленном nginx. Например, C:\nginx-1.14.2\conf\nginx.conf

* **Установка в качестве основной кодировки файлов UTF-8**

#Charset

charset utf-8;

* **Определение единой точки входа в приложение**

#entry pont

rewrite /.\* /entry.php;

* **Запрет доступа к определённой директории**

#Forbidden directory

location /forbidden {

deny all;

}

* **Организация доступа к admin директории посредством base-аутентификации**

#Authorized access admin directory

location /admin {

auth\_basic "Authentification required.";

auth\_basic\_user\_file C:\Users\anton.levchenko\Projects\University\ui\UI\_task\_2\_web\_services\app\forbidden\.htpasswd;

}

* **Установка правила для кросс-доменных запросов**

# Cross-domain requests

if ($request\_method = 'OPTIONS') {

add\_header 'Access-Control-Allow-Origin' '\*';

add\_header 'Access-Control-Allow-Methods' 'GET, POST, OPTIONS';

add\_header 'Access-Control-Allow-Headers' 'DNT,User-Agent,X-Requested-With,If-Modified-Since,Cache-Control,Content-Type,Range';

add\_header 'Access-Control-Max-Age' 1728000;

add\_header 'Content-Type' 'text/plain; charset=utf-8';

add\_header 'Content-Length' 0;

return 204;

}

if ($request\_method = 'POST') {

add\_header 'Access-Control-Allow-Origin' '\*';

add\_header 'Access-Control-Allow-Methods' 'GET, POST, OPTIONS';

add\_header 'Access-Control-Allow-Headers' 'DNT,User-Agent,X-Requested-With,If-Modified-Since,Cache-Control,Content-Type,Range';

add\_header 'Access-Control-Expose-Headers' 'Content-Length,Content-Range';

}

if ($request\_method = 'GET') {

add\_header 'Access-Control-Allow-Origin' '\*';

add\_header 'Access-Control-Allow-Methods' 'GET, POST, OPTIONS';

add\_header 'Access-Control-Allow-Headers' 'DNT,User-Agent,X-Requested-With,If-Modified-Since,Cache-Control,Content-Type,Range';

add\_header 'Access-Control-Expose-Headers' 'Content-Length,Content-Range';

}

* **Кэширование на уровне сервера**

#Cache

proxy\_cache\_path C:/nginx-1.14.2/cache keys\_zone=one:10m;

Для корректной работы php-файлов потребовалось добавить конструкцию вида:

location ~ \.php$ {

fastcgi\_pass 127.0.0.1:9123;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

include fastcgi\_params;

}

И обеспечить работу php-cgi.

Результаты работы тестовых запросов, отправленных с помощью Postman приведены в директории nginx/screenshots.

**IIS**

В репозитории конфигурационные файлы iis iis/global/web.config и iis/forbidden/web.config. global/web.config должен быть помещён в корневую папку приложения. forbidden/web.config – в папку forbidden.

* **Установка в качестве основной кодировки файлов UTF-8**

<?xml version="1.0" encoding="UTF-8"?>

* **Определение единой точки входа в приложение**

<rule name="Rule">

<match url="^/.\*\.php$" />

<action type="Rewrite" url="entry.php" appendQueryString="true" />

<conditions trackAllCaptures="true">

</conditions>

</rule>

* **Запрет доступа к определённой директории**

<security>

<requestFiltering>

<hiddenSegments>

<add segment="forbidden" />

</hiddenSegments>

</requestFiltering>

</security>

* **Установка правила для кросс-доменных запросов**

<add name="Access-Control-Allow-Origin" value="\*" />

* **Кэширование на уровне сервера**

<clientCache cacheControlMode="NoControl" />

1. **Реализовать схему, при которой Nginx принимает все запросы и в случае получения запроса к статическим данных отдает их, а в случае обращения к скрипту, переадресует запрос к Apache (Nginx как прослойка между клиентом и Apache сервером).**

**Nginx + apache**

В репозитории конфигурационный файл apache – nginx-with-apache/httpd.conf. Должен быть помещён по адресу apache/conf/httpd.conf при установленном apache. Например, C:\xampp\apache\conf\httpd.conf.

Конфигурационный файл nginx – nginx-with-apache/nginx.conf. Должен быть помещён по адресу conf/nginx.conf при установленном nginx. Например, C:\nginx-1.14.2\conf\nginx.conf

Изменения в nginx.conf:

server {

listen 127.0.0.1:80;

server\_name localhost;

autoindex on;

#set default charset

charset utf-8;

# Передаем динамику серверу apache, статику оставляем серверу nginx.

location ~ \.(jpg|jpeg|gif|png|ico|css|zip|tgz|gz|rar|pdf|txt|tar|html)$ {

#WebSite root

root C:\Users\anton.levchenko\Projects\University\ui\UI\_task\_2\_web\_services\app;

}

# Передача запроса серверу apache.

location / {

# Адрес сервера apache

proxy\_pass http://127.0.0.1:8080/;

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $remote\_addr;

proxy\_connect\_timeout 120;

proxy\_send\_timeout 120;

proxy\_read\_timeout 180;

client\_max\_body\_size 10m;

client\_body\_buffer\_size 128k;

}

#Authorized access directory

location ^~ /admin {

auth\_basic "Authentification required.";

auth\_basic\_user\_file C:\Users\anton.levchenko\Projects\University\ui\UI\_task\_2\_web\_services\app\forbidden\.htpasswd;

}

# Запрещаем обращение к .ht

location ~ /\.ht {

deny all;

}

}

Изменения в httpd.conf:

Listen 8080