JWT简介

什么是JWT

JSON Web Token（JWT）是一个开放的行业标准（RFC 7519），它定义了一种简介的、自包含的协议格式，用于在通信双方传递json对象，传递的信息经过数字签名可以被验证和信任。JWT可以使用HMAC算法或使用RSA的公钥/私钥对来签名，防止被篡改。

官网： https://jwt.io/

标准： https://tools.ietf.org/html/rfc7519

JWT令牌的优点：

1. jwt基于json，非常方便解析。
2. 可以在令牌中自定义丰富的内容，易扩展。
3. 通过非对称加密算法及数字签名技术，JWT防止篡改，安全性高。
4. 资源服务使用JWT可不依赖认证服务即可完成授权。

缺点：

1. JWT令牌较长，占存储空间比较大。

JWT组成

一个JWT实际上就是一个字符串，它由三部分组成，头部、载荷与签名。

头部(Header)

头部用于描述关于该JWT的最基本的信息，例如其类型（即JWT）以及签名所用的算法（如HMAC SHA256或RSA）等。这也可以被表示成一个JSON对象。

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

- `typ`：是类型。
- `alg`：签名的算法，这里使用的算法是HS256算法

我们对头部的json字符串进行BASE64编码（网上有很多在线编码的网站），编码后的字符串如下：

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9
```

`Base64`是一种基于64个可打印字符来表示二进制数据的表示方法。由于2的6次方等于64，所以每6个比特为一个单元，对应某个可打印字符。三个字节有24个比特，对应于4个Base64单元，即3个字节需要用4个可打印字符来表示。JDK 中提供了非常方便的`BASE64Encoder`和`BASE64Decoder`，用它们可以非常方便的完成基于 BASE64 的编码和解码。

负载(Payload)

第二部分是负载，就是存放有效信息的地方。这个名字像是特指飞机上承载的货品，这些有效信息包含三个部分：

- 标准中注册的声明（建议但不强制使用）

```
iss: jwt签发者
sub: jwt所面向的用户
aud: 接收jwt的一方
exp: jwt的过期时间，这个过期时间必须要大于签发时间
nbf: 定义在什么时间之前，该jwt都是不可用的.
iat: jwt的签发时间
jti: jwt的唯一身份标识，主要用来作为一次性token,从而回避重放攻击。
```

- 公共的声明

公共的声明可以添加任何的信息，一般添加用户的相关信息或其他业务需要的必要信息.但不建议添加敏感信息，因为该部分在客户端可解密.

- 私有的声明

私有声明是提供者和消费者所共同定义的声明，一般不建议存放敏感信息，因为base64是对称解密的，意味着该部分信息可以归类为明文信息。

这个指的就是自定义的claim。比如下面那个举例中的name都属于自定的claim。这些claim跟JWT标准规定的claim区别在于：JWT规定的claim，JWT的接收方在拿到JWT之后，都知道怎么对这些标准的claim进行验证(还不知道是否能够验证)；而private claims不会验证，除非明确告诉接收方要对这些claim进行验证以及规则才行。

```
{
  "sub": "1234567890",
  "name": "John Doe",
  "iat": 1516239022
}
```

其中`sub`是标准的声明，`name`是自定义的声明（公共的或私有的）

然后将其进行base64编码，得到Jwt的第二部分：

```
eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkphbWVzIiwiYWRtaW4iOnRydWV9
```

提示：声明中不要放一些敏感信息。

签证、签名（signature）

jwt的第三部分是一个签证信息，这个签证信息由三部分组成：

1. header (base64后的)
2. payload (base64后的)
3. secret（盐，一定要保密）

这个部分需要base64加密后的header和base64加密后的payload使用.连接组成的字符串，然后通过header中声明的加密方式进行加盐secret组合加密，然后就构成了jwt的第三部分：

```
8HI-LodOncfVDnbKIPJJqLH998duF9DSDGkx3gRPNVI
```

将这三部分用.连接成一个完整的字符串,构成了最终的jwt:

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.8HI-LodOncfVDnbKIPJJqLH998duF9DSDGkx3gRPNVI
```

注意：`secret`是保存在服务器端的，`jwt`的签发生成也是在服务器端的，`secret`就是用来进行`jwt`的签发和`jwt`的验证，所以，它就是你服务端的私钥，在任何场景都不应该流露出去。一旦客户端得知这个`secret`,那就意味着客户端是可以自我签发`jwt`了。

JJWT简介

什么是JJWT

JJWT是一个提供端到端的JWT创建和验证的Java库。永远免费和开源(Apache License，版本2.0)，JJW很容易使用和理解。它被设计成一个以建筑为中心的流畅界面，隐藏了它的大部分复杂性。

规范官网：https://jwt.io/