Protecting Your API With OAuth

Dan Moore

CodeMash 2022

Jan 13, 2021

@mooreds

FusionAuth

About FusionAuth

- FusionAuth is the authentication and authorization platform built for developers, by developers.
- FusionAuth solves the problem of building essential user security without adding risk or distracting from the primary application.

We Are an OAuth Server

- Many clients protect APIs using our system

About Me

About Me

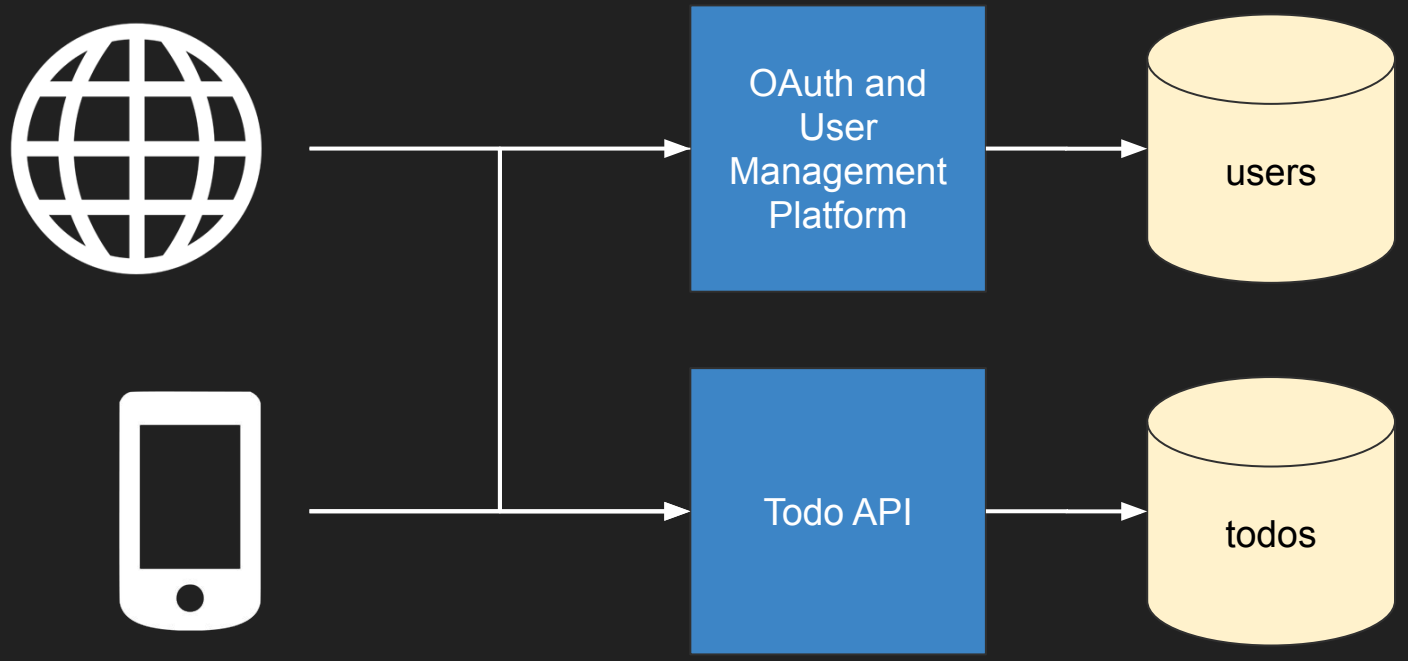
- Who cares

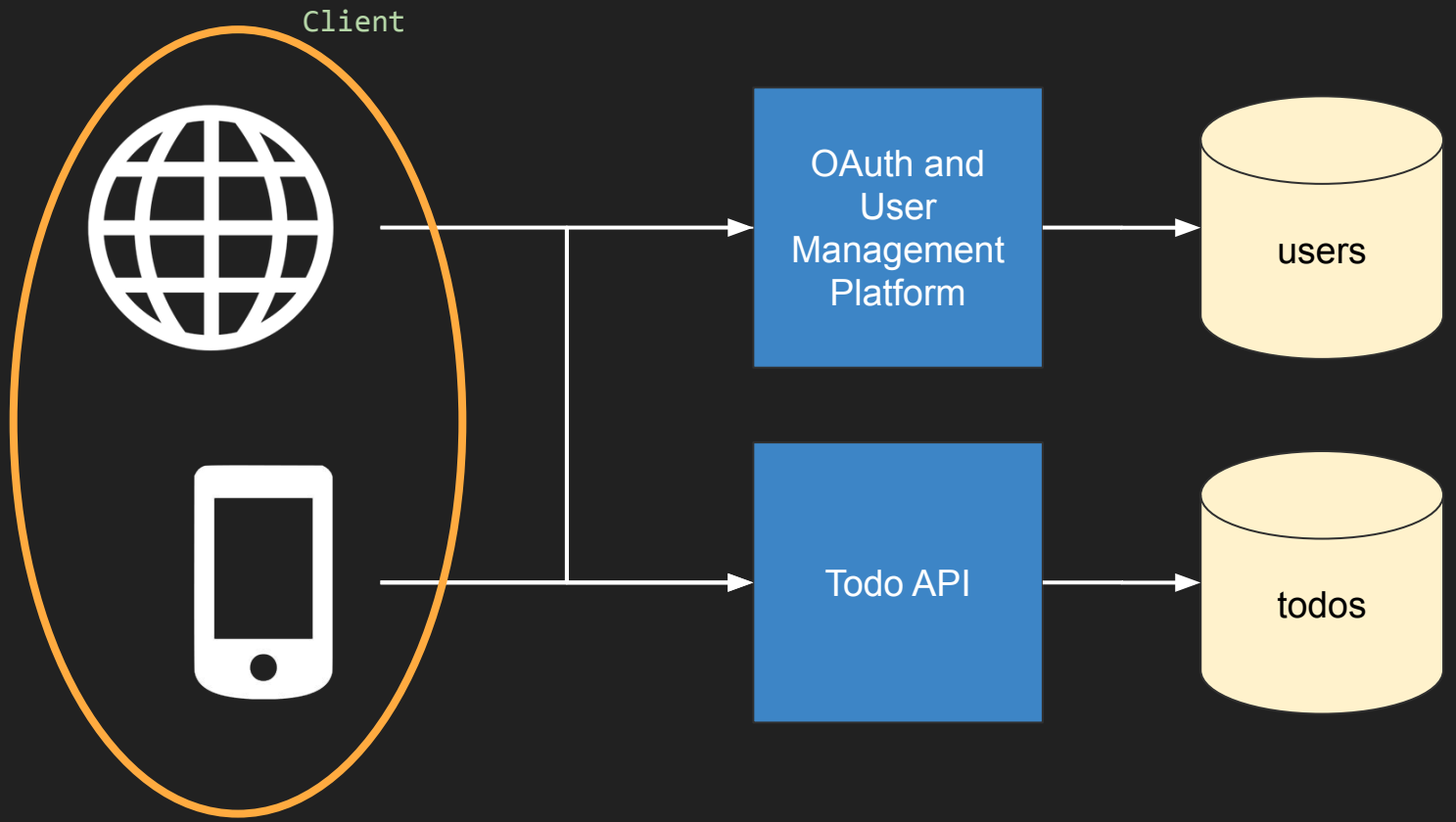
Questions

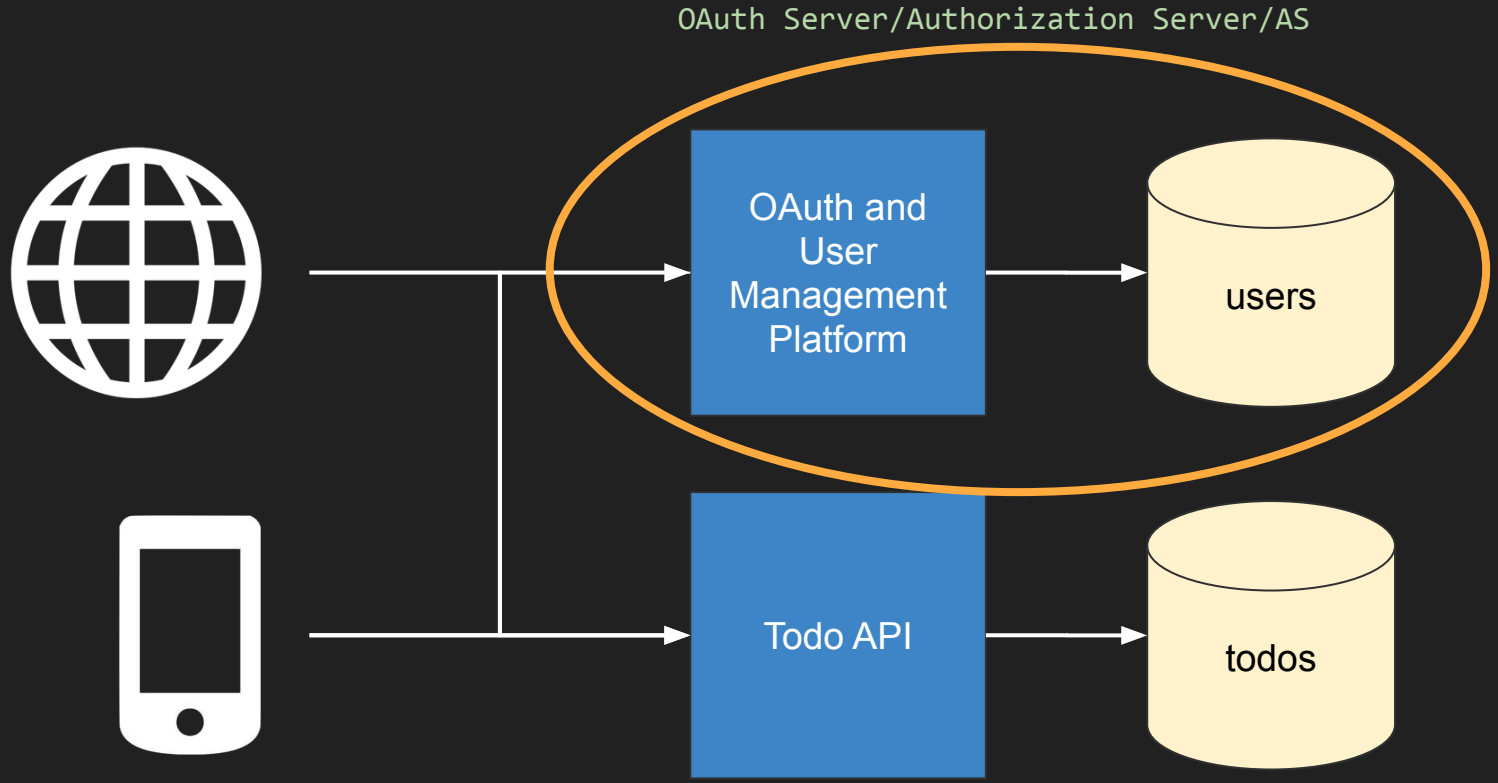
- Just ask
  - Please wave your hand

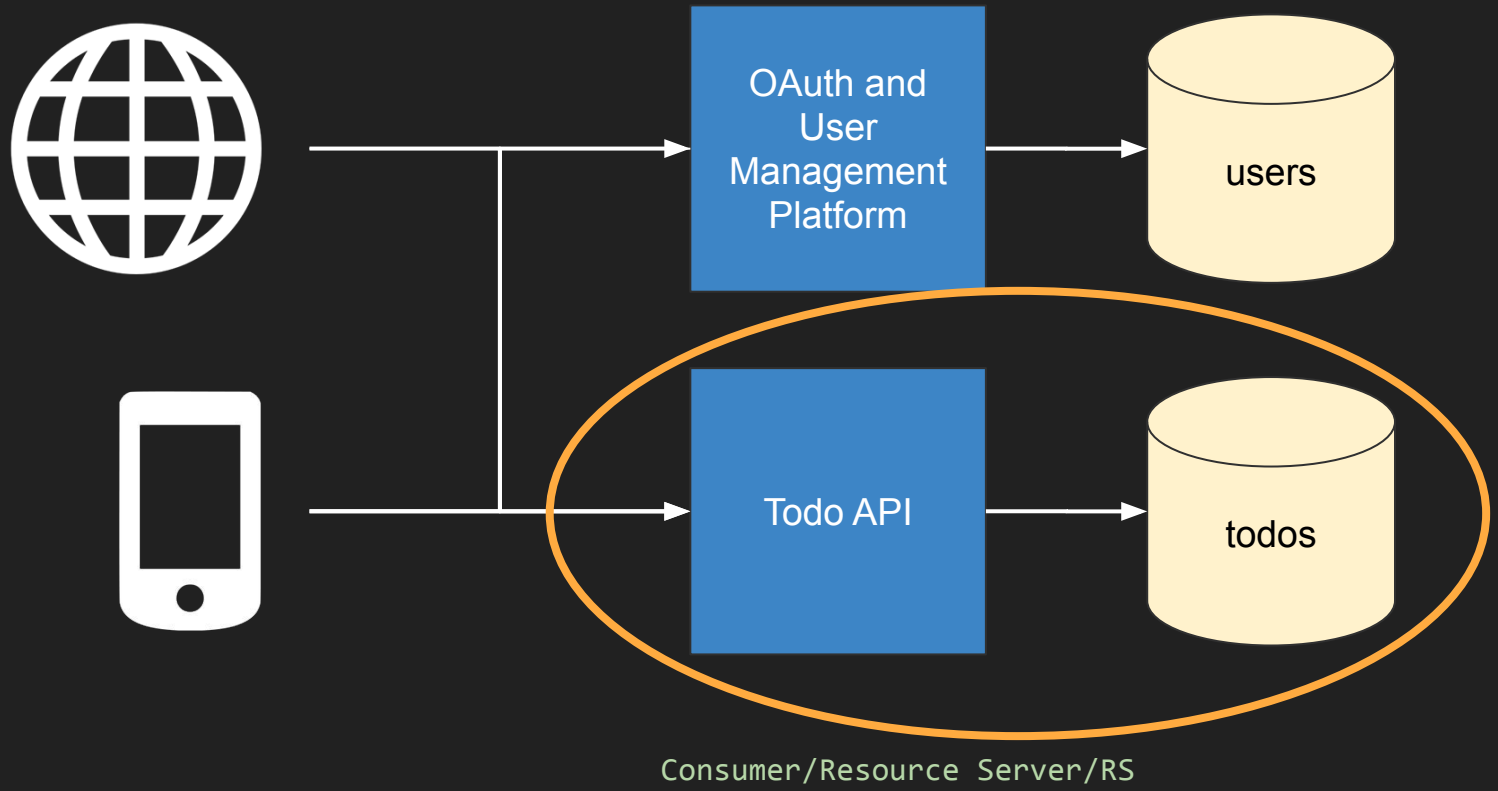
What I'll Cover

- What problem does OAuth solve
- How to get a token
- How to care for tokens
