Authentication

and

Authorization

Week 6, Day 1

Agenda for Today

- Authentication

  - Authorization

  - Local Auth

  - 0Auth

  - Auth0

- JSON Web Token (jwt)

- Express Middleware

Authentication

- Proving a user's identity
- Are you who you say you are?

Authentication

Authorization

- After a user has been authenticated

- What are they allowed to see and do?

- Asking for their own data vs asking for another person's data
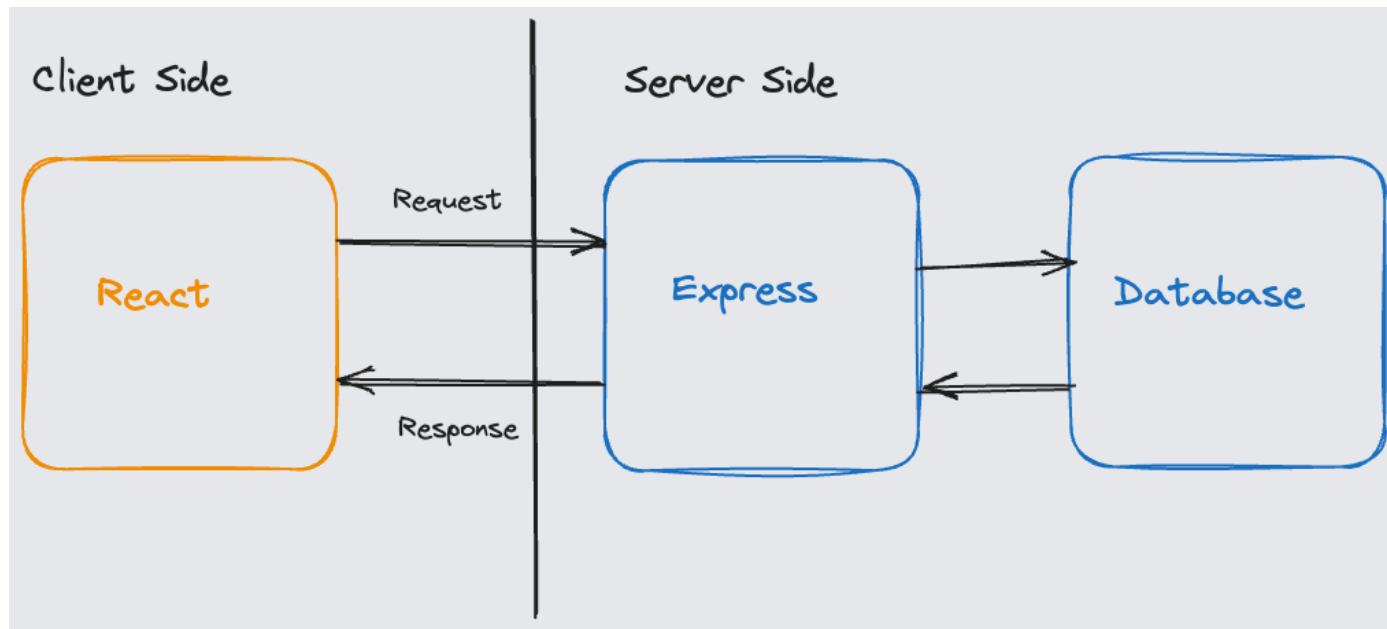
- Role based, users vs admins

