**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра Вычислительной техники**

**ОТЧЕТ**

**по лабораторной работе № 6**

**по дисциплине «Web программирование» Тема: «Аутентификация и авторизация пользователей Web-приложения»**

|  |
| --- |
| Студент гр. 3311 Баймухамедов Р. Р. \_\_\_\_\_\_ |
|  |
| Преподаватель Калмыков М.А. \_\_\_\_\_\_ |

Санкт-Петербург

2025

**Цель работы**

Знакомство со способами реализации аутентификации и авторизации пользователей Web-приложения

**Выполнение лабораторной работы**

Изменили следующие файлы

*# HTTP — пока без редиректа (включим в конце, когда https взлетит)*

server {

  listen 80*;*

  server\_name \_*;*

  root  /usr/share/nginx/html*;*

  index index.html*;*

  charset utf-8*;*

*# i18n*

  location ^~ /i18n/ {

    try\_files $uri =404*;*

    default\_type application/json*;*

    add\_header Cache-Control "no-store, no-cache, must-revalidate" always*;*

    expires -1*;*

  }

*# API (без auth на http)*

  location /api/ {

    proxy\_pass http://app:3000*;*

    proxy\_set\_header Host $host*;*

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

    proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for*;*

    proxy\_set\_header X-Forwarded-Proto $scheme*;*

  }

*# SPA*

  location / {

    try\_files $uri $uri/ /index.html*;*

  }

}

*# HTTPS*

server {

  listen 443 ssl*;      # без 'http2' в listen — убираем warning*

  http2 on*;            # так правильно включать http2 в новых версиях*

  server\_name \_*;*

  ssl\_certificate     /etc/nginx/certs/server.crt*;*

  ssl\_certificate\_key /etc/nginx/certs/server.key*;*

  ssl\_protocols TLSv1.2 TLSv1.3*;*

  root  /usr/share/nginx/html*;*

  index index.html*;*

  charset utf-8*;*

*# i18n*

  location ^~ /i18n/ {

    try\_files $uri =404*;*

    default\_type application/json*;*

    add\_header Cache-Control "no-store, no-cache, must-revalidate" always*;*

    expires -1*;*

  }

*# API под Basic-Auth*

  location /api/ {

    auth\_basic "Movies API (lab6)"*;*

    auth\_basic\_user\_file /etc/nginx/conf.d/.htpasswd*;*

    proxy\_pass http://app:3000*;*

    proxy\_set\_header Host $host*;*

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

    proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for*;*

    proxy\_set\_header X-Forwarded-Proto $scheme*;*

  }

*# SPA*

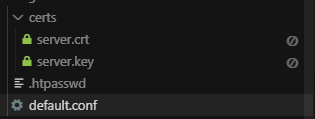
  location / {

    try\_files $uri $uri/ /index.html*;*

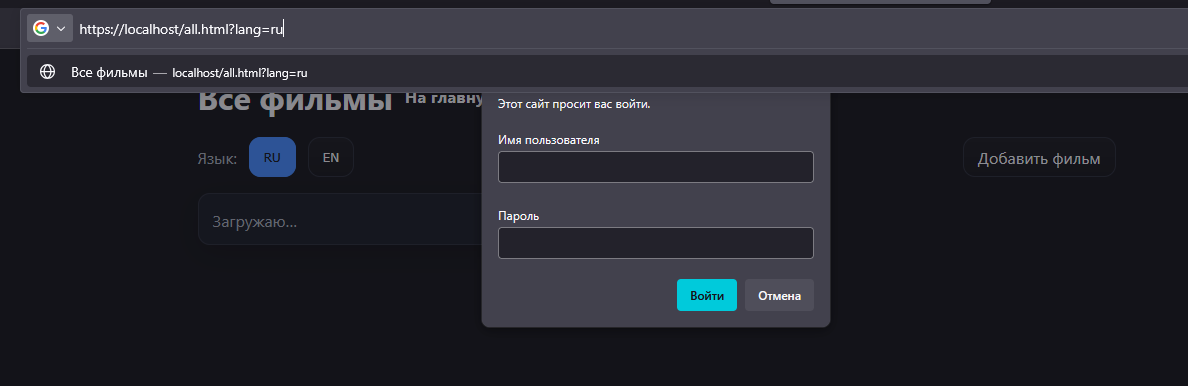
  }

}

Сгенерировали server.crt и server.key, а также .htpasswd



**Пример работоспособности программы**



**Заключение**

В данной лабораторной работе мы познакомились со способами реализации аутентификации и авторизации пользователей Web-приложения