  - As a client
  - As an consumer


OAuth and User Management Platform
users
Todo API
todos

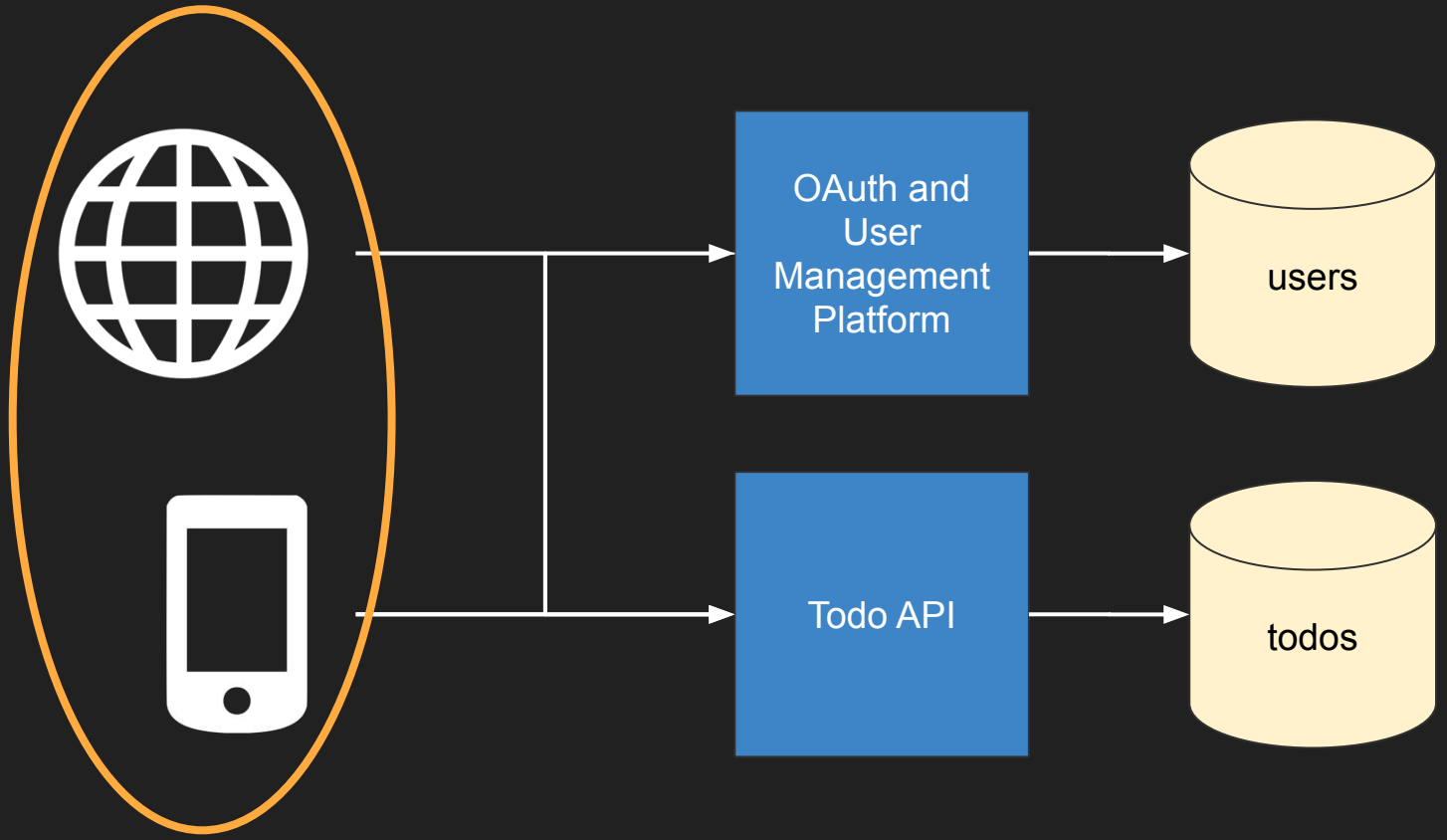

Client
OAuth and User Management Platform
users
Todo API
todos


OAuth Server/Authorization Server/AS
OAuth and User Management Platform
users
Todo API
todos


OAuth and User Management Platform
users
Todo API
todos
Consumer/Resource Server/RS

