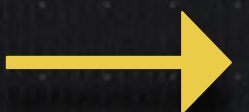


Introduction to **JWT** as an **Absolute Beginner**

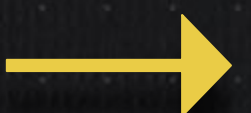


Vikas Rajput
@vikasrajputin



1. JWT stands for **JSON Web Token**

- It's a token that is used to authenticate and authorize users in an application.
- "authenticate" means who they're.
- "authorize" means what they can access.
- The token itself contains, all the necessary information about the user, like user ID and role, etc, in a JSON.



- JWT tokens are typically generated by the server and sent to the client after a successful login.
- The client can then use the JWT token (with each request) to authenticate and authorize itself to the server.
- Typically the token looks like this:

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.  
eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4  
gRG91IiwiaXNTb2NpYWwiOnRydWV9.  
4pcPyMD09olPSyXnrXCjTwXyr4BsezdI1AVTmud2fU4
```



2. JWT has three parts:

- ◆ Header (highlighted in red below)
 - ◆ Payload (highlighted in pink below)
 - ◆ Signature (highlighted in blue below)
- On left you can see the encoded token, on right we can see decoded JSON object with 3 parts.

The screenshot shows a JWT decoding interface with two main panels: 'Encoded' and 'Decoded'.

Encoded Panel: Contains a text area with the encoded token: `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJLV_adQssw5c`. The token is divided into three color-coded sections: red for the header, pink for the payload, and blue for the signature.

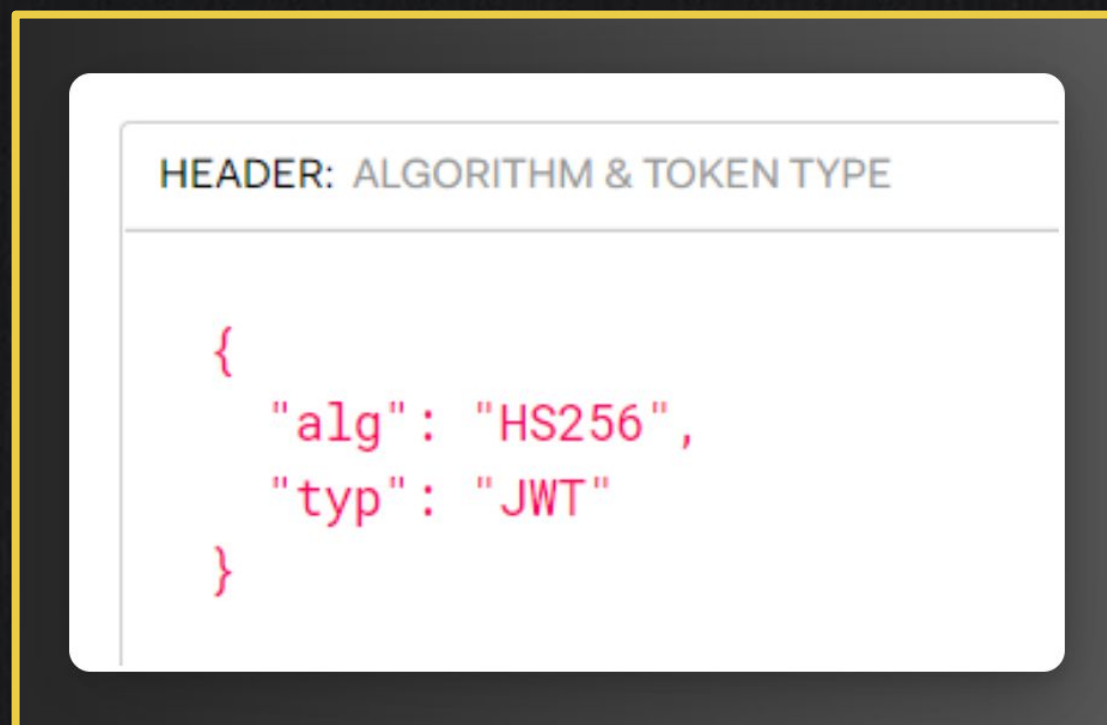
Decoded Panel: Shows the decoded components of the token:

- HEADER: ALGORITHM & TOKEN TYPE:** `{ "alg": "HS256", "typ": "JWT" }`
- PAYLOAD: DATA:** `{ "sub": "1234567890", "name": "John Doe", "iat": 1516239022 }`
- VERIFY SIGNATURE:** Shows the signature verification process using HMACSHA256, with a text input for the secret: `your-256-bit-secret`.

Red arrows indicate the mapping from the encoded token to the decoded sections: the first part of the token (red) maps to the header, the second part (pink) maps to the payload, and the third part (blue) maps to the signature verification section.



- The header typically consists of two parts: the type of the token, which is usually JWT, and the signing algorithm being used, such as HMAC SHA256 or RSA.



