

26. Authentication (Backend)

Table of Content

1. JWT access token and refresh tokens

1. JWT access token and refresh tokens

▼ JSON Web Token

JSON Web Token (JWT) is an open standard based on JSON to create access tokens that allow the use of application or API resources. This token will incorporate the information of the user who needs the server to identify it, as well as additional information that may be useful (roles, permissions, etc.).

It may also have a **validity period**. Once this validity period has elapsed, the server will no longer allow access to resources with this token. In this step, the user will have to get a new access token by reauthentication or with some additional method: **refresh token**.

▼ Types of Token

There are many types of tokens, although in authentication with JWT the most typical are **access token and refresh token**.

- **Access token**: It contains all the information the server needs to know if the user / device can access the resource you are requesting or not. They have usually expired tokens with a short validity period.
- **Refresh token**: The refresh token is used to generate a new access token. Typically, if the access token has an expiration date, once it expires, the user would have to authenticate again to obtain an access token. With refresh token, this step can be skipped and with a request to the API get a new access token that allows the user to continue accessing the application resources.

▼ Flow

After an access token is generated, sometimes you might have to renew the old token due to expiration or security concerns. You can renew an access token using a refresh token, by issuing a REST call to the Token API with the following parameters. A refresh token has to be obtained before using it with a grant type such as the authorization code or password grant type. Using the obtained refresh token, you can obtain a new access token without having to go through any other additional steps.

The diagram below illustrates the refresh token grant flow.