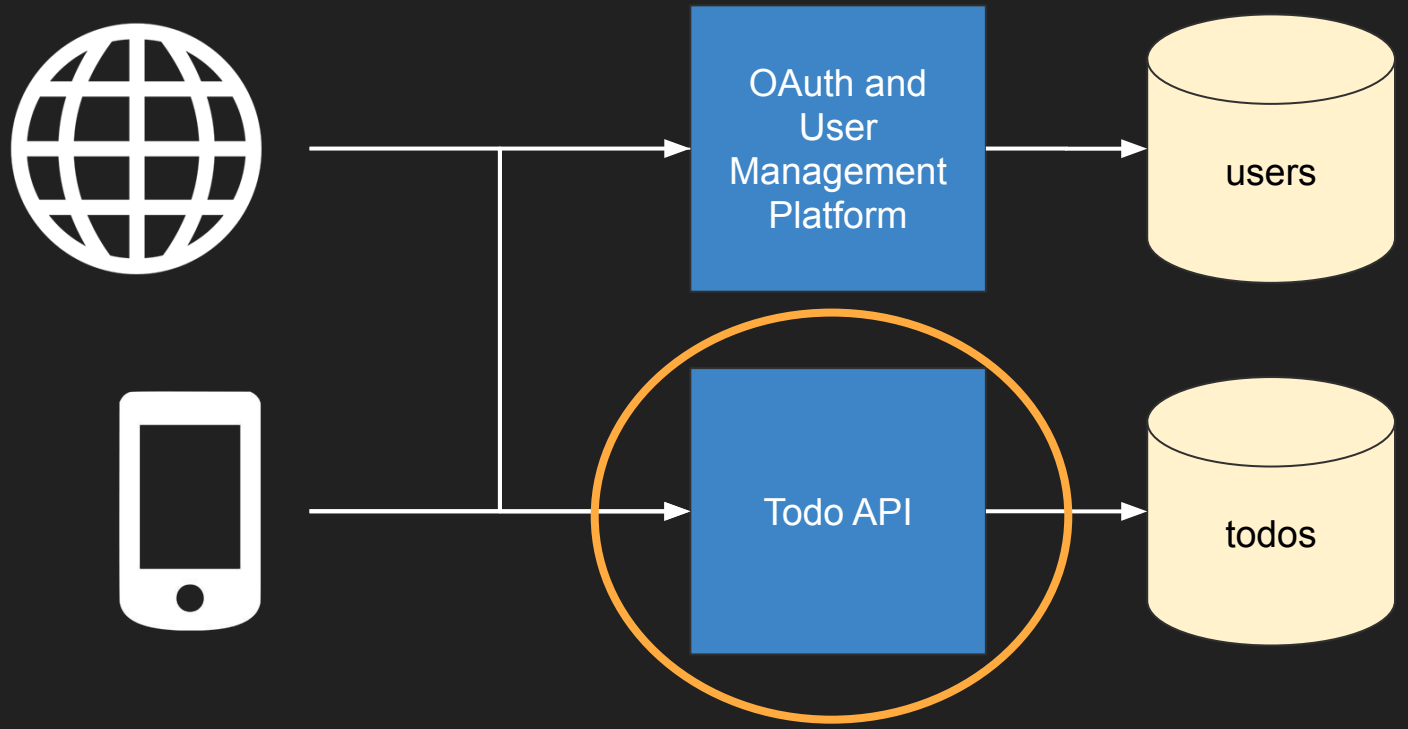
What Problem Does OAuth Solve

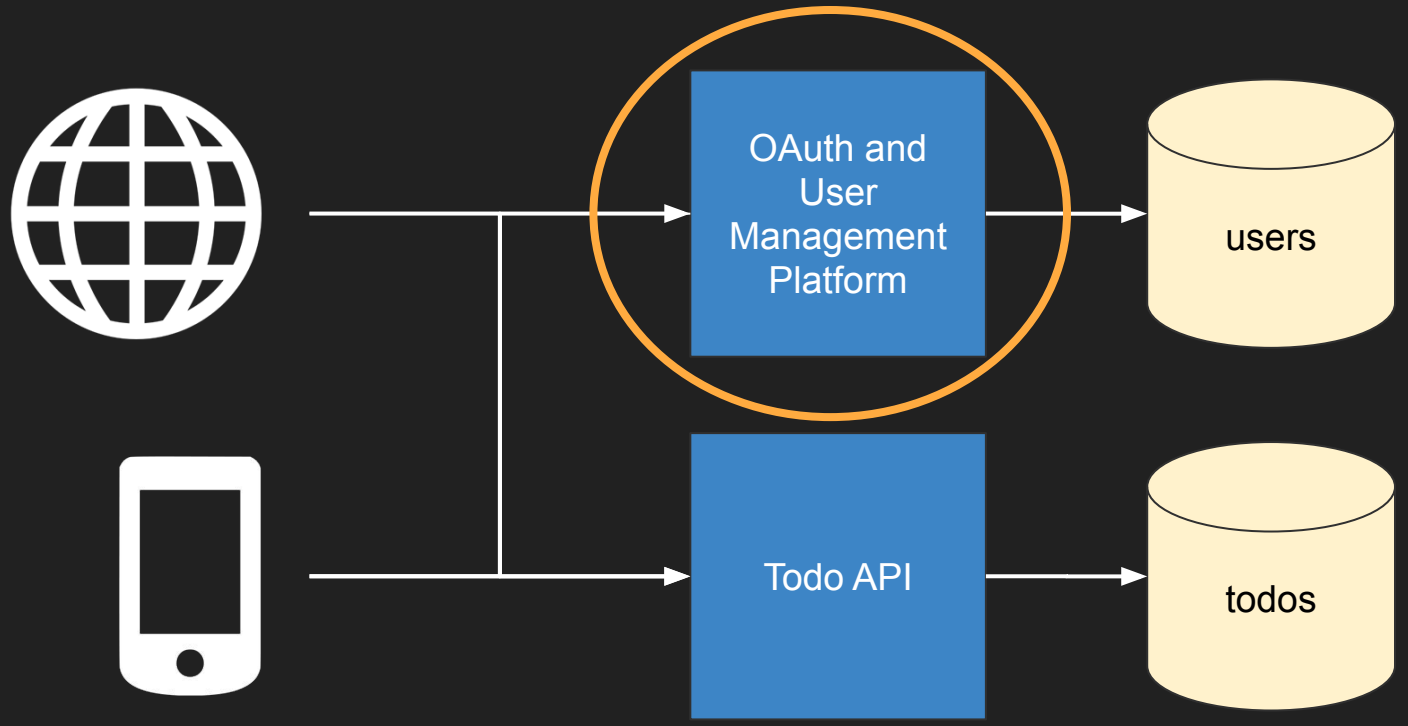
Secure Delegated Access



@mooreds

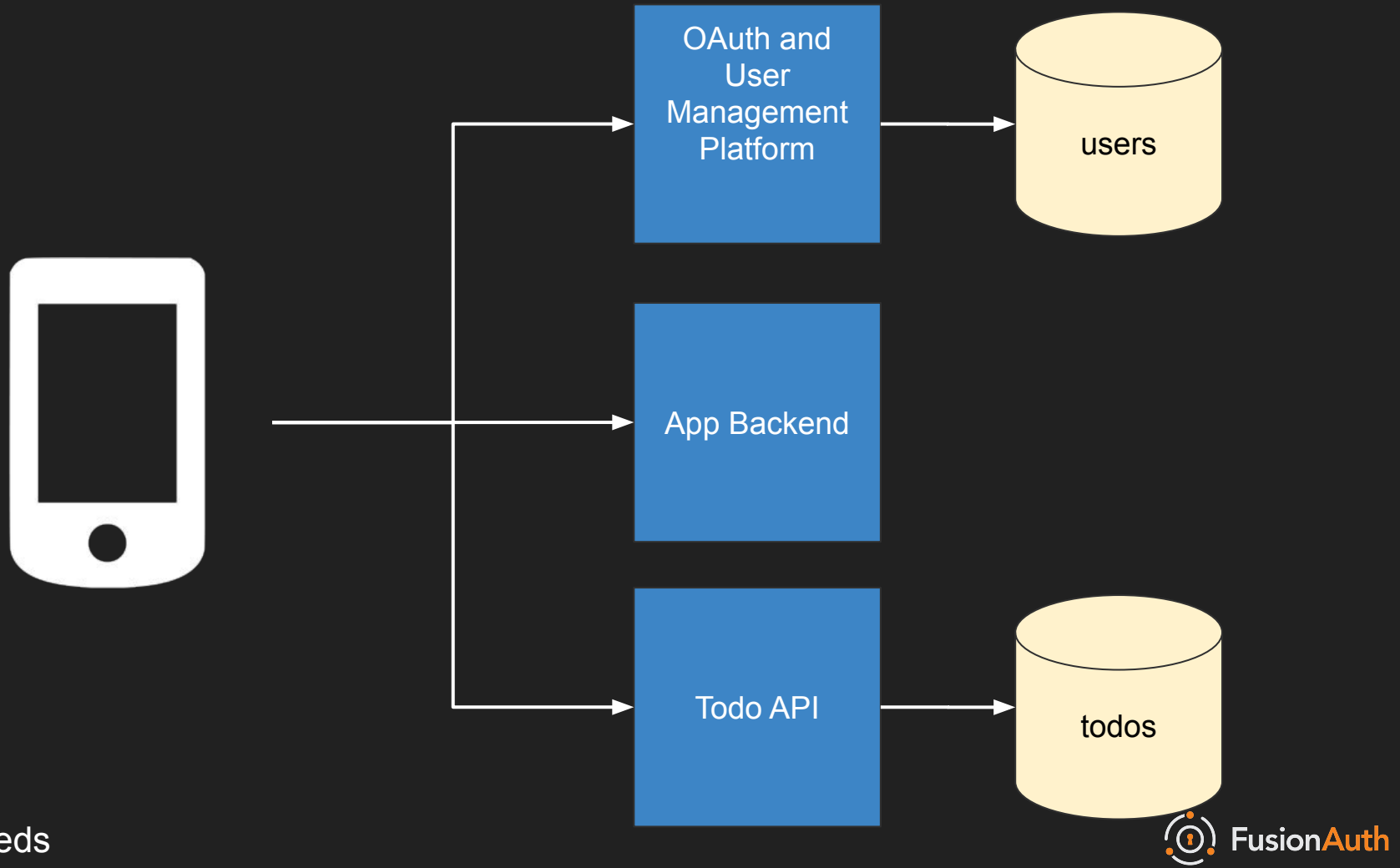
