**1. JWT là gì?**

**JWT (JSON Web Token)** là một chuẩn mở (RFC 7519) dùng để truyền thông tin một cách an toàn giữa các bên dưới dạng đối tượng JSON. Thông tin này được mã hóa và ký số (hoặc mã hóa đối xứng) để đảm bảo tính xác thực và toàn vẹn của dữ liệu.

**Ưu điểm:**

* Không cần lưu session trên server: Tất cả thông tin được lưu trong token → giúp stateless.
* Gọn nhẹ: Dạng text (chuỗi), có thể truyền qua header HTTP, URL, cookie.
* Dễ dàng xác thực và phân quyền: Chứa thông tin user, role, expiration...

**Nhược điểm:**

* Không thể thu hồi token trừ khi có blacklist → nếu bị lộ token, hacker có thể dùng cho đến khi hết hạn.
* Kích thước lớn hơn sessionID truyền thống vì chứa nhiều metadata.
* Không tự mã hóa payload → ai cũng đọc được nội dung payload nếu không mã hóa.

**2. Cấu trúc của JWT**

Một chuỗi JWT gồm 3 phần, phân cách nhau bằng dấu chấm (.):

<Header>.<Payload>.<Signature>

Ví dụ JWT:

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.

eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkxpbmgiLCJpYXQiOjE1MTYyMzkwMjJ9.

SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV\_adQssw5c

**2.1 Header – chứa metadata của token:**

{

"alg": "HS256", // Thuật toán ký (HMAC SHA-256)

"typ": "JWT" // Loại token

}

→ Được Base64Url-encoded, ví dụ: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9

**2.2 Payload – chứa dữ liệu (claims):**

{

"sub": "1234567890", // ID của subject (người dùng)

"name": "Linh", // Dữ liệu tùy ý

"iat": 1516239022 // issued at: thời điểm phát hành token

}

→ Cũng được Base64Url-encoded. Payload có thể chứa:

* **Registered claims**: như iss (issuer), exp (expiration), sub, aud, iat...
* **Public claims**: dùng chung, tránh trùng tên.
* **Private claims**: do hệ thống định nghĩa.

**2.3 Signature – chữ ký số:**

Được tạo bằng:

HMACSHA256(

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

secret)

→ Đảm bảo token không bị chỉnh sửa. Nếu sửa, signature sẽ không khớp.

**3. Các trường hợp sử dụng phổ biến của JWT**

* **Xác thực người dùng (Authentication)**: Sau khi login thành công, server trả về JWT → client gửi JWT trong các request tiếp theo để xác minh.
* **Phân quyền truy cập (Authorization)**: JWT chứa vai trò (roles), quyền (permissions) để xác định người dùng có quyền truy cập tài nguyên nào.
* **Trao đổi thông tin an toàn giữa các hệ thống**: Ví dụ microservices, hệ thống bên thứ 3.
* **Lưu trạng thái client (stateless session)**: Thay vì lưu session trong server, JWT chứa dữ liệu trạng thái.

**Tóm lại:** JWT là một giải pháp hiệu quả cho các ứng dụng web hiện đại trong việc xác thực và phân quyền. Tuy nhiên, cần kết hợp thêm biện pháp bảo mật (https, giới hạn thời gian sống, lưu trữ an toàn...) để tránh rủi ro.