Міністерство освіти і науки України

Національний технічний університет України

„КПІ імені Ігоря Сікорського ”

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

ЗВІТ

лабораторна робота №2 з

курсу «Безпека програмного забезпечення»

Тема: «Засвоювання базових навичок OAuth2 авторизаційного протокола»

Перевірив: Виконав:

Група ІП-11 Головня Олександр

Київ 2024

Завдання:

1) Використовуючи наведені налаштування, створити запит на отримання токену

через client_credential grant

https://auth0.com/docs/api/authentication#client-credentials-flow

curl --request POST \

--url 'https://YOUR_DOMAIN/oauth/token' \

--header 'content-type: application/x-www-form-urlencoded' \

--data

'audience=API_IDENTIFIER&grant_type=client_credentials&client_id=YOUR_CLIENT_ID&client_secret=YOUR_CLIENT_SECRET'

Domain: kpi.eu.auth0.com

ClientID: JIvCO5c2IBHlAe2patn6l6q5H35qxti0

Client Secret: ZRF8Op0tWM36p1_hxXTU-B0K_Gq_-eAVtlrQpY24CasYiDmcXBhNS6IJMNcz1EgB

Audience: https://kpi.eu.auth0.com/api/v2

2) Створити юзера з власним email в системі використовуючи метод

https://auth0.com/docs/api/management/v2#!/Users/post_users

та отриманий токен

Для отримання додаткового балу – зробити власний акаунт в auth0 https://auth0.com/
Створити application та запити описані вище вже використовуючи власні налаштування. Детальну інформацію розбирали на практичному завданні.

Результати виконання:

Повний лістинг файлів можна знайти тут:

https://github.com/YeaLowww/KPI-ALL/tree/main/Fourth%20year/Software%20security

Підставивши усі параметри, отримаємо відповідний запит:

curl --request POST --url "https://kpi.eu.auth0.com/oauth/token" --header "content-type:application/x-www-form-urlencoded" --data "audience=https://kpi.eu.auth0.com/api/v2/&grant_type=client_credentials&client_id=JIvCO5c2IBHlAe2patn6l6q5H35qxti0&client_secret=ZRF8Op0tWM36p1_hxXTU-B0K_Gq_-eAVtlrQpY24CasYiDmcXBhNS6IJMNcz1EgB"

curl --request POST --url 'https://kpi.eu.auth0.com/oauth/token' --header 'content-type:application/x-www-form-urlencoded' --data 'audience=https://kpi.eu.auth0.com/api/v2/&grant_type=client_credentials&client_id=JIvCO5c2IBHlAe2patn6l6q5H35qxti0&client_secret=ZRF8Op0tWM36p1_hxXTU-B0K_Gq_-eAVtlrQpY24CasYiDmcXBhNS6IJMNcz1EgB'

Результат запиту можна побачити на рисунку:

```
C:\Users\Саша Головня\Desktop\security\lab2>curl --request POST --url "https://kpi.eu.auth0.com/oauth/token" --header "content-type:application/x-www-form-urlencoded" --data "audience=https://kpi.eu.auth0.com/api/v2/&grant_type=client_credentials&client_id=JIvCO5c2IBHlAe2patn6l6q5H35qxti0&client_secret=ZRF8Op0tWM36p1_hxXTU-B0K_Gq_-eAVtlrQpY24CasYiDmcXBhNS6IJMNcz1EgB"
{"access_token":"eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6IjVCZTlBZFhrMERaUjhmR1dZYjdkViJ9.eyJpc3MiOiJodHRwczovL2twaS5ldS5hdXRoMC5jb20vIiwic3ViIjoiSkl2Q081YzJJQkhsQWUycGF0bjZsNnE1SDM1cXh0aTBAY2xpZW50cyIsImF1ZCI6Imh0dHBzOi8va3BpLmV1LmF1dGgwLmNvbS9hcGkvdjIvIiwiaWF0IjoxNzEzMDExNjk1LCJleHAiOjE3MTMwOTgwOTUsInNjb3BlIjoicmVhZDp1c2VycyBjcmVhdGU6dXNlcnMiLCJndHkiOiJjbGllbnQtY3JlZGVudGlhbHMiLCJhenAiOiJKSXZDTzVjMklCSGxBZTJwYXRuNmw2cTVIMzVxeHRpMCJ9.QJoyWnQEsQDqlWDUcqeKLaonvHaciAyyp87LIQNQ5P9wHp3HWvrQvZTqGwm0MbereR09VT9_Qie9F2zUzlmYsCafNHAWm2rAPQ3qVBt6GIU_-nVB49R3gzYZhJTl0reZ65cljgKY17A4dkOhmHrDlrXMTMp9f6OPWFvVZhBOLY5_e0N-IWhyXDf2ugYZusi37ea9tL0mqctL4mHKxFnKxDuetEDA6GX39ju6hylOP0JcV9ZnFsJkhtNSqeMrKnjIF2x32U55HEJsG_puZ7Yq3GWaPd8BcM-YptOIfrb_LQAM-J0KVOxYI2hqwdQPtmafjeeL2PhMsw8yKOKLAY-img","scope":"read:users create:users","expires_in":86400,"token_type":"Bearer"}
C:\Users\Саша Головня\Desktop\security\lab2>
```