FusionAuth


OAuth and User Management Platform
users
Todo API
todos


OAuth and User Management Platform
users
Todo API
todos

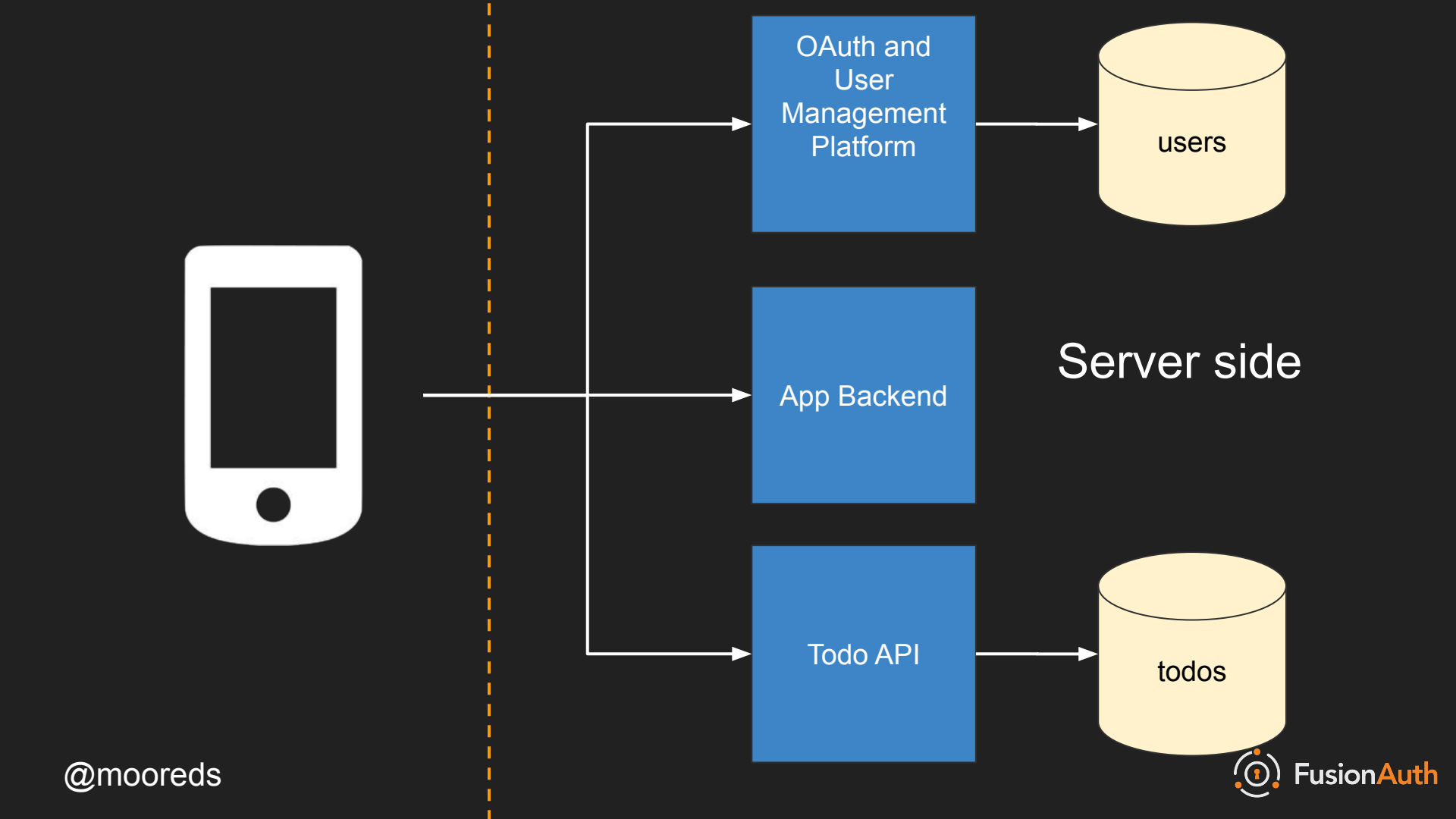
How To Get a Token

