Autenticazione con JWT

In un flusso tipico di autenticazione con JWT (JSON Web Token), il processo di convalida avviene nel seguente modo:

1. **Login dell'utente**: L'utente invia le proprie credenziali al backend tramite una richiesta POST sicura.
2. **Verifica delle credenziali**: Il backend verifica le credenziali e, se sono corrette, genera un JWT.
3. **Invio del JWT al frontend**: Il backend invia il JWT al frontend come parte della risposta alla richiesta di login.
4. **Storage del JWT**: Il frontend salva il JWT, solitamente in `localStorage` o `sessionStorage`, per mantenerlo attraverso le sessioni di navigazione.
5. **Utilizzo del JWT per le richieste successive**: Per ogni richiesta successiva che richiede autenticazione, il frontend invia il JWT al backend nell'header `Authorization` della richiesta HTTP, tipicamente come un Bearer Token.
6. **Convalida del JWT dal backend**: Il backend, per ogni richiesta ricevuta, estrae il JWT dall'header `Authorization`, verifica la firma del token per assicurarsi che non sia stato manomesso, controlla la validità del token (scadenza, emittente, audience, ecc.), e se il token è valido, permette all'utente di accedere alla risorsa richiesta.
7. **Risposta del backend**: Se il token è valido, il backend procede con la richiesta e restituisce la risposta desiderata. Se il token non è valido (ad esempio, è scaduto o non è stato firmato correttamente), il backend restituirà un errore, spesso con uno stato HTTP 401 (Unauthorized) o 403 (Forbidden).
8. **Rinnovo del JWT**: Se il token è scaduto, il frontend può avere una logica per rinnovarlo. Questo può essere fatto automaticamente utilizzando un token di aggiornamento che il backend fornisce insieme al JWT o richiedendo all'utente di riautenticarsi.

In pratica, la convalida del JWT si svolge interamente lato server. La verifica lato client è limitata a controlli base come la scadenza del token o per estrarre informazioni dal payload decodificato (ad esempio, il nome utente o i ruoli). Tuttavia, per motivi di sicurezza, non dovresti mai fidarti completamente della convalida lato client, poiché il codice JavaScript può essere manomesso da un attaccante.