Angular 5 + JWT Autorización simple

Introducción

Hola a Todos bienvenidos una vez más a mis videos tutoriales pelfsoft: Mi nombre Es MO y voy a intentar de mostrarlos como hacer que una app FrontEnd se autentique y transmita información segura a travez de peticiones REST FULL

Arquitectura

La arquitectura global de nuestro desarrollo estará dividida en dos partes:

# Por un lado tendremos el BackEnd :

Que será el encargado de autenticación a través una WEB Rest API totalmente desacoplada del cliente y será desarrollada con NODE JS / Express

# Por el otro Tendremos el cliente/app

Que será una aplicación desarrollada con Angular 5 basado en el estructurado scaffolding propuesto por Angular CLI (Explicar un poco)

### Angular CLI que es?

**https://github.com/angular/angular-cli/**

Es una herramienta que permite inicializar, estructurar y generar fácilmente componentes para una aplicación Angular2 de una forma ordenada y estandarizada: Propone una serie de comandos que permiten crear módulos componentes servicios etc También hay comandos para compilar y hacer distribuciones de raleases.

En otras palabras nos da, una vez generada la app básica, todo un ecosistema completo para q podamos ejecutar compilar testear y distribuir fácilmente nuestro app.

Angular2

# Autenticación

La autenticación se realizara a travez de un estándar 🡪 JWT que define una manera compacta y segura de transmitir información entre las partes como JSON objects..-

La información puede ser verificada dado que es digitalmente cifrada con un algoritmo HMAC (simétrico) o RSA Asimétrico con clave pub/priv

Cuando la podemos usar a la seguridad basada en tokens?

**A) Authentication**

**B) Information Exchange**: Se puede asegurar de q quien envía cierta data es quien dice ser mediante el uso de clave pública privada

Formato de un Token:

* **Header** : aquí ponemos el algoritmo ej a: "HS256",
* **Payload** : contiene el Claims o datos de la Indentity Usuario perfilt etc

*{ "sub": "1234567890", "name": "John Doe", "admin": true}*

* **Signature** = Es la firma digital que tienen en cuenta

base64(header) + base64(payload) + SECRET

Ejemplo eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.XbPfbIHMI6arZ3Y922BhjWgQzWXcXNrz0ogtVhfEd2o

Bien …habiendo ya echo una vista muy global de la teoria pasaremos a lo practico y manos al codigo:

**Paso 1**

Comprobar que tengamos instalado los siguientes ítems

* Node: 8.9.1
* Angular CLI: 1.6.7
* Angular: 5.2.3
* typescript: 2.4.2
* Git 2.16.2.
* Visual **code**
* **Contar con la herramienta Postman para hacer pruebas a nuestro server**

<http://localhost:8080/api/pub/users/>

Paso dos: Arquitectura

Arquitectura



# Estructura

apiServer :

server.js Instancia el servidor en un Puerto determinado. Es el start-up del servicio

middleware: Modulo que configura express funciones de seguridad, de trazabilidad , CORS (para permitir peticiones de otros dominios)

bodyParser: permite que cuando un navegador haga un post desde un formulario puedan ser interpretados puedan ser interpretados vengan como vengan : Form,

Funciones que permitan tracear el uso de la app

App.use significa: usar en todas las funciones lo siguiente:

App.use((req,res)=>{

//hacer halgo para todos

next(); continuar

});