Authentication

Local Auth

- Managing users login data ourselves on our own servers

- (e.g. korihi client)
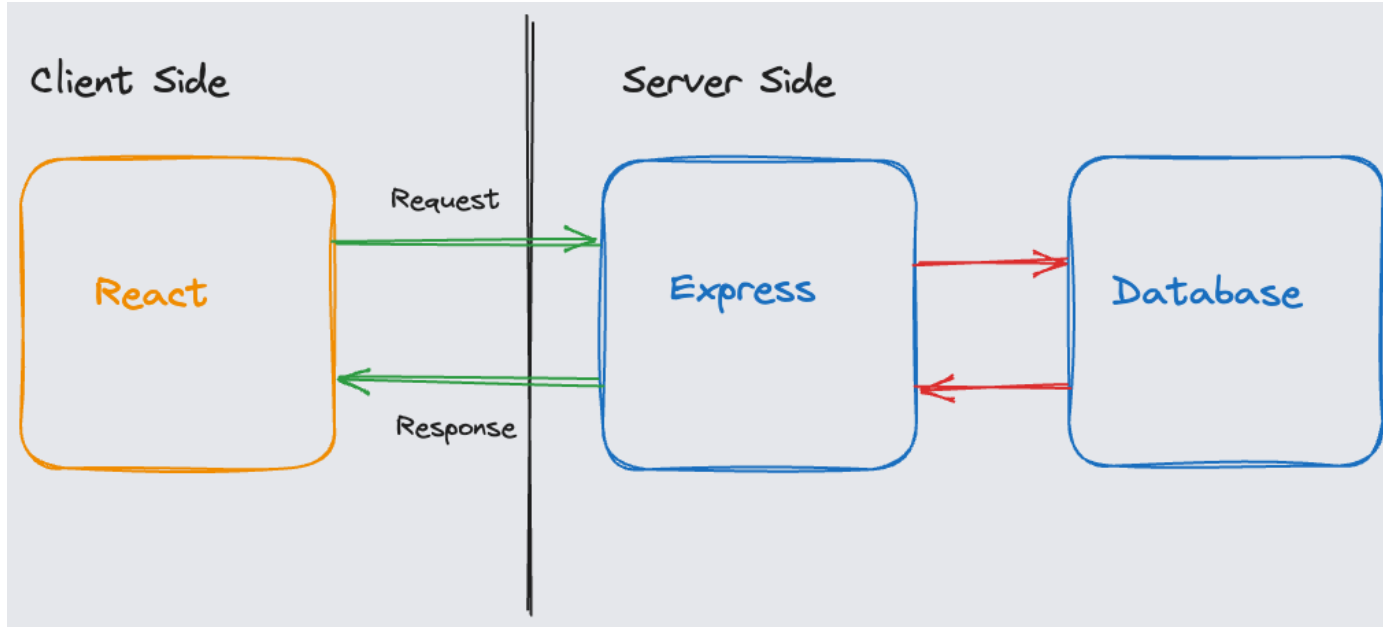
- Independent

- Private

- High Risk
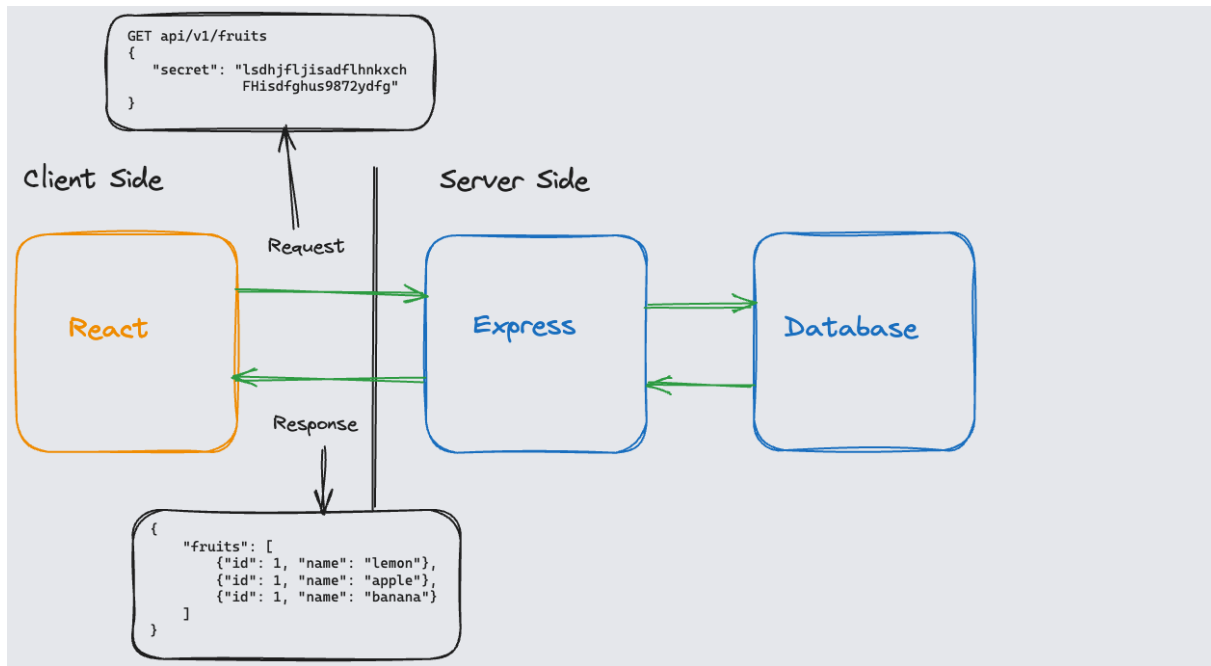
Authentication

Local Auth

Authentication

