

JWT Structure Explained

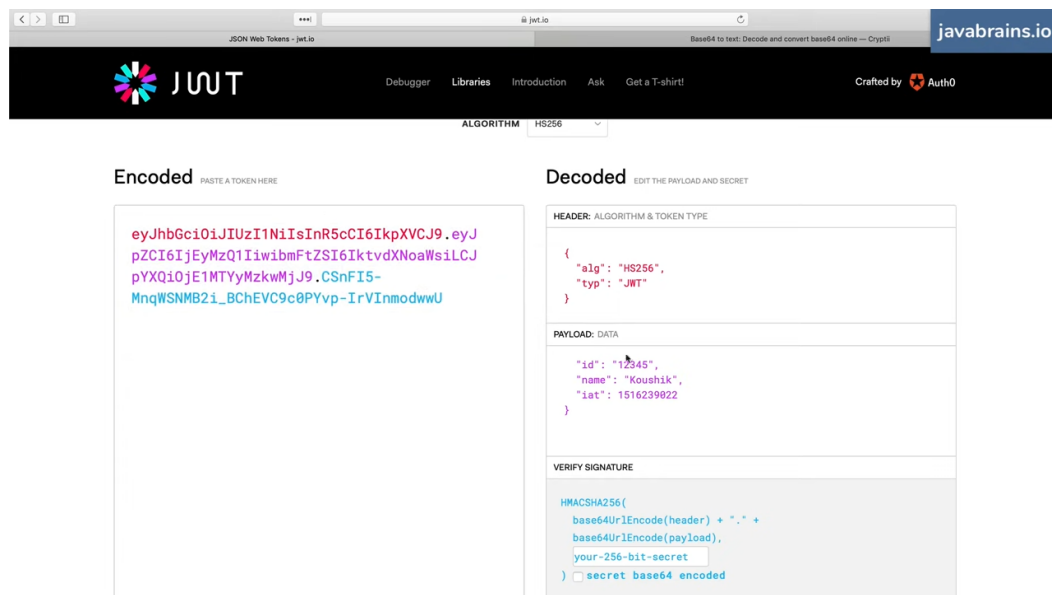
- 3 parts (separated by '.'): Header, Payload, Signature

javabrain.io

Sample JWT

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36P0k6yJV_adQssw5c
```

- jwt.io



The screenshot shows the JWT.io website interface. At the top, there's a navigation bar with the JWT logo and links for Debugger, Libraries, Introduction, Ask, and Get a T-shirt. Below the navigation bar, there's a section for "Encoded" and "Decoded". The "Encoded" section has a text area with a sample JWT token: `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36P0k6yJV_adQssw5c`. The "Decoded" section shows the decoded token structure:

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

PAYLOAD: DATA

```
{
  "id": "i2345",
  "name": "Koushik",
  "iat": 1516239022
}
```

VERIFY SIGNATURE

```
HMACSHA256(
  base64urlEncode(header) + "." +
  base64urlEncode(payload),
  your-256-bit-secret
) ☐ secret base64 encoded
```

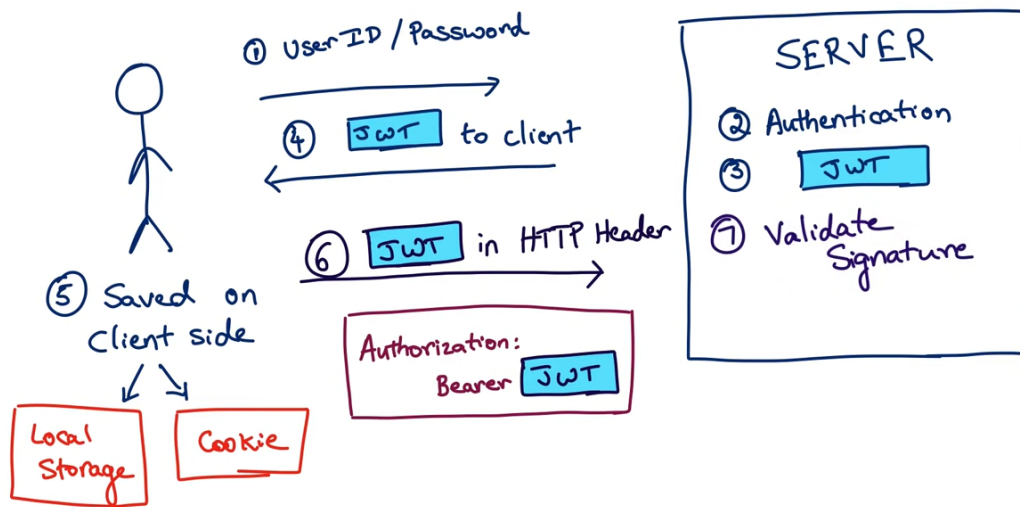
- Payload in JWT is a base64 encoding
- Header contains the type of token and the algorithm used to verify the signature
- base64 thing is only for convenience
- The signature is possible to be computed only by the server which issued it

- The signature is used (by the server) to verify if any change has been made to the token (payload?).
- $\text{signature} = f(\text{header}, \text{payload}, \text{secret})$, where f is a cryptographic hash function (the algo specified in the header) and the secret is only available to the server

JWT is for **authorization**

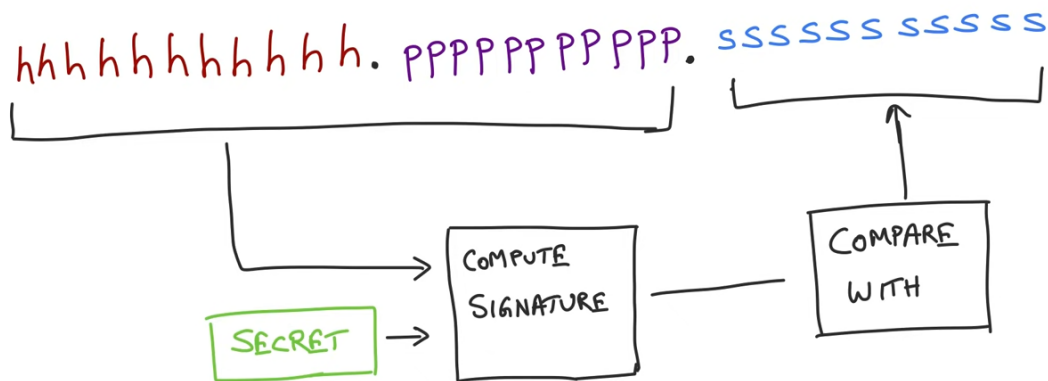
Overall flow

javabrain.io



Signature verification

javabrain.io



Issues

- Do not send confidential issues in payload (anyone can decode)
- What if someone gets hold of the full JWT itself?

To ensure the latter doesn't happen, use `https` or OAuth

- Disabling JWTs (in case of session IDs, it is just enough to terminate the session)

javabrain.io