Ми отримали наш токен:

{"access_token":"eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6IjVCZTlBZFhrMERaUjhmR1dZYjdkViJ9.eyJpc3MiOiJodHRwczovL2twaS5ldS5hdXRoMC5jb20vIiwic3ViIjoiSkl2Q081YzJJQkhsQWUycGF0bjZsNnE1SDM1cXh0aTBAY2xpZW50cyIsImF1ZCI6Imh0dHBzOi8va3BpLmV1LmF1dGgwLmNvbS9hcGkvdjIvIiwiaWF0IjoxNzEzMDExNjk1LCJleHAiOjE3MTMwOTgwOTUsInNjb3BlIjoicmVhZDp1c2VycyBjcmVhdGU6dXNlcnMiLCJndHkiOiJjbGllbnQtY3JlZGVudGlhbHMiLCJhenAiOiJKSXZDTzVjMklCSGxBZTJwYXRuNmw2cTVIMzVxeHRpMCJ9.QJoyWnQEsQDqlWDUcqeKLaonvHaciAyyp87LIQNQ5P9wHp3HWvrQvZTqGwm0MbereR09VT9_Qie9F2zUzlmYsCafNHAWm2rAPQ3qVBt6GIU_-
nVB49R3gzYZhJTl0reZ65cljgKY17A4dkOhmHrDlrXMTMp9f6OPWFvVZhBO

LY5_e0N-
IWhyXDf2ugYZusi37ea9tL0mqctL4mHKxFnKxDuetEDA6GX39ju6hylOP0JcV9
ZnFsJkhtNSqeMrKnjIF2x32U55HEJsG_puZ7Yq3GWaPd8BcM-
YptOIfrb_LQAM-J0KVOxYI2hqwdQPtmafjeeL2PhMsw8yKOKLAY-
img","scope":"read:users create:users","expires_in":86400,"token_type":"Bearer"}

2.

Створимо користувача з іменем Sasha Holovnia, скринькою fakeyealow@gmail.com та ніком YeaLow. Запит матиме такий вигляд:

curl --request POST --url "https://kpi.eu.auth0.com/api/v2/users" --header "Content-Type: application/json" --header "Accept: application/json" --data "{\"email\":\"fakeyealow@gmail.com\",\"user_metadata\":{},\"blocked\":false,\"email_verified\":false,\"app_metadata\":{},\"given_name\":\"Sasha\",\"family_name\":\"Holovnia\",\"name\":\"Sasha\",\"nickname\":\"YeaLow\",\"picture\":\"https://ktpu.kpi.ua/wp-content/uploads/2016/12/89dd0d645592478b0b8933e8eb525806.jpg\",\"connection\":\"Username-Password-Authentication\",\"password\":\"IP11Holovnia\",\"verify_email\":false}" --header "Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6IjVCZTlBZFhrMERaUjhmR1dZYjdkViJ9.eyJpc3MiOiJodHRwczovL2twaS5ldS5hdXRoMC5jb20vIiwic3ViIjoiSkl2Q081YzJJQkhsQWUycGF0bjZsNnE1SDM1cXh0aTBAY2xpZW50cyIsImF1ZCI6Imh0dHBzOi8va3BpLmV1LmF1dGgwLmNvbS9hcGkvdjIvIiwiaWF0Ijox NzEzMDExNjk1LCJleHAiOjE3MTMwOTgwOTUsInNjb3BlIjoicmVhZDp1c2VycyBjcmVhdGU6dXNlcnMiLCJndHkiOiJjbGllbnQtY3JlZGVudGlhbHMiLCJhenAiOiJKSXZDTzVjMklCSGxBZTJwYXRuNmw2cTVIMzVxeHRpMCJ9.QJoyWnQEsQDqlWDUcqeKLaonvHaciAyyp87LIQNQ5P9wHp3HWvrQvZTqGwm0MbereR09VT9_Qie9F2zUzlmYsCafNHAWm2rAPQ3qVBt6GIU_-nVB49R3gzYZhJTl0reZ65cljgKY17A4dkOhmHrDlrXMTMp9f6OPWFvVZhBOLY5_e0N-
IWhyXDf2ugYZusi37ea9tL0mqctL4mHKxFnKxDuetEDA6GX39ju6hylOP0JcV9

