در دنیای روزمره امنیت وب، یکی از ابزارهای محبوب و قدرتمند برای احراز هویت و انتقال اطلاعات امن، **JWT (JSON Web Token)** است. در ادامه مفهوم JWT، مراحل کارکرد آن، نحوه راه‌اندازی در دات‌نت (با استفاده از JWT Bearer authentication) و نحوه استفاده از آن به همراه کوکی توضیح داده می‌شود.

۱. مفهوم JWT

**JWT** یک توکن استاندارد مبتنی بر JSON است که شامل سه بخش اصلی می‌شود:

* **Header:** شامل اطلاعات مربوط به الگوریتم امضا (مانند HMACSHA256، RSA و …) و نوع توکن (JWT) است.
* **Payload:** شامل کلید-مقدار (claims) است که اطلاعات کاربردی مانند شناسه کاربر، نقش‌ها (roles) یا سایر اطلاعات اضافی را به همراه می‌آورد.
* **Signature:** این بخش با امضای دیجیتال (با استفاده از یک کلید مخفی یا کلید عمومی/خصوصی در حالت RSA) ایجاد می‌شود تا از دستکاری توکن جلوگیری کند.

مثال یک JWT (ساختار بصورت Base64Url رمزگذاری شده):

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJJRCI6IjEyMyIsIlJvbGUiOiJBZG1pbiJ9.nL8P5j2s5Ka6aO7NfE85tS\_ZC63kTy\_ag9a5jQ9s\_Jw



هر قسمت را می‌توان به صورت جداگانه decode کرد و اطلاعات موجود در آن را دید، ولی بدون کلید، امضای دیجیتال (Signature) قابل جعل نیست.

۲. کاربرد JWT در دات‌نت

الف) احراز هویت API

در برنامه‌های وب و API‌های RESTful، معمولاً پس از اعتبارسنجی کاربر (مثلاً در اکشن Login)، یک JWT برای او تولید می‌شود. سپس این توکن را در درخواست‌های بعدی (معمولاً در header با کلید Authorization و مقدار Bearer) به سرور می‌فرستند تا هویت کاربر بررسی شود.

ب) استفاده از کوکی برای JWT

در برخی سناریوها (به‌ویژه با برنامه‌های SPA یا وب‌اپلیکیشن‌های سنتی)، می‌توانید به جای نگهداری توکن در حافظه سمت کاربر (مثلاً localStorage) آن را در **کوکی‌های HTTP-only** ذخیره کنید.  
مزیت اصلی استفاده از کوکی HTTP-only این است که از دسترسی جاوااسکریپت به توکن جلوگیری می‌شود (حفاظت در برابر حملات XSS) و کوکی می‌تواند به صورت خودکار با هر درخواست به سرور ارسال شود.