Local Auth

Authentication

Local Auth

Authentication

0Auth - The Protocol

- A Protocol that allows us to login to one account for another website

- e.g. using your gmail account to sign into other websites
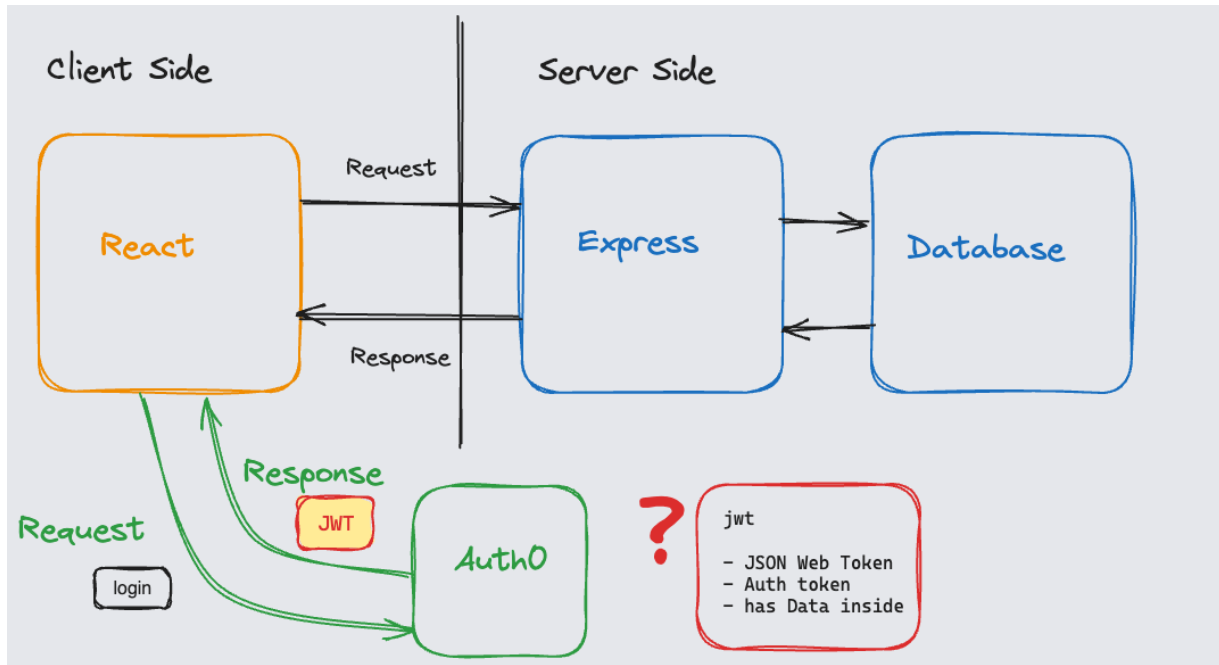
Authentication

