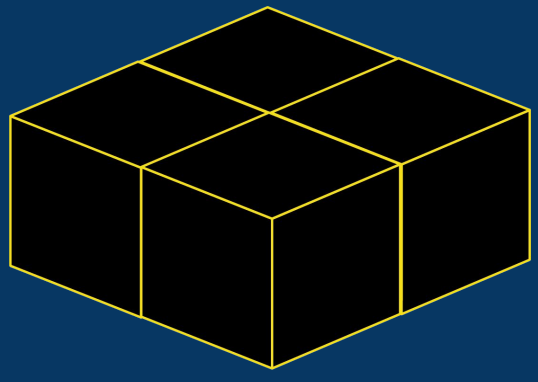


Curso de
Autenticación con Passport.js

Guillermo Rodas

@glrodasz

Guillermo Rodas

@glrodasz

Engineer at Auth0

GDE Web Technologies

Co-organizer of Medellín CSS, CSS Conf CO and Medellín Identidad y Seguridad


Auth0
https://auth0.com/careers


SALIDA
NIKE SB
Wilson
angular
Medellín
CSS
https://www.meetup.com/medellincss

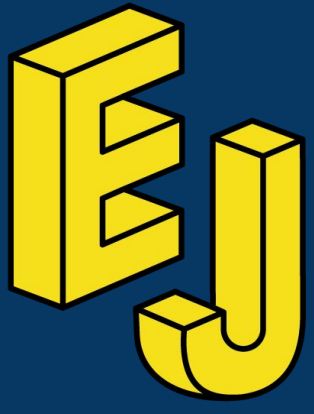
IDENTITY & SECURITY
Meetup

Medellín

Identidad y Seguridad

https://www.meetup.com/Medellin-Identidad-y-Seguridad


CSS
conf
COLOMBIA

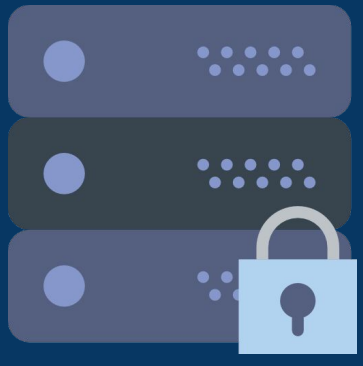


Escuela de JavaScript

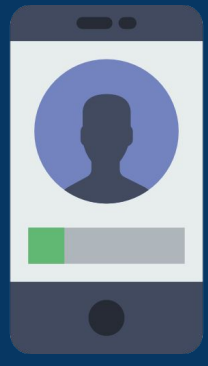
Requisitos

Conocimientos básicos
de Node.js

Stack de seguridad moderno



Classic security is intranet-only



Mobile Revolution

Stack de seguridad moderno

JSON Web Tokens

OAuth 2.0

OpenID Connect

¿Qué es la Autenticación?

Verificar la identidad
de un usuario

¿Qué es la Autorización?

Permitir a un usuario acceso limitado a nuestros recursos.

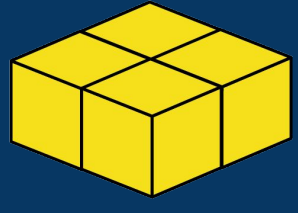
Introducción a las sesiones

Resumen

- Stack de seguridad moderno
- ¿Qué es la autenticación?
- ¿Qué es la autorización?
- Introducción a las sesiones