ZnFsJkhtNSqeMrKnjIF2x32U55HEJsG_puZ7Yq3GWaPd8BcM-YptOIfrb_LQAM-J0KVOxYI2hqwdQPtmafjeeL2PhMsw8yKOKLAY-img

Результат виконання – створений користувач, що бачимо на рисунку:

Додаткове завдання:

Створимо файл ENV для збереження ключів, токенів:

AUTH0_URL = https://dev-eh2lg4o74zpq0730.us.auth0.com

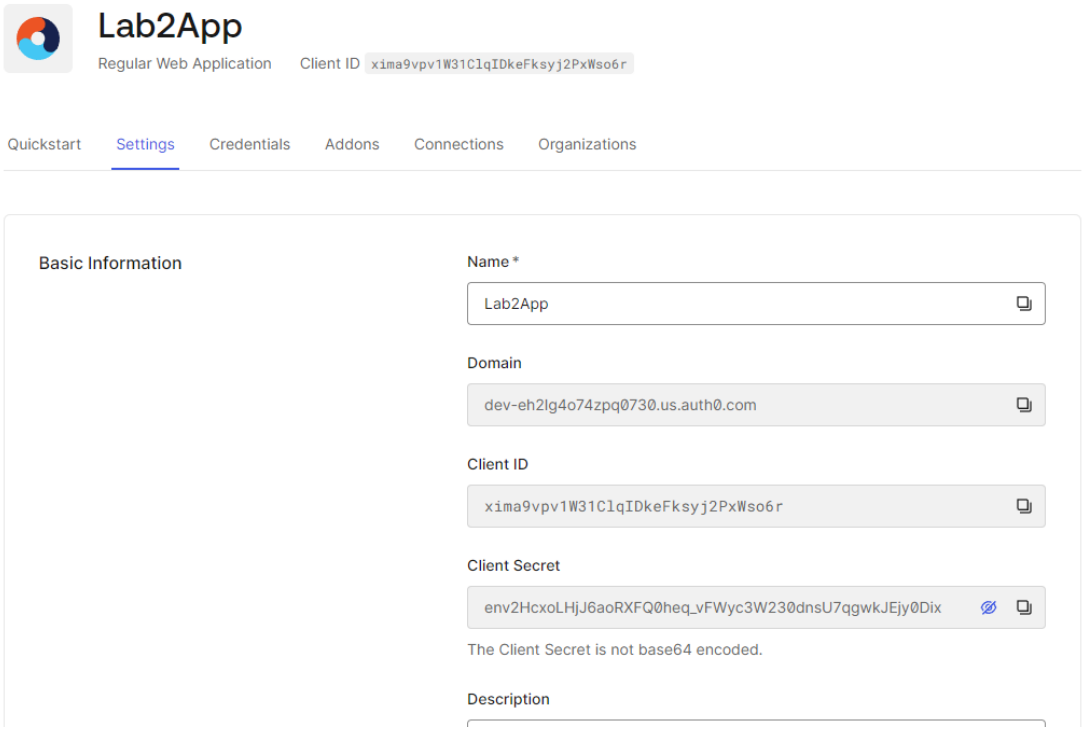
AUTH0_TOKEN=

AUTH0_CLIENT_ID= xima9vpv1W31ClqIDkeFksyj2PxWso6r

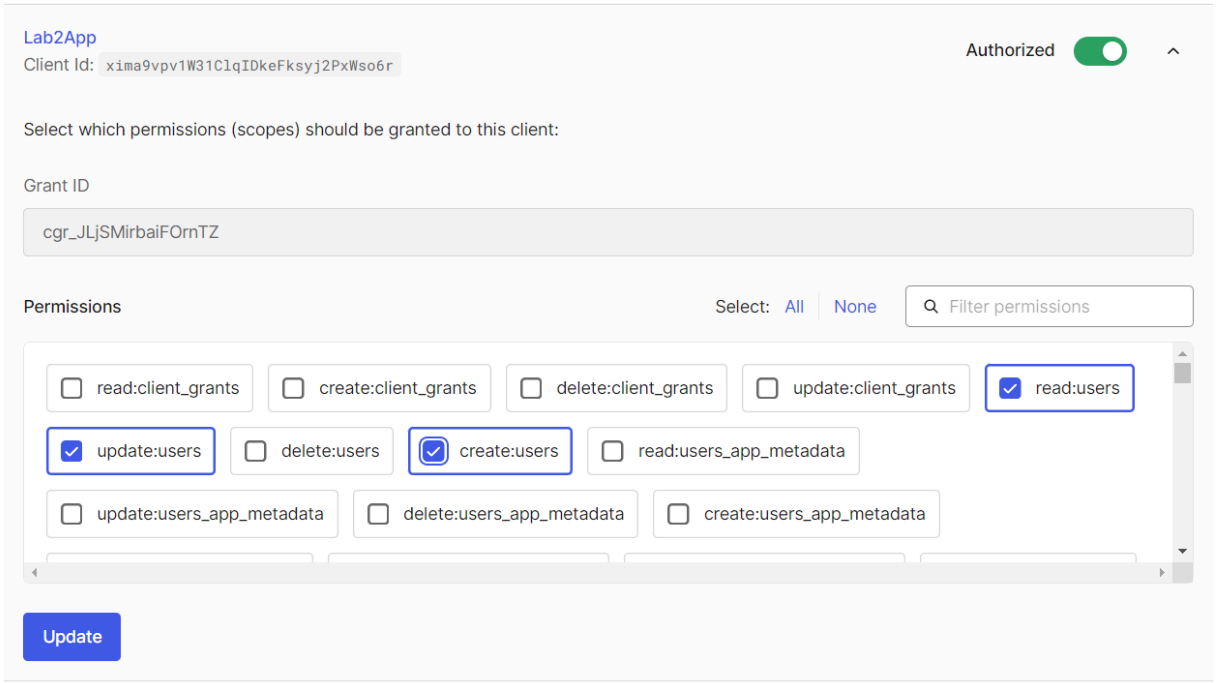
AUTH0_CLIENT_SECRET=

env2HcxoLHjJ6aoRXFQ0heq_vFWyc3W230dnsU7qgwkJEjy0DixRqmgVhseoz_vO

Створимо application lab2app:

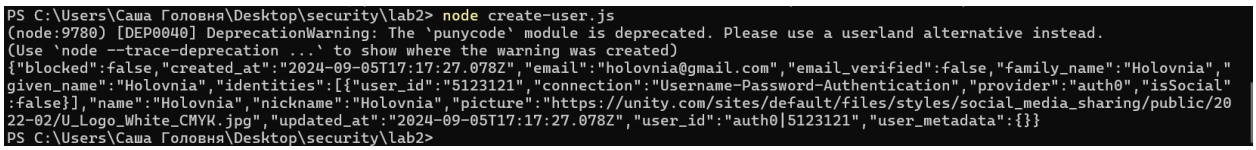


Надаємо доступи в API для нашого application:



За допомогою js файлу main.js, отримуємо токен:

За допомогою js файлу create-user.js, створимо користувача:



Висновки: У ході виконання лабораторної роботи було отримано практичні навички роботи з OAuth2. Завдання полягало у отриманні токена у системі КПІ, а також створення користувача.

Лістинг коду.

```
const uuid = require('uuid');
const express = require('express');
const onFinished = require('on-finished');
const bodyParser = require('body-parser');
const path = require('path');
const port = 3000;
const fs = require('fs');
const jwt = require('jsonwebtoken');

const app = express();
app.use(bodyParser.json());
app.use(bodyParser.urlencoded({ extended: true }));

const JWT_SECRET = 'secret';

const users = [
    {
        login: 'Login2',
        password: 'Password2',
        username: 'Username2',
```

```
    },
    {
        login: 'Login1',
        password: 'Password1',
        username: 'Username1',
    }
]

app.get('/', verifyToken, (req, res) => {
    res.json({
        username: req.user.username,
        logout: 'http://localhost:3000/logout'
    });
})

app.get('/logout', (req, res) => {
    res.redirect('/');
});

app.post('/api/login', (req, res) => {
    const { login, password } = req.body;

    const user = users.find((user) => {
        return user.login === login && user.password === password;
    });

    if (user) {
        const token = jwt.sign({ username: user.username, login: user.login },
JWT_SECRET);

        res.json({ token });
    } else {
        res.status(401).send('Unauthorized');
    }
});

app.listen(port, () => {
    console.log(`Example app listening on port ${port}`)
})

function verifyToken(req, res, next) {
    const token = req.header('Authorization');

    if (!token) {
        return res.sendFile(path.join(__dirname + '/index.html'));
    }

    jwt.verify(token, JWT_SECRET, (err, decoded) => {
        if (err) {
            return res.sendFile(path.join(__dirname + '/index.html'));
        }

        req.user = decoded;
        next();
    });
}
```