HASURA (https://hasura.io/)   Star   20.2k (https://github.com/hasura/graphql-engine/)

All (/blog/)   GraphQL (/blog/tagged/graphql/)

Serverless (/blog/tagged/serverless/)

Remote Joins (/blog/tagged/remote-joins/)

Authentication (/blog/tagged/authentication/)   React (/blog/tagged/react/)

🔍 Search

The Ultimate Guide to handling JWTs on frontend clients (GraphQL)

09 SEPTEMBER, 2019 | 15 MIN READ

JWTs (JSON Web Token, pronounced 'jot') are becoming a popular way of handling auth. This post aims to demystify what a JWT is, discuss its pros/cons and cover best practices in implementing JWT on the client-side, keeping security in mind. Although, we've worked on the examples with a GraphQL clients, but the concepts apply to any frontend client.

- Introduction to JWT (#intro)

  - Security considerations (#jwt_security)

  - JWT structure (#jwt_structure)

  - Why JWT instead of session tokens (#jwt_vs_session)

- Login (#login)

  - Persisting token or not? (#jwt_persist)

  - Token usage on the client (#login_usage)

(https://getpocket.com/save?url=https://hasura.io/blog/best-practices-of-using-jwt-with-graphql/)

(https://wa.me/?text=https://hasura.io/blog/best-practices-of-using-jwt-with-graphql/)

(https://twitter.com/intent/tweet?&text=The Ultimate Guide to handling JWTs on frontend clients (GraphQL)&url=https://hasura.io/blog/best-practices-of-using-jwt-with-graphql/)

(http://www.linkedin.com/shareArticle?mini=true&url=https://hasura.io/blog/best-practices-of-using-jwt-with-graphql/&title=The Ultimate Guide to handling JWTs on frontend clients (GraphQL)&summary=The Ultimate Guide to handling JWTs on frontend clients (GraphQL)&source=https://hasura.io/blog/best-practices-of-using-jwt-with-graphql/)