DigitalHouse >
Coding School

JWT
Json Web Token

“

JWT es el acrónimo de **JSON Web Token** y en esencia es un estándar para hacer autenticación de token de forma segura y confiable.

JWT:

ANATOMÍA

ANATOMÍA DE UN JWT

JWT está dividido en 3 secciones separadas por puntos:

1. **Encabezado**
   a. Se encarga de describir el Token.
2. **Payload**
   a. Contiene la petición **JWT**.
3. **Firma**
   a. Se usa para verificar la integridad de un **JWT**.

DECODIFICANDO UN PAYLOAD

Cuando decodificamos un payload obtenemos este objeto JSON que podría ser parecido a:

{}

```
{
 "id": "123",
 "pet_id": "2"
}
```

En este caso indica que mi id en la base de datos es 123 y el id de mi mascota el 2.

Es decir, el payload es efectivamente la información que desees almacenar en el JWT.

DECODIFICANDO UN HEADER

Cuando decodificamos un header obtenemos este objeto JSON que podría ser parecido a:

{}

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

Este json describe primero que es un JWT y luego un algoritmo que de encriptación.

LA FIRMA

Por último la **firma** del **JWT** se genera usando los dos campos anteriores en base64 y una llave secreta que solo se sepa en los servidores que creen o usen el JWT para usar un algoritmo de encriptación.

Ejemplo en pseudocódigo:

```
HMACSHA256(BASE64URL(header)) . BASE64URL(payload,secret)
```

¿QUE PASA SI NO COINCIDE?

Si no coinciden es porque hay algo en el **header** o en el **payload** que fue alterado y el pedido no pasa la etapa de **autenticación**. Es por eso que la **encriptación** resulta clave para validar los pedidos.

En este caso, al **cambiar** el "secret-key", vemos que la firma no es válida "Invalid Signature".
Esto quiere decir que no podemos confiar en el Token JWT, porque que alguien lo puede haber firmado de forma maliciosa o cambiado algo del **payload**.