Authorization Code Grant

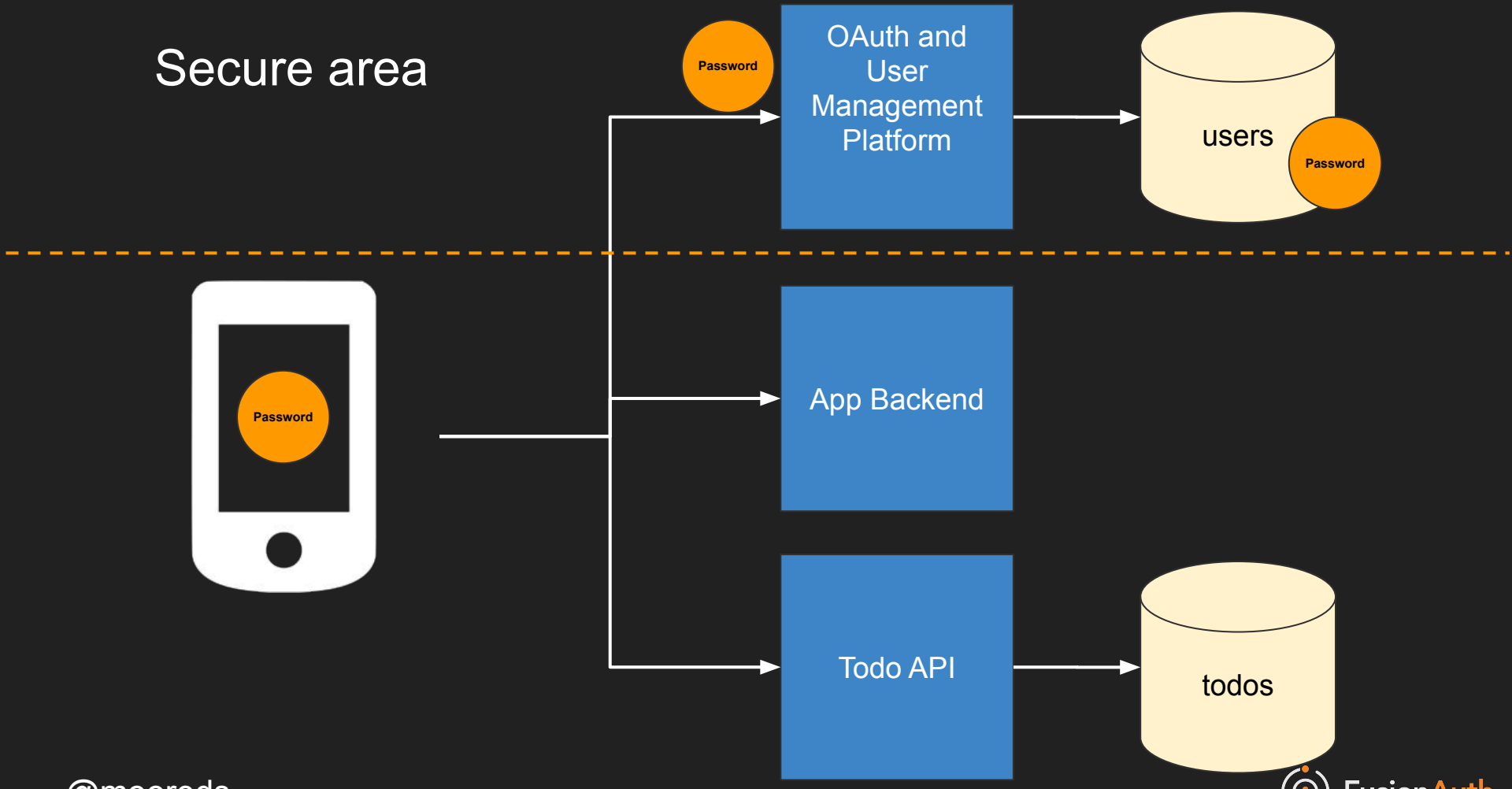


@mooreds

FusionAuth


OAuth and User Management Platform
users
App Backend
Server side
Todo API
todos
@mooreds
FusionAuth