```
HEADER: ALGORITHM & TOKEN TYPE

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



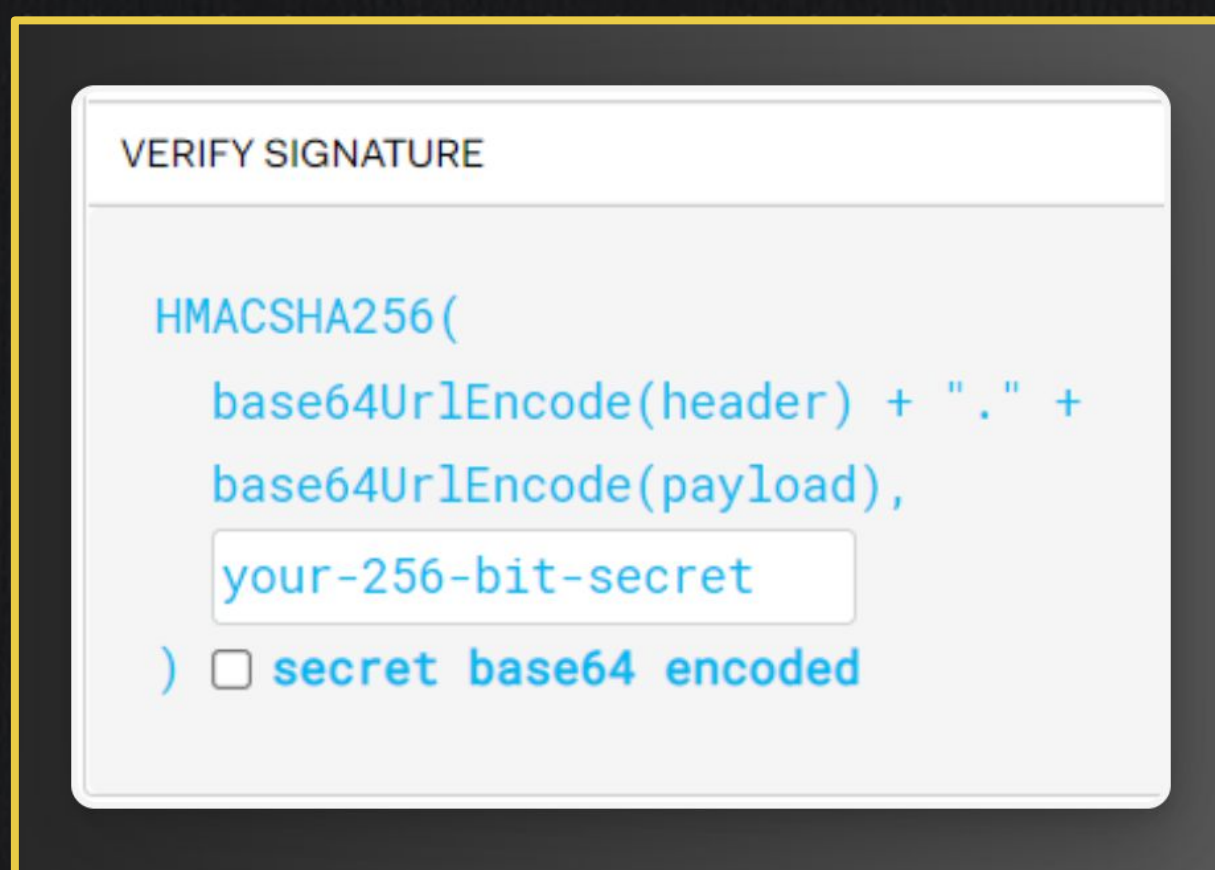
- The payload contains the claims, which are statements about an entity (typically, the user) and additional metadata.
- Claims are typically represented as key-value pairs and can include information such as the user's ID, name, email, and roles.

```
PAYLOAD: DATA

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



- The signature is used to verify that the sender of the JWT is who it says it is and to ensure that the message has not been tampered with.



- That's a quick introduction to JWT!
- We will see more in-depth concepts of JWT in the upcoming posts.



❤️ **Thanks** for reading !

For more content on
Java & Backend Development,
follow me on below handles



Vikas Rajput
@vikasrajputin