Challenge

Compartir un ejemplo de autenticación y autorización



Conoce qué son los JSON Web Tokens

Anatomía de un JWT

RFC 7519

JSON Web Token

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c

Header

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

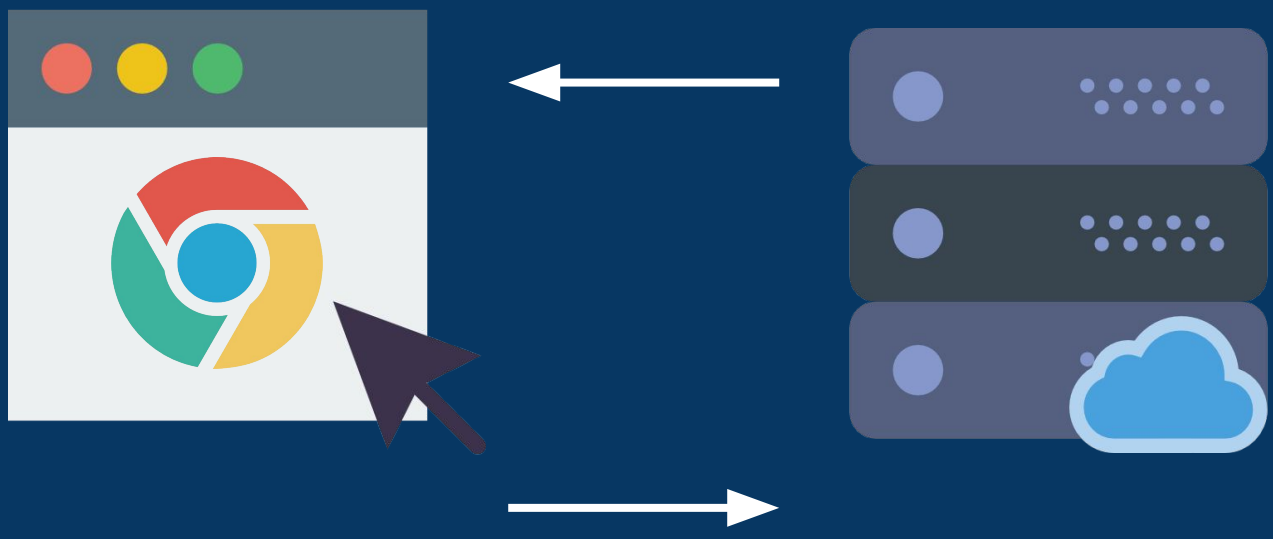
Payload

```
{
  "sub":
"1234567890",
  "name": "John
Doe",
  "iat": 1516239022
}
```

Signature

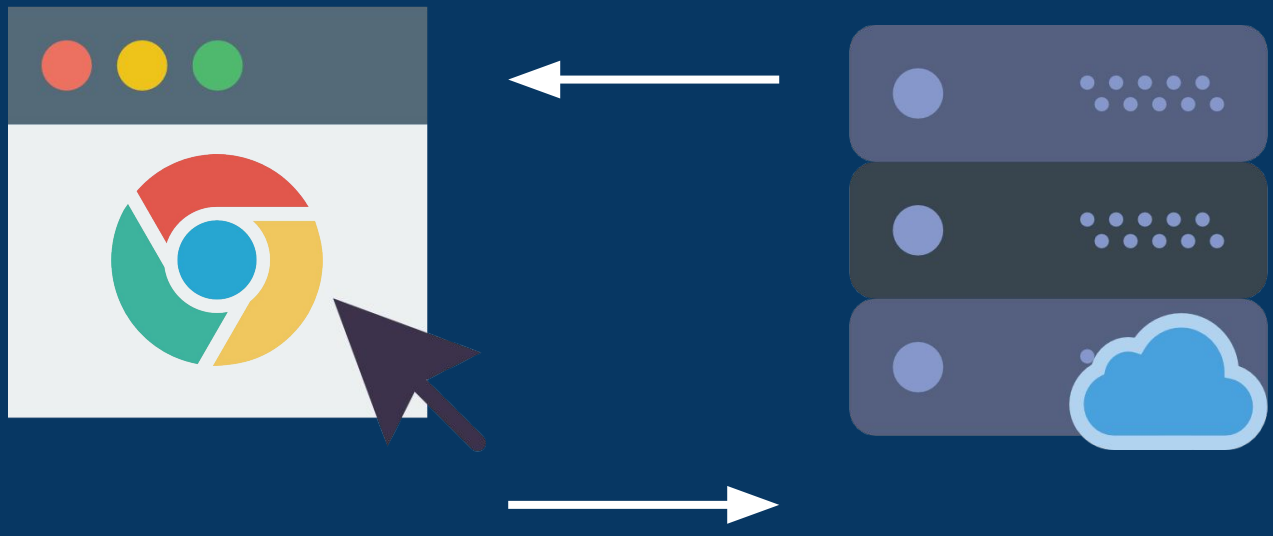
```
HMACSHA256(
  base64UrlEncode(header) +
"." +
  base64UrlEncode(payload),
  your-256-bit-secret
)
```