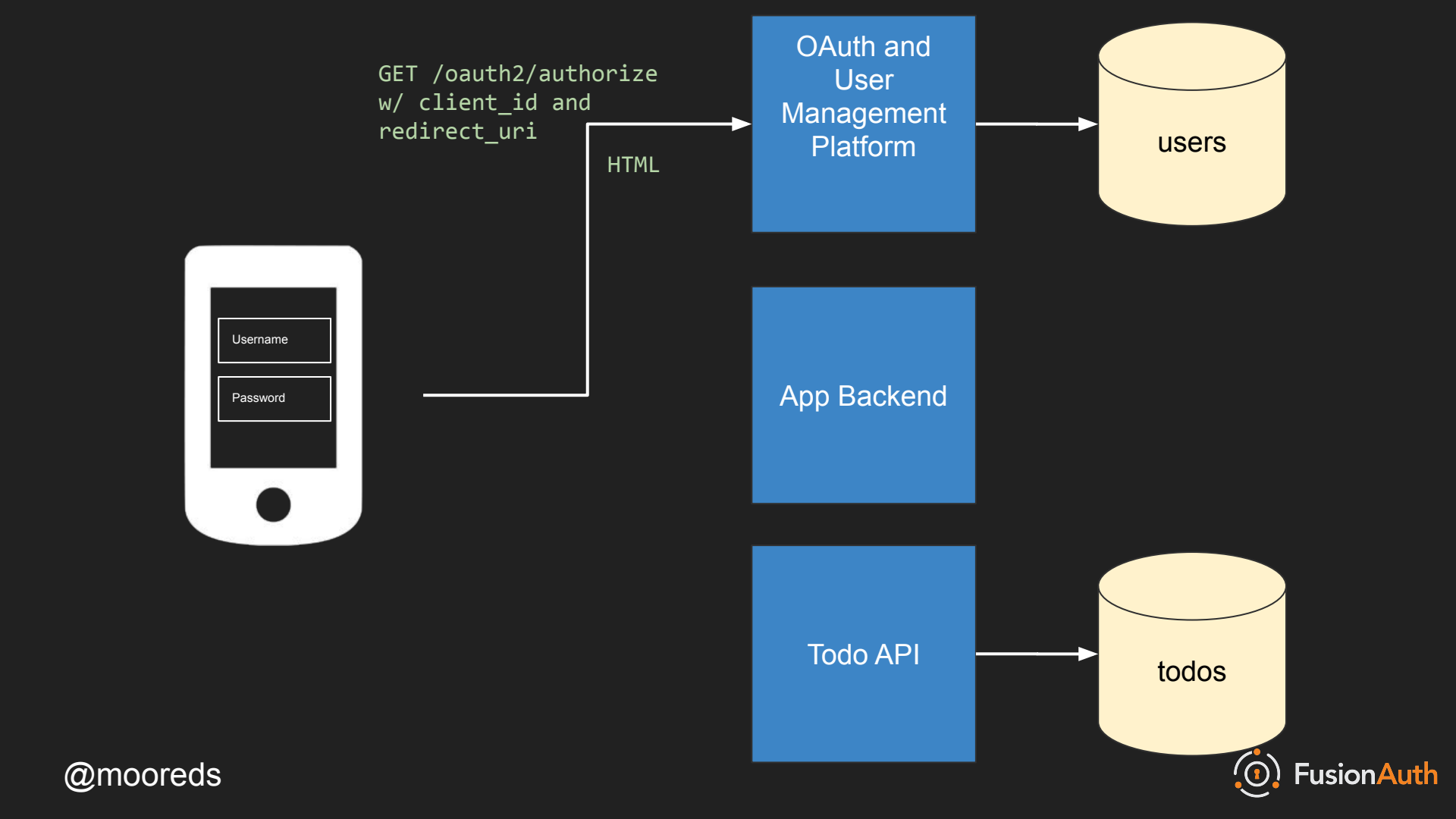
Secure area

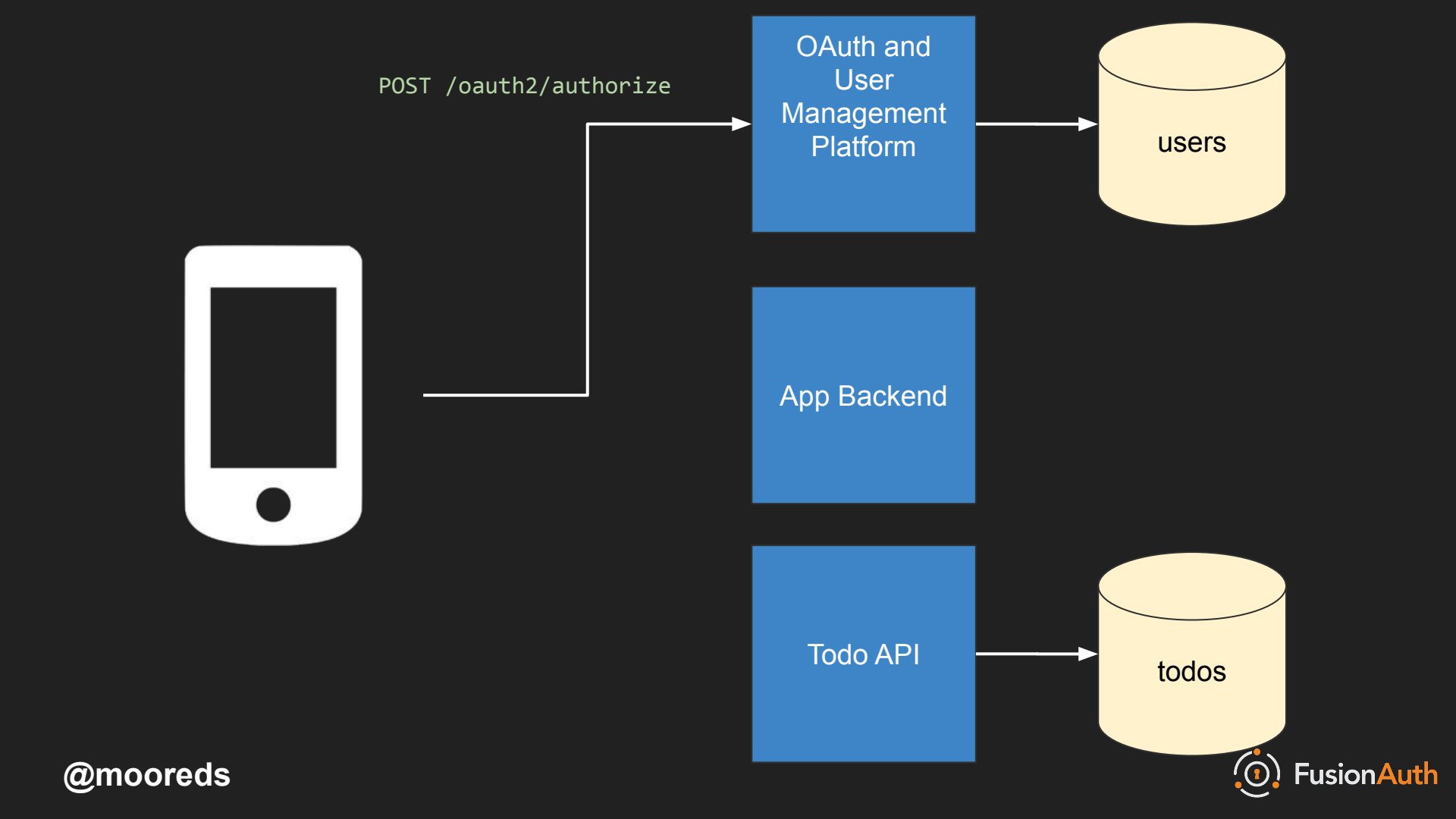


@mooreds

FusionAuth

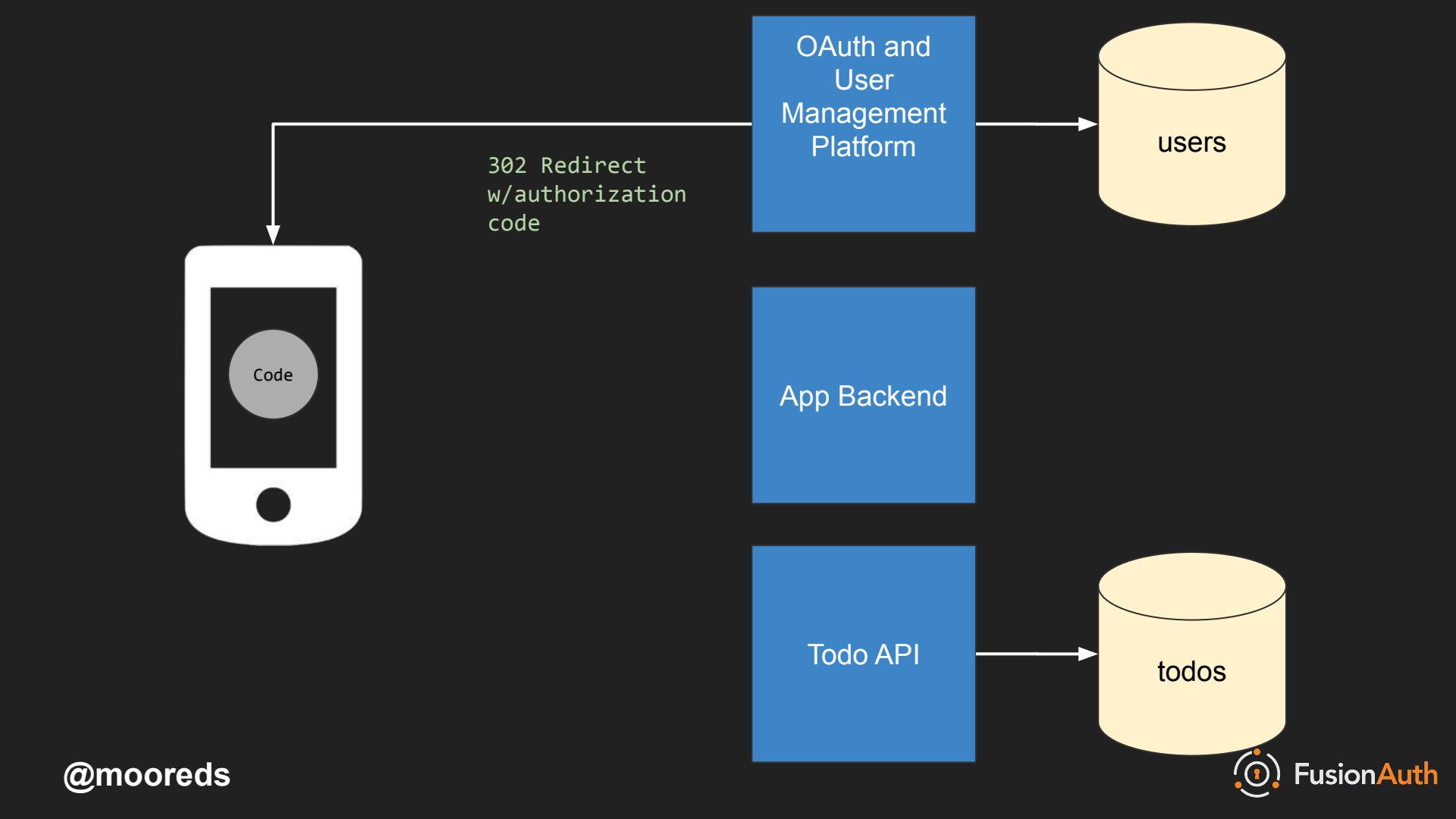
How Does the Authorization Code Grant Work?


