Certified Tech Developer
The Ultimate Degree

Front End II

O que é um JWT?

Em poucas palavras, é um padrão de criptografia aberto usado para transmitir informações com segurança entre duas partes. A forma como essa informação é transmitida é através de um objeto JSON (com certeza você se lembra que vimos como o formato JSON era usado para compartilhar informações entre cliente e servidor). Sua peculiaridade é que as informações transmitidas podem ser verificadas, pois o JWT é assinado digitalmente.

Agora, como é composto um JWT? Basicamente, a estrutura mais simples de um JWT consiste em três partes: header, payload e signature.

Exemplo:

`xxxxx.yyyyy.zzzzz`

Vamos dar uma olhada rápida em cada uma dessas partes:

- **Header**: esta parte contém as informações sobre o tipo de token (JWT) e o algoritmo de criptografia usado. Sua estrutura é a seguinte:

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

- **Payload**: é a parte mais relevante do ponto de vista da autorização, pois aqui você encontra as informações do usuário, que podem incluir, por exemplo, a função que esse usuário tem dentro do aplicativo:

```
{
  "sub": "1234567890",
  "name": "John Doe",
  "admin": true
}
```

- **Signature**: é a parte que garante a autenticidade da informação incluída no JWT, permitindo a sua verificação.

Até agora, vimos como um JWT é composto. Agora, é hora de nos perguntarmos como podemos usá-lo no processo de autorização que vimos anteriormente.

De modo geral, quando uma pessoa faz login em um determinado aplicativo, o servidor verifica as credenciais inseridas (nome de usuário e senha). Se estiverem corretos, o servidor **autentica** o usuário dentro do aplicativo e envia um JWT como resposta à solicitação. Isso é armazenado no lado do cliente e enviado ao servidor a cada nova solicitação feita para acessar um determinado serviço dentro do aplicativo. Como vimos acima, o token contém as informações do usuário (por exemplo, sua função), o servidor pode acessar essas informações ao receber a solicitação, e assim, validar se o usuário está **autorizado** a realizá-la. Se for assim, o servidor processará a solicitação e enviará a resposta correspondente. Caso contrário, será retornado um erro indicando que a pessoa não está autorizada.

Em resumo, o JWT é uma ferramenta muito útil para comunicação entre cliente e servidor, pois permite compartilhar informações do usuário de maneira segura e eficiente, além de acessar essas informações para validar as funções e permissões de cada pessoa que acessa o nosso aplicativo.

Caso pretenda aprofundar-se neste tema, deixamos o link para a documentação oficial do JWT: https://jwt.io/#debugger-io.