Rahul Choudhary • 2nd

58K+ @LinkedIn | Tech Lead | Tech: NodeJS, NestJS, Angular, Express, MongoDB, MySQL, Rabb...

1d •

Follow

Things Every Developer Should Know: JSON Web Token (#JWT).

JWTs are one of the most widely used methods for API #authentication, providing a secure, stateless and scalable way to verify clients.

Here's a simple-to-understand breakdown of how it works, step by step:

1) Client authentication

The client (a user, app, or device) provides credentials (eg; username/password) to the #authentication server.

2) Server verification

The authentication #server checks the credentials against its #database or identity provider to confirm their validity.

3) JWT issuance

If authentication is successful, the server:

☑ Generates a JWT with claims (eg; user ID, roles, permissions).

☑ Signs the JWT using a secret key (HS256) or a private key (RS256).

4) Token delivery

The server sends the signed JWT back to the client in the response.

5) Secure storage

The client stores the JWT securely to prevent unauthorized access. HTTP-only cookies are the most secure and widely used method.

6) API requests with JWT

For each request to a protected API, the client includes the JWT in the Authorization header:

`Authorization: Bearer <JWT>`

7) Server validates the JWT

The API server verifies the JWT before granting access by checking:

☑ Signature – Confirms token integrity (not tampered with).

☑ Expiration – Ensures the token hasn't expired.

☑ Audience (aud claim) – Checks if the token is meant for this API.

☑ Issuer (iss claim) – Confirms the token was issued by a trusted authority.

If the JWT is valid, the server grants access to the requested resource. Otherwise, it rejects the request (401 Unauthorized).

8) Token expiration & refresh

Since JWTs expire for security reasons, the client needs a refresh token to get a new one:

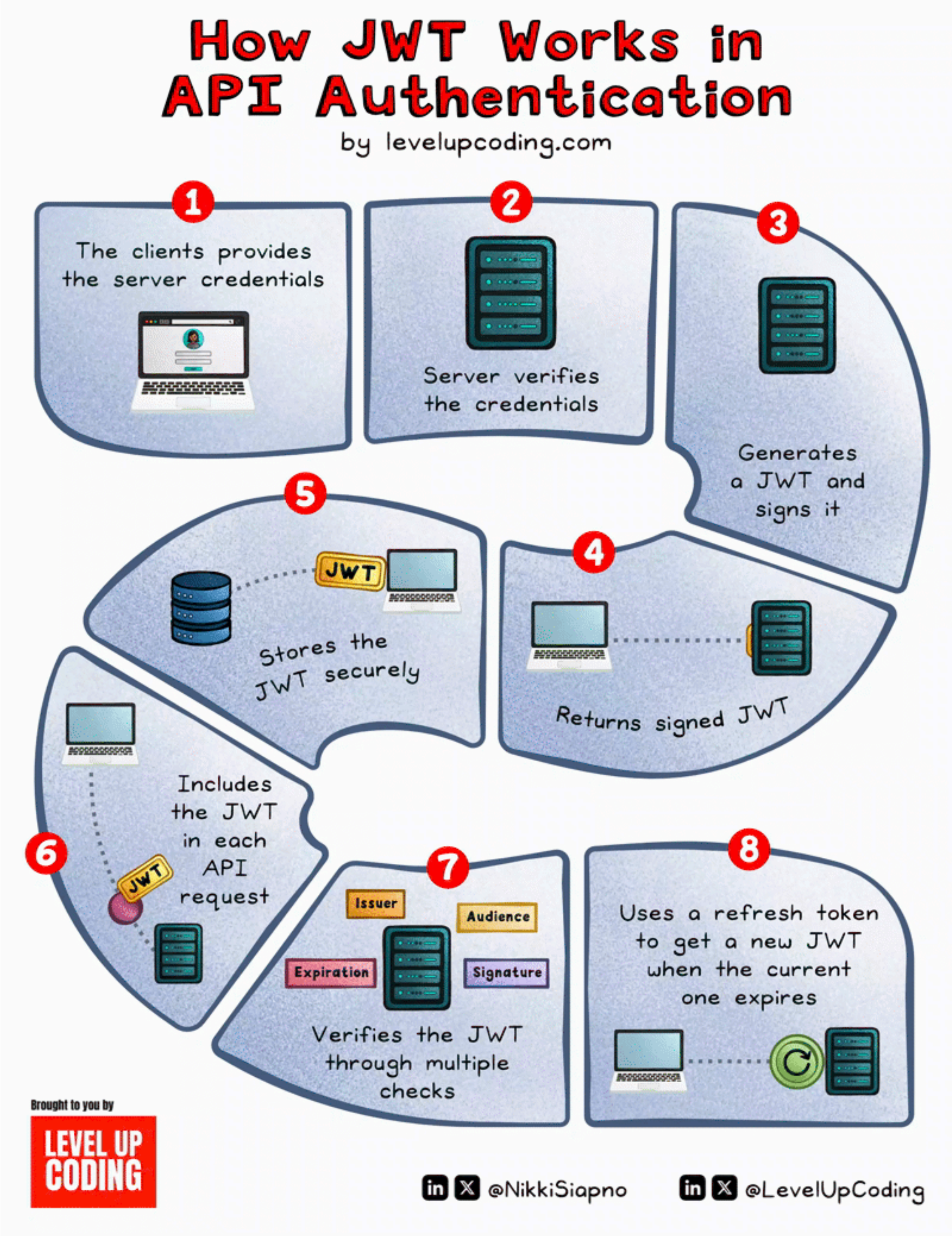
↳ Client sends refresh token to the server.

↳ Server verifies & issues a new JWT if the refresh token is valid.

↳ New JWT replaces the expired one, and the client continues making requests.

This workflow ensures secure, stateless, and efficient authentication for APIs while keeping performance and scalability in check.

learn free W3Schools.com JavaScript Mastery



How JWT Works in API Authentication

by levelupcoding.com

1 The clients provides the server credentials

2 Server verifies the credentials

3 Generates a JWT and signs it

4 Returns signed JWT

5 Stores the JWT securely

6 Includes the JWT in each API request

7 Verifies the JWT through multiple checks

8 Uses a refresh token to get a new JWT when the current one expires

Brought to you by LEVEL UP CODING

@NikkiSiapno @LevelUpCoding

355

19 comments 80 reposts