JWT Authentication

Introduction

By Sergey Kargopolov

- POST /users
- GET /users/<userId> Require authentication
- PUT /users/<userId> Require Authentication
- DELETE /users/<userId> Require Authentication
- POST /login

Users Microservice

Users Db
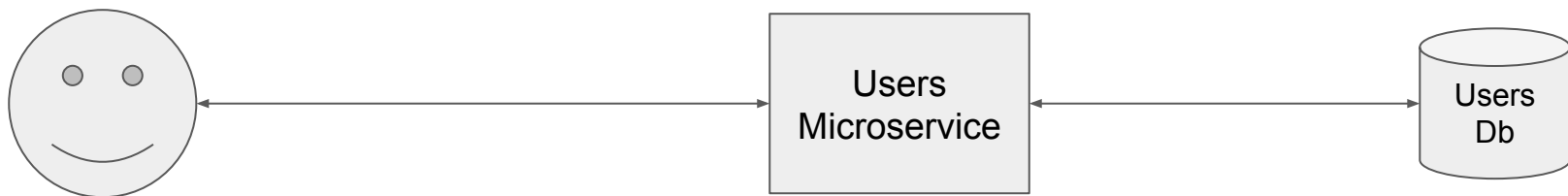
User authentication is the process of confirming a user's identity by checking their credentials, like a username and password.

By Sergey Kargopolov

POST **/login**

```
{
    "email":"test1@com",
    "password":"123456789"
}
```

- POST /users
- GET /users/<userId> `Require  authentication`
- PUT /users/<userId> `Require Authentication`
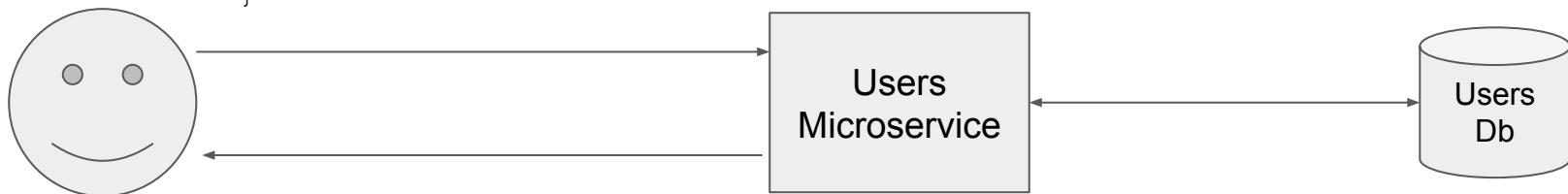- DELETE /users/<userId> `Require Authentication`
- POST /login

Users
Microservice

Users
Db

JSON Web Token(JWT)

eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJlNTI3YmRhZS0
2ZTJkLTRjMDYtYjQ3ZC1iNzE3OWVkNWI0OWEiLCJ
pYXQiOjE3Mjk5NjY2ODgsImV4cCI6MTcyOTk3MDI4
OH0.Sn6L2JV4X7ZC3bE-Qkw2qRwd12ruiRiX_P4-Yd
ErFDIIuYJdPIZeUfVepniDhQatZBLX_wfbl6W9OIuRA
Yechg

JSON Web Token(JWT)

eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2IkIiA6ICItNUIsX2I0cUktdWFvaEl3d244UHY3WEM2UEktU3BNbmZCRnIJZUx6QTJNIn0.eyJleHAiOjE1OTM0NTcxMTAsImlhdCI6MTU5MzQ1NjgxMCwiYXV0aF90aW1lIjoxNTkzNDU2NjE2LCJqdGkiOiJiMDkyOGEzNy01M2Y0LTQxZDQtODZkMy0wY2E2OTI2Njg4OTkiLCJpc3MiOiJodHRwOi8vbG9jYWxob3N0OjgwODAvYXV0aC9yZWFsbXMvYXBwc2RldmVsb3BlcmJsb2ciLCJzdWIiOiIxZGRlM2ZjMy1jNmRiLTQ5ZmItOWIzZC03OTY0YzVjMDY4N2EiLCJ0eXAiOiJCZWFyZXIiLCJhenAiOiJwaG90by1hcHAtY29kZS1mbG93LXJlc3QtY2xpZW50Iiwic2Vzc2lvbl9zdGF0ZSI6Ijc5OWFhOTJiLWVhNjctNDFiNy1hNzE1LWVkMGExNTI0MTY0NCIsImFjciI6IjAiLCJzY29wZSI6Im9wZW5pZCBwcm9maWxlIiwibmFtZSI6IkthcmdvcG9sb3YiLCJwcmVmZXJyZWRfdXNlcm5hbWUiOiJzZXJnXXkiLCJmYW1pbHlfbmFtZSI6IkthcmdvcG9sb3YifQ.FQQihDLOGU7UOeGNKzfZ9kqgRNqxpc2GLV1kB2xWbvk1fEkTX2m9y-jtEsXukjx0Ew0ZpRC9jR1fZaDfIWomzz7cYBsY5r6rl0SadxNJU6c7gUl4XTzaFhOpIyPl4DGff9DaIW1SAPV0RdskKKjmCoboj_CS1AfsVF2JFbujksmsnkTxrJ56oKom1UjrpQEIV_81bQRQ6jQI54_EdIcaqc6TvSTNosW6j7I_LSPqcLZtSVrQaa7pEDeeNgPdNY0mMYmNgGd1YcDsers4g3IihySikIALcPma3KZX-DPQ7RTbWXbrPNDwbjXd4Anft0YrMRHQfJYBmOqHeGkA287gQQ

