What is JSON Web Token?

{Json Web Token}

JWT is an open standard that defines a compact and self-contained way for securely transmitting information between parties as a JSON object. This information can be verified and trusted because it is digitally signed.

When should you use JSON Web Tokens?

Authorization: This is the most common scenario for using JWT.

Information Exchange: JWT are a good way of securely transmitting information between parties.

What is the JSON Web Token structure?

In its compact form, JWT consist of three parts separated by dots (`.`) , which are:

Header

Payload

Signature

Therefore, a JWT typically looks like the following.

```
xxxxx.yyyyy.zzzzz
```

Header

```
{
"alg" : "HS256"
"type" : "JWT"
}
```

Base64 encoded x.y.z

Payload

The second part of the token is the **payload**, which contains the claims. Claims are statements about an entity (typically, the user) and additional data.

There are three types of claims:

- **Registered** claims
- **Public** claims
- **Private** claims

Header

```
{
"alg" : "HS256"
"type" : "JWT"
}
```

Data

```
{
"key" : "foo"
}
```

Base64 encode

Base64 encode

x.y.z

Signature

To create the signature part you have to take the encoded header, the encoded payload, a secret, the algorithm specified in the header, and sign that.

For example if you want to use the HMAC SHA256 algorithm, the signature will be created in the following way:

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  secret)
```

1 login

username, password

2 validate credentials

3 create & sign JWT with secret

Authorization: Bearer JWT

4 store JWT locally

5 /resource/user

Authorization: Bearer JWT

validates signature

OK

6 data

Find it helpful?

Follow for more