Autenticación tradicional vs JWT



Cookie
Created

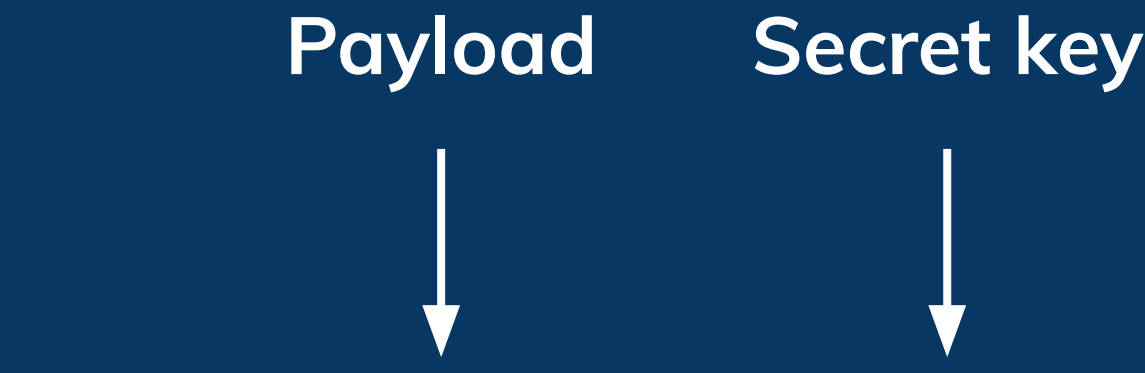
Session
Created



Token
Saved

Token
Signed

Firmando nuestro JWT




```
jwt.sign({ sub: user.id }, 'secret',
options);
```

Verificando nuestro JWT

Secret key

Decoded token

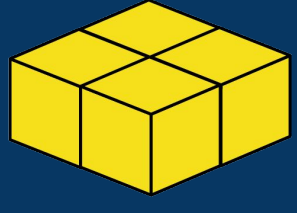
```
jwt.verify(token, 'secret', function(err,
decoded){});
```

Resumen

- Anatomía de un JWT
- Autenticación tradicional vs JWT
- Firmando nuestro JWT
- Verificando nuestro JWT
- Server-side vs Client-side sessions
- Buenas prácticas con JWT

Challenge

Crea un JWT y hazle debugging usando jwt.io



Cómo funcionan las Cookies

¿Qué son las cookies y para qué sirven?

Manejo de sesión usando Cookies

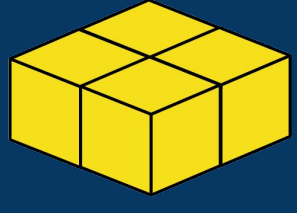
Cookies vs Session Storage vs Local Storage

Resumen

- ¿Qué son las cookies y para qué sirve?
- Manejo de sesión usando Cookies
- Cookies vs Session Storage vs Local Storage

