**Міністерство освіти і науки України  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки**

Лабораторна робота 4

з дисципліни  
«Безпера програмного забезпечення»

Виконав: Перевірив:

студент групи ІП-03 Іваніщев Б.В   
 Янишин Віталій

Київ 2023

**Лабораторна робота 4**

Засвоювання базових навичок OAuth2 авторизаційного протокола

**Завдання:**

Використовуючи наведені налаштування з лабораторної роботи 2 - 3 та приведених запитів модифікувати аплікейшен <https://github.com/Kreolwolf1/auth_examples/tree/main/token_auth>

Використовуючи перевірку юзера та отримання токена з auth0 (password grant type)

Надати код модифікованного аплікейшена.

**Для отримання додаткового балу:** додатково розшири аплікайшен створенням юзера та перевіркою життя токена (у разі близького завершення – оновити токен використовуючи refresh-token grant type)

**Виконання:**

Модифікований код на отримання токена через password grant type:

app.post("/api/login", (req, res) => {

  const { login, password } = req.body;

  const options = {

    method: "POST",

    url: `https://${yourDomain}/oauth/token`,

    headers: { "content-type": "application/x-www-form-urlencoded" },

    form: {

      client\_id: client\_id,

      client\_secret: client\_secret,

      audience: API\_IDENTIFIER,

      grant\_type: 'password',

      username: login,

      password: password,

    },

  };

  request(options, (error, response, body) => {

    if (error) {

      console.error(error);

      obj.status(401).send();

    }

    const obj = JSON.parse(body);

    console.log(obj);

    req.session.username = login;

    req.session.login = login;

    req.session.access\_token = obj.access\_token;

    req.session.expires\_at = Math.floor(Date.now() / 1000) + obj.expires\_in;

    req.session.refresh\_token = obj.refresh\_token;

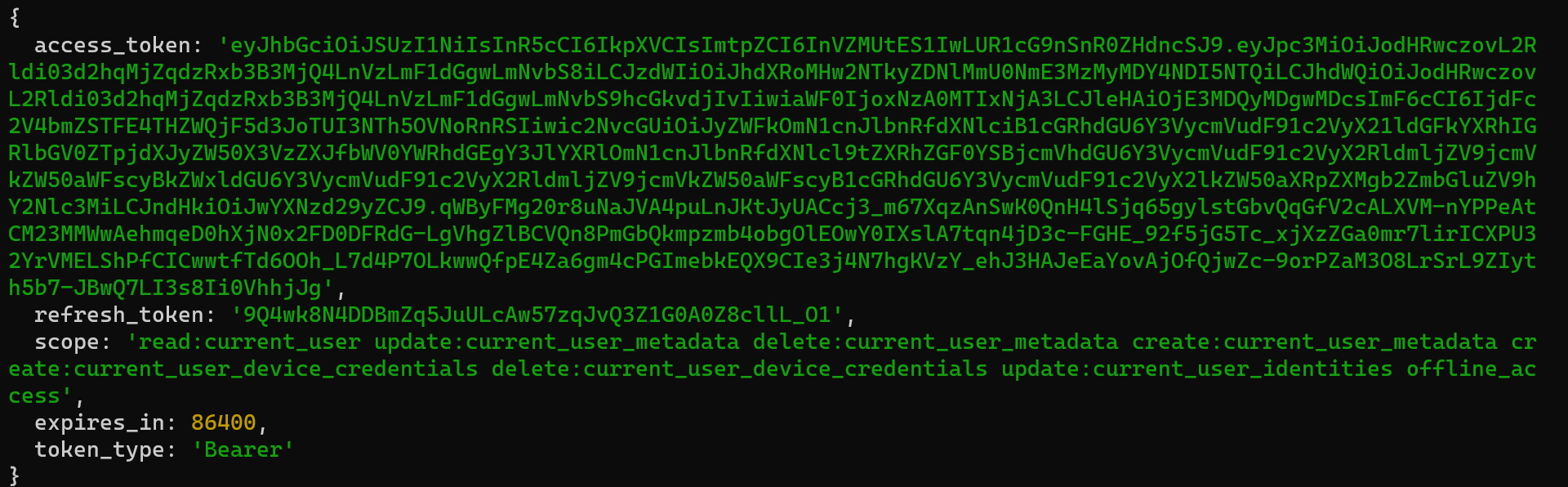
    res.json({ token: req.sessionId });

  });

});

