قلعه CastleAPI و نگهبان‌ها

فرض کن پروژه‌ی من یه قلعه است به اسم **CastleAPI** . این قلعه اتاق‌های زیادی داره که هر کدوم اطلاعات مهمی نگه می‌دارن:
مثل «لیست کاربران»، «پست‌ها»، «نوتیفیکیشن‌ها» و...
من باید جوری طراحی می‌کردم که هر کسی نتونه سرشو بندازه پایین و وارد بشه!

برای همین یه نگهبان اصلی دم در گذاشتم به اسم **Auth Server** :

کاربری برای مثال نقی فرم ثبت‌نام رو پر می‌کنه یا لاگین می‌کنه. حالا نگهبان بهش میگه:

"باشه، توی دیتابیس ما شناخته شدی. بیا این دو تا چیز رو بگیر و برو داخل قلعه".

حالا اون دوتا چیز چی ان؟

۱) **Access Token** : بلیت ورود محدود

اولین چیزی که به نقی داده میشه یه **Access Token** هستش.

Access Token یک توکن رمزنگاری‌شده‌ست که معمولاً از نوع JWT هست که اطلاعات زیر رو توش داره:

- کاربر کیه؟(subject: user_id)
- از طرف کی صادر شده؟(issuer)
- چه مجوزهایی داره؟(scope)
- تا کی اعتبار داره؟(exp)

این توکن باید توی هر درخواست همراه نقی باشه تا نگهبان‌های داخل قلعه بررسی کنن ببینن اجازه داره یا نه.

نکته امنیتی : چون اگر این توکن دزدیده بشه، مهاجم می‌تونه مثل نقی رفتار کنه، باید **عمرش کوتاه** باشه. مثلاً فقط ۱۵ دقیقه.

۲) **Refresh Token** : کلید تمدید مجوز

حالا اگه Access Token منقضی بشه چی؟ کاربر باید دوباره لاگین کنه؟

نه ! این‌جاست که **Refresh Token** وارد میشه.

نگهبان دوم یه **کلید مخفی بلندمدت** به نقی میده. این کلید فقط مخصوص خودشه و می‌تونه هر وقت Access Token منقضی شد، باهاش یه توکن جدید بگیره.

مقایسه ی این دو توکن :

مورد	Access Token	Refresh Token
کاربرد	دسترسی به منابع	گرفتن Access جدید
عمر	کوتاه	بلند (مثلاً ۷ روز)
محل نگه‌داری	حافظه یا Secure Cookie	فقط HttpOnly Cookie
امنیت	در معرض خطر بیشتر	بسیار حساس

من تو پروژه ام Refresh Token رو در یک **HttpOnly cookie** ذخیره کردم، یعنی از جاوااسکریپت هم قابل دسترسی نیست برای جلو گیری از حملات injection(XSS)

یک مدل دیگه هم توکن داریم که زیاد استفاده میشه ! بهش میگیم ID token

یه‌جاهایی فقط لازمه بدونم طرف کیه. نه اینکه به API خاصی دسترسی داشته باشه، فقط مثلاً اسمش چیه، ایمیلش چیه.

برای اینکار **ID Token** استفاده می‌کنم که اطلاعات هویتی توش هست، ولی اصلاً برای "دسترسی به منابع" نیست.

این بیشتر در پروتکل **OpenID Connect** کاربرد داره، مثلاً وقتی از گوگل لاگین می‌کنی و فقط مشخصاتت برمی‌گرده.

ساختار فنی توکن ها (JWT)

JWT این شکلیه:

```
eyJhbGciOi... (header).eyJzdWIiOi... (payload).SflKx... (signature)
```

بخش	توضیح
Header	الگوریتم رمزنگاری و نوع توکن
Payload	اطلاعات درباره کاربر، زمان انقضا، مجوزها
Signature	امضای دیجیتال با کلید سرور برای جلوگیری از تقلب