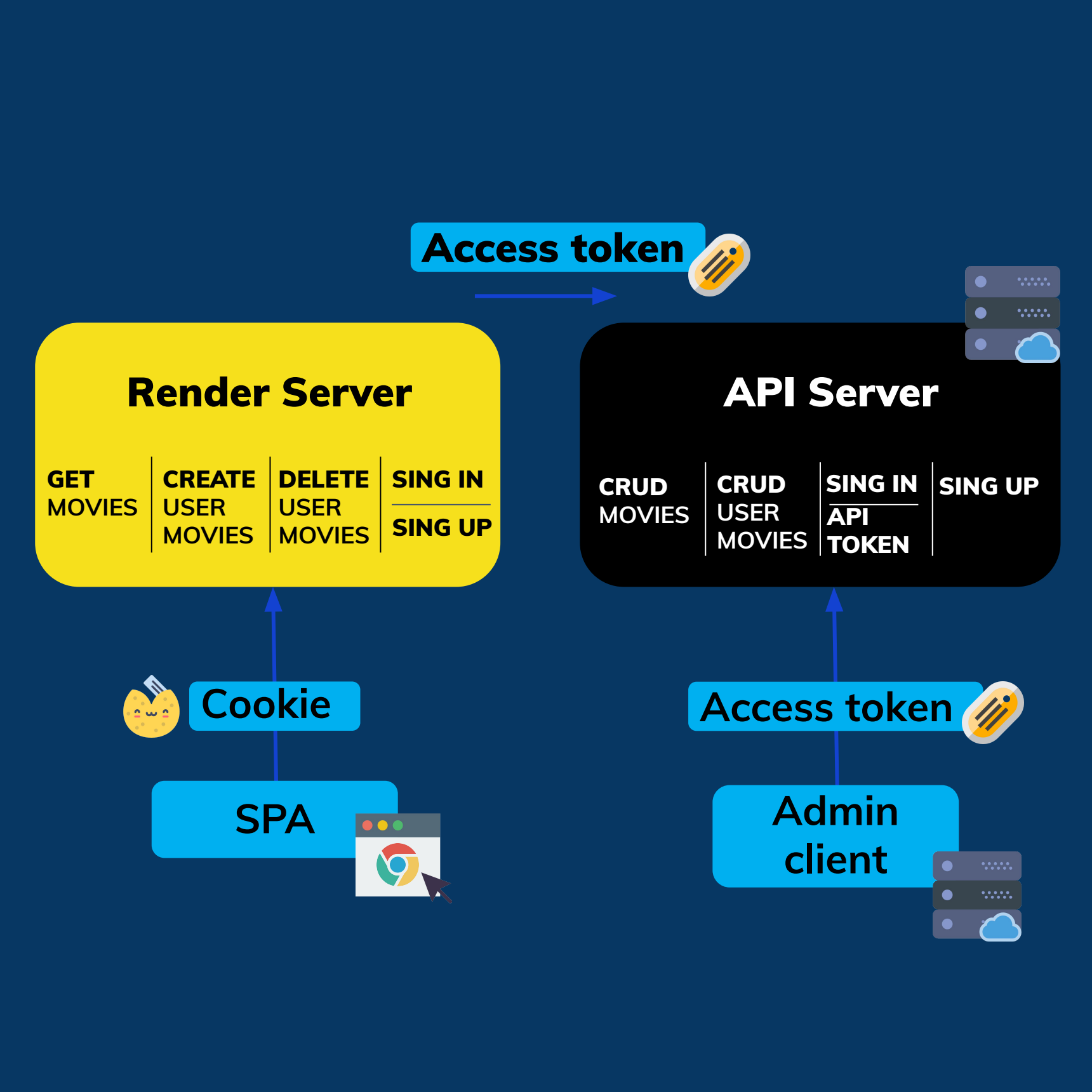
Challenge

Discutir si fuéramos a implementar autenticación que opción entre Cookies y Web Storage sería ideal



Implementa autenticación en Express usando Passport.js

Arquitectura del proyecto Platzi Video


Access token
Render Server
GET MOVIES
CREATE USER MOVIES
DELETE USER MOVIES
SING IN
SING UP
API Server
CRUD MOVIES
CRUD USER MOVIES
SING IN
API TOKEN
SING UP
Cookie
Access token
SPA
Admin client