Auth0 - The Platform

- A Service that manages user's login data for us
- Low Risk
- Dependent on the platform provider
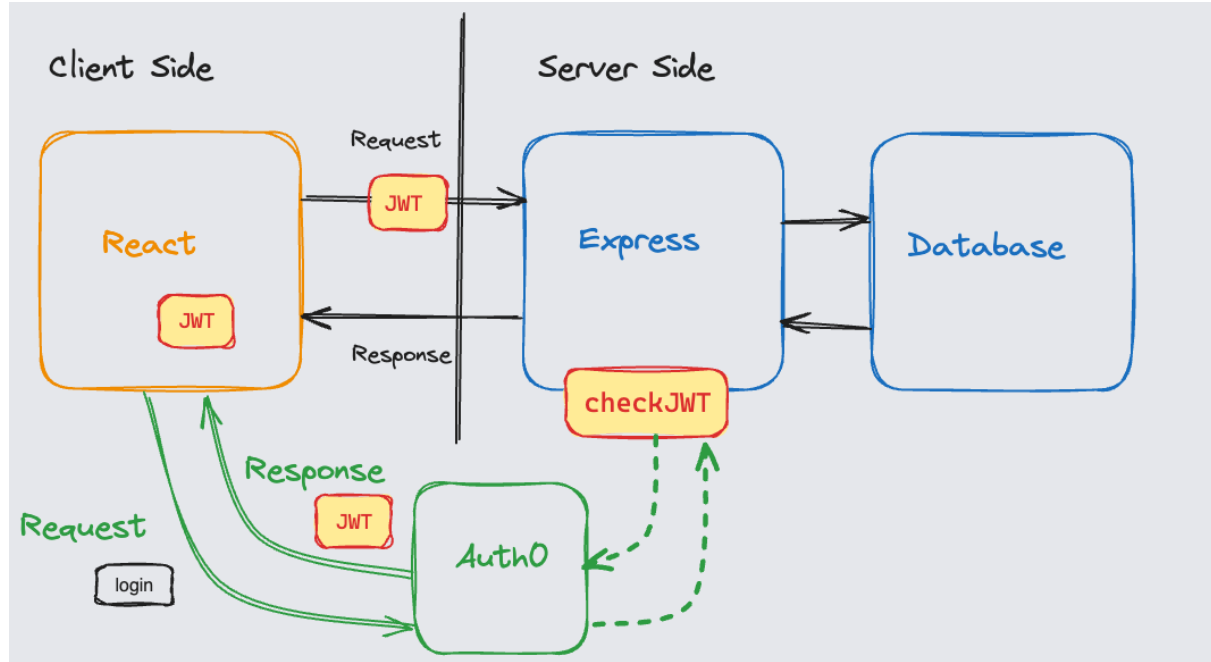- Less Private

Authentication

Auth0 - The Platform `Demo JWT`

Authentication

Auth0 - The Platform

Express Middleware

```
1   // GET /api/v1/fruits — No Middleware — PUBLIC
2   router.get('/', async (req, res) => {
3     res.send("Everybody can see this")
4   })
5
6   // POST /api/v1/fruits — with middleWare — ACCESS REQUIRED
7   router.post('/', checkJwt, async (req, res) => {
8     res.send("Congrats, youre authorized!")
9   })
10
11  // Example of a middleWare function
12
13  function checkJWT (req: Request, res: Response, next: NextFunction) {
14   if (spookyAuthStuff()) {
15    next()
16   } else {
17    res.sendStatus(401) // unauthorized
18   }
19  }
```

Challenge

- jwt-auth

- Set up an auth0 account and tenant and follow instructions :)