**Understanding JWT (JSON Web Token):**

**What is JWT?:**

* JWT stands for **JSON Web Token**
* It is an **Internet standard** (IETF RFC 7519) for creating JSON-based access tokens
* JWT is typically used to **pass the identity** of authenticated users between client and server or between services

**JWT Process Flow:**

**Steps:**

1. Client sends **username and password** to the server
2. Server **validates credentials**, creates a JWT token, and sends it back to the client
3. Client stores and **attaches the token** in the Authorization header of future requests
4. Server **validates the token** on each request before processing it

**Structure of a JSON Web Token:**

A JWT consists of **three parts** separated by dots (.):

1. **Header:**
   * Contains metadata about the type of token and the signing algorithm used (e.g., HS256)

{

"alg": "HS256",

"typ": "JWT"

}

1. **Payload:**
   * Contains the **claims** (user data). This can include sub (subject/user), name, role, iat (issued at), etc.

{

"loggedInAs": "admin",

"iat": 1422779638

}

1. **Signature:**
   * Created using the encoded header, payload, a secret key, and the algorithm specified in the header
   * This ensures the token is **not tampered**

HMAC\_SHA256(

secret,

base64urlEncoding(header) + '.' +

base64urlEncoding(payload)

)

**Final Token Format:**

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJsb2dnZWRJbkFzIjoiYWRtaW4iLCJpYXQiOjE0MjI3Nzk2Mzh9.OU1cvPMmynEIEDW29T-SaZotWusb7s8n\_R1N41hG7UY