Agregando la colección de usuarios

Agregando la colección de películas de usuario

Configuración de Passport.js

Implementación de estrategias en Passport.js

Implementación de nuestro Sign-in

Implementación de nuestro Sign-up

Protegiendo nuestras rutas con Passport.js

Middleware para el manejo de scopes

Configuración del Render Server

Comunicación máquina a máquina

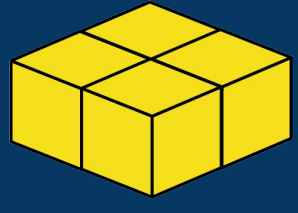
Implementación de las rutas de las películas de usuario

Resumen

- Configuración de nuestro proyecto
- Configuración de Passport.js
- Implementación de estrategias en Passport.js
- Implementación de nuestro Sign-in y Sign-up
- Protegiendo nuestras rutas con Passport.js
- Implementando recordar sesión
- Comunicación máquina a máquina

Challenge

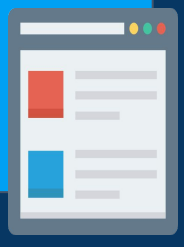
Diferencia entre Authorization basic y bearer



¿Qué es OAuth 2.0 y OpenID Connect?

¿Qué es OAuth 2.0?

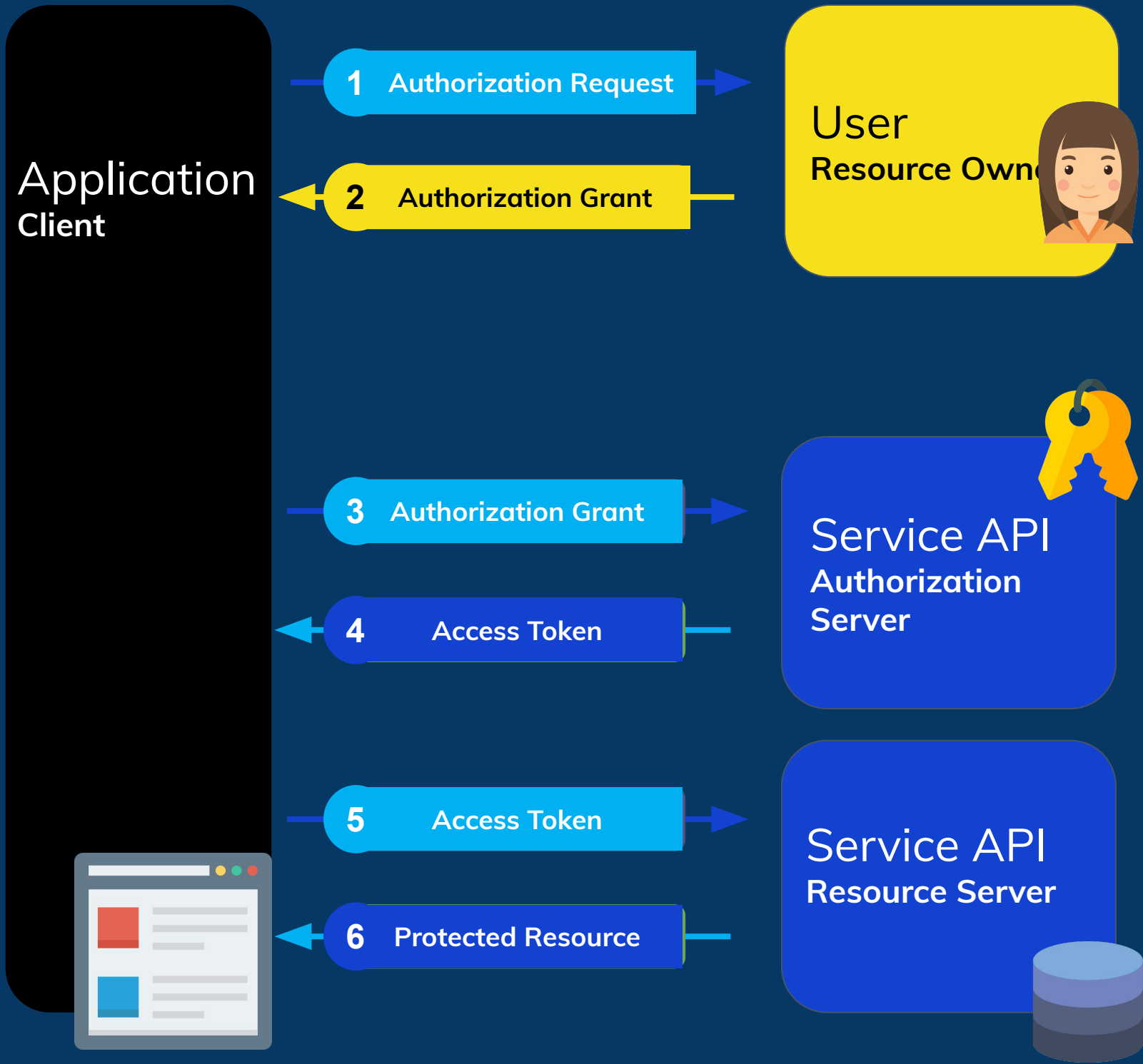
Application
Client



Service API
Authorization Server

User
Resource Owner

Service API
Resource Server


Application
Client
1 Authorization Request
2 Authorization Grant
User
Resource Owne
3 Authorization Grant
4 Access Token
Service API
Authorization
Server
5 Access Token
6 Protected Resource
Service API
Resource Server

Brother
Client

1 Authorization Request

2 Authorization Grant

User
Resource Owne

3 Authorization Grant

4 Access Token

Parents
Authorization Server