**Скріншоти:**





Створення юзера:

app.post("/api/login", (req, res) => {

  const { login, password } = req.body;

  const options = {

    method: "POST",

    url: `https://${yourDomain}/oauth/token`,

    headers: { "content-type": "application/x-www-form-urlencoded" },

    form: {

      client\_id: client\_id,

      client\_secret: client\_secret,

      audience: API\_IDENTIFIER,

      grant\_type: "password",

      username: login,

      password: password,

      scope: 'offline\_access',

    },

  };

  request(options, (error, response, body) => {

    if (error) {

      console.error(error);

      obj.status(401).send();

    }

    const obj = JSON.parse(body);

    console.log(obj);

    req.session.username = login;

    req.session.login = login;

    req.session.access\_token = obj.access\_token;

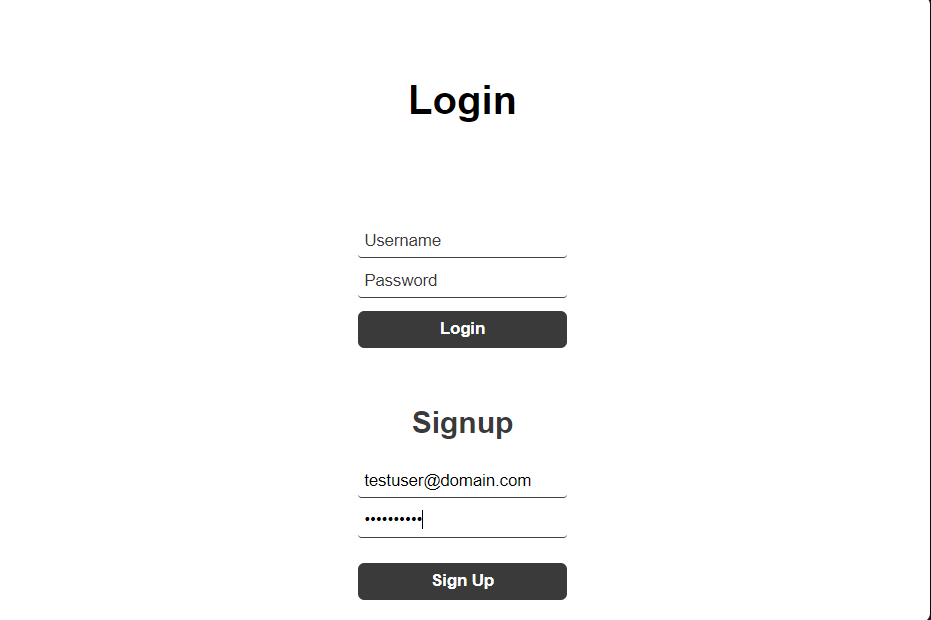
    req.session.expires\_at = Math.floor(Date.now() / 1000) + obj.expires\_in;

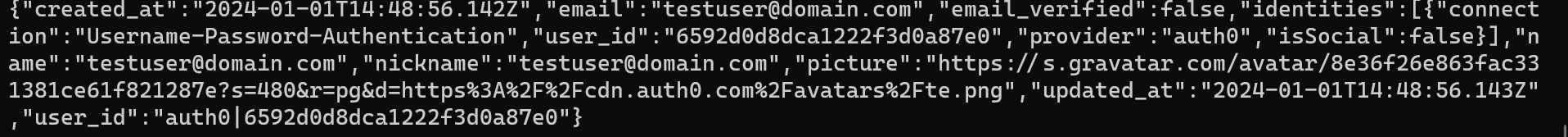
    req.session.refresh\_token = obj.refresh\_token;

    res.json({ token: req.sessionId });

  });

});





Оновлення токена:

Токен оновлюється кожен раз при завантаженні сторінки(для демонстрації)

const refreshToken = (refreshToken) => {

  return new Promise((resolve, reject) => {

    const options = {

      method: "POST",

      url: `https://${yourDomain}/oauth/token`,

      headers: {

        "content-type": "application/x-www-form-urlencoded",

      },

      form: {

        client\_id: client\_id,

        client\_secret: client\_secret,

        grant\_type: "refresh\_token",

        refresh\_token: refreshToken,

      },

    };

    request(options, (error, response, body) => {

      if (error) reject(error);

      resolve(body);

    });

  });

};

app.get("/", async (req, res) => {

  if (req.session.access\_token) {

    if (true) {

      const response = await refreshToken(req.session.refresh\_token);

      const responseObj = JSON.parse(response);

      req.session.access\_token = responseObj.access\_token;

      req.session.expires\_at =

        Math.floor(Date.now() / 1000) + responseObj.expires\_in;

      console.log(`token refreshed:

        ${response}

      `);

    }

    return res.json({

      username: req.session.username,

      logout: "http://localhost:3000/logout",

    });

  }

  res.sendFile(path.join(\_\_dirname + "/index.html"));

});



