**What is a JWT?**

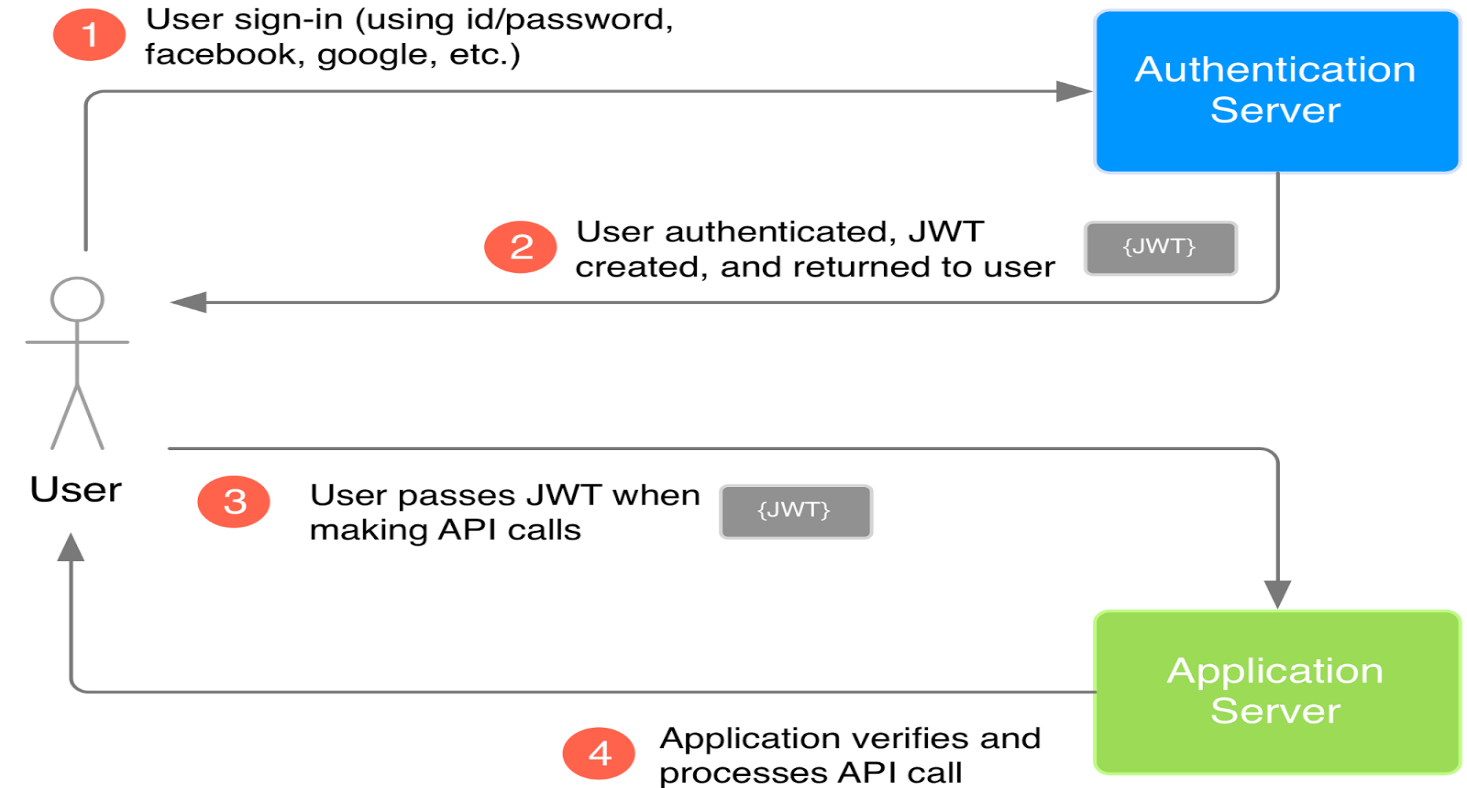
A **JSON Web Token (JWT)** is a secure way to transmit information between parties as a JSON object. It’s commonly used for:

* **Authentication** (verifying user identity)
* **Authorization** (checking user permissions)
* **Secure data exchange** (sending trusted user data between services)

**Why Use JWTs?**

* **Compact** – Can be sent via URL, POST, or HTTP headers.
* **Self-contained** – Stores user info (like ID & role) inside the token.
* **Secure** – Digitally signed to prevent tampering.

**How JWTs Work**



**1. User Logs In**

* The client (e.g., a web app) sends a **username & password** to the server.
* The server checks credentials.

**2. Server Creates a JWT**

If login is successful:

* The server generates a **signed JWT** containing user data.
* The token is sent back to the client.

**3. Client Uses JWT for Future Requests**

* The client stores the JWT (usually in **localStorage** or **cookies**).
* For every new request, the client sends the JWT in the Authorization **header**:

**http:** Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

**4. Server Validates the JWT**

* + The server checks:
* **Signature** (was this token issued by us?)
* **Expiry time** (is the token still valid?)
* **User permissions** (does this user have access?)
* If valid → request is approved.
* If invalid → returns 401 Unauthorized.

## ****JWT Structure****

A JWT has **3 parts**, separated by dots(.):

Text : xxxxx.yyyyy.zzzzz

1. **Header** (Algorithm & Token Type)

{

"alg": "HS256", // Signing algorithm (e.g., HMAC-SHA256)

"typ": "JWT" // Token type

}

1. **Payload** (User Data + Claims)

{

"sub": "12345", // Subject (user ID)

"name": "Gopika", // Custom data

"iat": 1712345678, // Issued At (timestamp)

"exp": 1712349278 // Expiry (timestamp)

}

1. **Signature** (Prevents Tampering)

HMACSHA256(

base64UrlEncode(header) + "." + base64UrlEncode(payload),

secretKey

)

## ****Testing JWTs with jwt.io****

1. Go to **jwt.io**
2. In the **"Decoded"** section:

* **Header**: Paste

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

* **Payload**: Paste

{"sub":"1234567890","name":"Gopika","iat":1516239022}

1. In **"Verify Signature"**, enter.

Secretkey

1. **Check the "Encoded" section** – it should match:

**eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV\_adQssw5c**