What is JSON Web Token(JWT)

- An open standard (RFC 7519) for securely transmitting information between parties as a JSON object.
- A compact, self-contained token consisting of three parts: Header, Payload, and Signature

By Sergey Kargopolov

JSON Web Token(JWT)

eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6IClCltNUIsX2I0cUktdWFvaEl3d244UHY3WEM2UEktU3BNbmZCRnlJZUx6QTJNIn0.eyJleHAiOjE1OTM0NTcxMTAsImlhdCI6MTU5MzQ1NjgxMCwiYXV0aF90aW1lIjoxNTkzNDU2NjE2LCJqdGkiOiJiMDkyO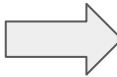GEzNy01M2Y0LTQxZDQtODZkMy0wY2E2OTI2Njk4ODkiLCJpc3MiOiJodHRwOi8vbG9jYWxob3N0OjgwODAvYXV0aC9yZWFsbXMvYXBwc2Rldmlsb3BlcmJsb2ciLCJzdWIiOiIxZGRlM2ZjMy1jNmRiLTQ5ZmItOWlzZC03OTY0YzVjMDY4N2EiLCJ0eXAiOiJCZWFyZXIiLCJhenAiOiJwaG90by1hcHAtY29kZS1mbG93LXJlc3QtY2xpZW50Iiwic2Vzc2lvbl9zdGF0ZSI6Ijc5OWFhOTJiLWVhYjctNDFiNy1hNzE1LWVkMGExNTI0MTY0NCIsImFjciI6IjAiLCJzY29wZSI6Im9wZW5pZCBwcm9maWxlIiwibmFtZSI6Ikthcmdvb9sb3YiLCJwcmVmZXJyZWRfdXNlcm5hbWUiOiJzZXJnZXkiLCJmYW1pbHlfbmFtZSI6Ikthcmdvb9sb3YifQ.FQQihDLOGU7UOeGNKzfZ9kqgRNqxpc2GLV1kB2xWbvk1fEkTX2m9y-jtEsXukjx0Ew0ZpRC9jR1fZaDfIWomzz7cYBsY5r6rl0SadxNJU6c7gUl4XTzaFhOpIyPl4DGff9DaIW1SAPV0RdskKKjmCoboj_CS1AfsVF2JFbujksmsnkTxrJ56oKom1UjrpQElV_81bQRQ6jQl54_Edlcaqc6TvSTNosW6j7l_LSPqcLZtSVrQaa7pEDeeNgPdNY0mMYmNgGd1YcDsers4g3lihySikIALcPma3KZX-DPQ7RTbWXbrPNDwbjXd4Anft0YrMRHQfJYBmOqHeGkA287gQQ
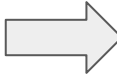
jwt.io

Header:
```json
{
  "alg": "HS512"
}
```

Payload:
```json
{
  "sub": "f3d60c7a-f1a2-4476-b24d-6f549a436047",
  "iat": 1730128410,
  "exp": 1730132010,
  "roles": ["ROLE_USER", "ROLE_ADMIN"],
  "authorities": ["READ_PROFILE", "EDIT_PROFILE", "DELETE_USER"]
}
```

By Sergey Kargopolov

Why we need JSON Web Token(JWT)

- Verify if the source is trusted,
- Verify user identity,
- Check user permissions.

By Sergey Kargopolov

By Sergey Kargopolov