5 Access Token

6 Protected Resource

Toy closet
Resource Server

¿Qué es OpenID Connect?

Identity

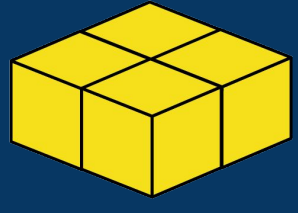
User information

Resumen

- ¿Qué es OAuth 2.0?
- ¿Qué es OpenID Connect?

Challenge

Listar las diferencias principales entre OAuth 2.0 y OpenID Connect



¿Cómo implementar autenticación con redes sociales?

Autenticación con Google usando OAuth 2.0

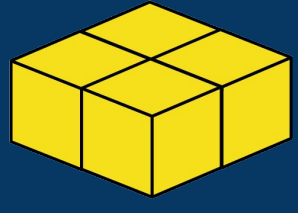
Autenticación con Twitter

Resumen

- Autenticación con Google
- Autenticación con Twitter
- Autenticación con Facebook

Challenge

Implementar autenticación con LinkedIn



Asegura tu aplicación de Express

Seguridad con Helmet

Vulnerabilidades con npm audit

Automatizar el chequeo de vulnerabilidades con Snyk

¿Qué es OWASP? y buenas prácticas de seguridad

Open Web Application Security Project

- Injection
- Broken Authentication
- Sensitive Data Exposure

Buenas prácticas

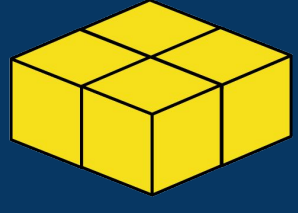
- Usa un gestor de contraseñas
- Usa multi-factor auth
- IRL security
- Mantén actualizadas tus aplicaciones y SO
- Mantén actualizadas tus dependencias
- Mantente informado

Resumen

- Seguridad con Helmet
- Vulnerabilidades con npm audit
- Automatizar el chequeo de vulnerabilidades con Snyk
- ¿Qué es OWASP y buenas prácticas de seguridad?

Challenge

Investiga que middleware podemos usar para mitigar el top 10 de vulnerabilidades de OWASP



Conclusiones

@glrodasz