What's the difference between JWTs and Bearer Token?

Asked 4 years, 2 months ago Active 3 years, 10 months ago Viewed 67k times

125

39

I'm learning something about Authorization like Basic, Digest, OAuth2.0, JWTs, and Bearer Token.

Now I have a question.

You know the JWTs is being used as an Access_Token in the OAuth2.0 standard. JWTs appears at RFC 7519, and Bearer Token is at RFC 6750 .

For example, the Bearer:

```
Authorization: Bearer <token>
```

I used to send token to server by AJAX or add token to the query string of the url. I know that a token can also be sent by adding it to a request header. Does that mean that token should be added to Authorization Bearer header?

Could you please tell me the relationship between JWTs and Bearer Token? Thanks a lot.

oauth token jwt

edited Mar 5 '17 at 20:27
JuJoDi
12.6k 20 73 120

asked Nov 2 '16 at 8:38
laoqiren
2,085 4 13 25

3 Answers

Active Oldest Votes

100

JWT is an encoding standard for tokens that contains a JSON data payload that can be signed and encrypted.

JWT can be used for many things, among those are bearer tokens, i.e. a piece of information that you can present to some service that by virtue of you having it (you being the "bearer") grants you access to something.

Bearer tokens can be included in an HTTP request in different ways, one of them (probably the preferred one) being the Authorization header. But you could also put it into a request parameter, a cookie or the request body. That is mostly between you and the server you are trying to access.

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

237k 91 468 618

Then how to parse the token in Authorization header, there is a Bearer , I have to use String.slice(), is there any middleware to parse it? – laoqiren Nov 2 '16 at 10:12

I'm using Node.js – laoqiren Nov 2 '16 at 10:14

Use the `auth-header` package if you want a minimal parser – Demurgos Jun 22 '17 at 10:04

1 Or just use String.slice() :) – Paul Siersma Jun 7 '18 at 14:26

158

Short answer

JWTs are a convenient way to *encode* and *verify* **claims**.

A Bearer token is just string, potentially arbitrary, that is used for authorization.

Context (story time)

A few years ago, before the JWT revolution, a `<token>` was just a string with no intrinsic meaning, e.g. 2pWS6RQmdZpE0TQ93X. That token was then looked-up in a database, which held the **claims** for that token. The downside of this approach is that DB access (or a cache) is required everytime the token is used.

JWTs *encode* and *verify* (via signing) their own **claims**. This allows folks to issue short-lived JWTs that are stateless (read: self-contained, don't depend on anybody else). They do not need to hit the DB. This reduces DB load and simplifies application architecture because only the service that issues the JWTs needs to worry about hitting the DB/persistence layer (the `refresh_token` you've probably come across).

answered Nov 2 '16 at 8:52

rmharrison

2,752 1 14 30

Thanks, and how about Mac in Authorization, are the Mac and Bearer the same? – laoqiren Nov 2 '16 at 10:13

Best answered elsewhere, e.g.: dzone.com/articles/oauth-20-bearer-token-profile – rmharrison Nov 2 '16 at 11:57

-5

JWTs work with two types of token, Parameter Token: Access token pass as parameter. Bearer Token: it's pass in header with 'Bearer'.

Please read the following question also:

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

edited May 23 '17 at 12:17

Community ♦
1 1

answered Nov 2 '16 at 8:50

Ehtasham Nasir
143 6

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.