# JWT (Json web token) Vs Custom Token

Asked 5 years, 5 months ago Active 4 years, 9 months ago Viewed 9k times

24

5

I was looking through the questions but I did not find anything which could solve my doubt. I found extensive information about JWT, but not much when comparing the advantages JWT could offer over generating a custom token to authentication requests against REST services.

What is the advantage to use a JWT (Json Web Token) over generating a custom generating token ? To generating the custom token I could use some hashing strategy or some unique random number generator.

If I generate a custom token, Could I have any security concerns ? Would you recommend to use any other authentication mecanism ?

Thanks!

security jwt rest-security

asked Jul 30 '15 at 17:54

Deibys

491 1 7 18

Hi Deibys, I've got the same question, have you found any convincing answer, and what approach did you finally applied? thanks – Adnane.T Nov 17 '15 at 19:57

Same here, given JWT should not be used in sessions, why should I not use my own token implementation. – Anuj Kumar Sep 2 '18 at 16:34

## 2 Answers

Active Oldest Votes

13

JWT tokens contain claims, which are statements about the subject (for example the logged in user). These statements can be things like name, email, roles etc. JWT tokens are digitally signed and not vulnerable to CSRF attacks.

These two characteristics make sure that the service receiving the token does not need to go back to the issuing authentication server to verify the validity of the token or get information about the subject.

This increases the ability of a system using JWT tokens to scale in a significant way. JWT tokens do require a secure transportation channel (HTTPS).

Tokens holding a session id on the other hand do need to contact the authentication server to validate the token (usually database lookup) and retrieve information on the subject (another database lookup).

Validation of HMAC tokens requires the knowledge of the secret key used to generate the token. Typically the receiving service (your API) will need to contact the authentication server as that server is where the secret is being kept.

HMAC tokens and session ids are typically stored in cookies. Cookies cannot be used for cross-domain service calls and need to be protected against CSRF attacks.

edited Mar 14 '16 at 22:45

answered Jul 31 '15 at 2:19
MvdD
17k 5 49 80

6 JWT itself does not provide any measures against CSRF nor XSS attacks. Signing is a measure against tampering the token. Token can still be stolen even if it is signed. Stormpath has good article describing where you should store the JWT tokens and how to protect against CSFR and XSS. stormpath.com/blog/… – Mika Tuupola Jul 31 '15 at 20:27

6 JWT tokens themselves do not protect against CSRF, but they are typically used in a bearer authentication scheme. The bearer authenication scheme is not vulnerable to CSRF. I read the Stormpath article, but do not agree with their recommendations. Storing JWT tokens in a cookie makes it more difficult for non-browser clients to consume your API and makes it hard to have your APIs in different domains. – MvdD Jul 31 '15 at 21:41

5 To be honest, I still don´t see the advantage of using JWT token over a custom generated token which could be hard to hack, if they are a hash and I have a datastorage in the server to validate if they are valid or not – Deibys Aug 3 '15 at 14:42

1 If you need to invalidate tokens, JWT tokens may not be the best option. But typically the lifetime of the token is short enough not to have to worry about invalidation. I would not store JWT tokens in a cookie. The bearer authentication scheme is usually used. – MvdD Aug 3 '15 at 15:28

2 Also to add this, the JWT tokens are not encrypted, so storing sensitive info there is problematic, as it resides in a client machine, and attacker that grabs the token can then discover more info about the user. you'd say it's not much of an issue as he can use that token to access the service and maybe receive the same info - but i'd say a session token is better encrypted than just signed – ArielB Sep 28 '17 at 12:55

1

From Django REST framework documentation,

> JSON Web Token is a fairly new standard which can be used for token-based authentication. Unlike the built-in TokenAuthentication scheme, JWT Authentication doesn't need to use a database to validate a token.

answered Dec 11 '15 at 8:55
Chemical Programmer