GET /oauth2/authorize
w/ client_id and
redirect_uri
HTML
OAuth and User Management Platform
users
Username
Password
App Backend
Todo API
todos
@mooreds
FusionAuth

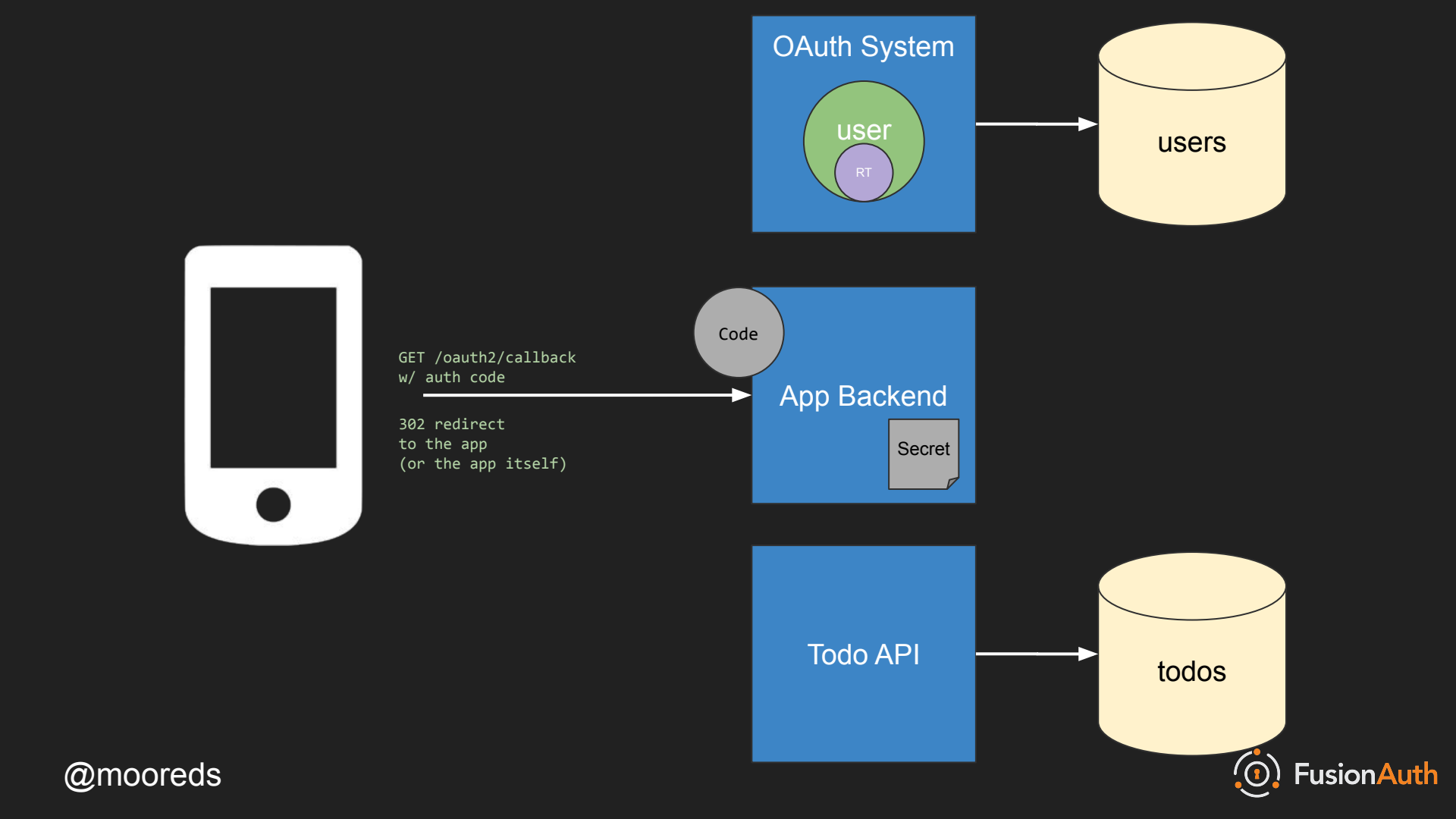


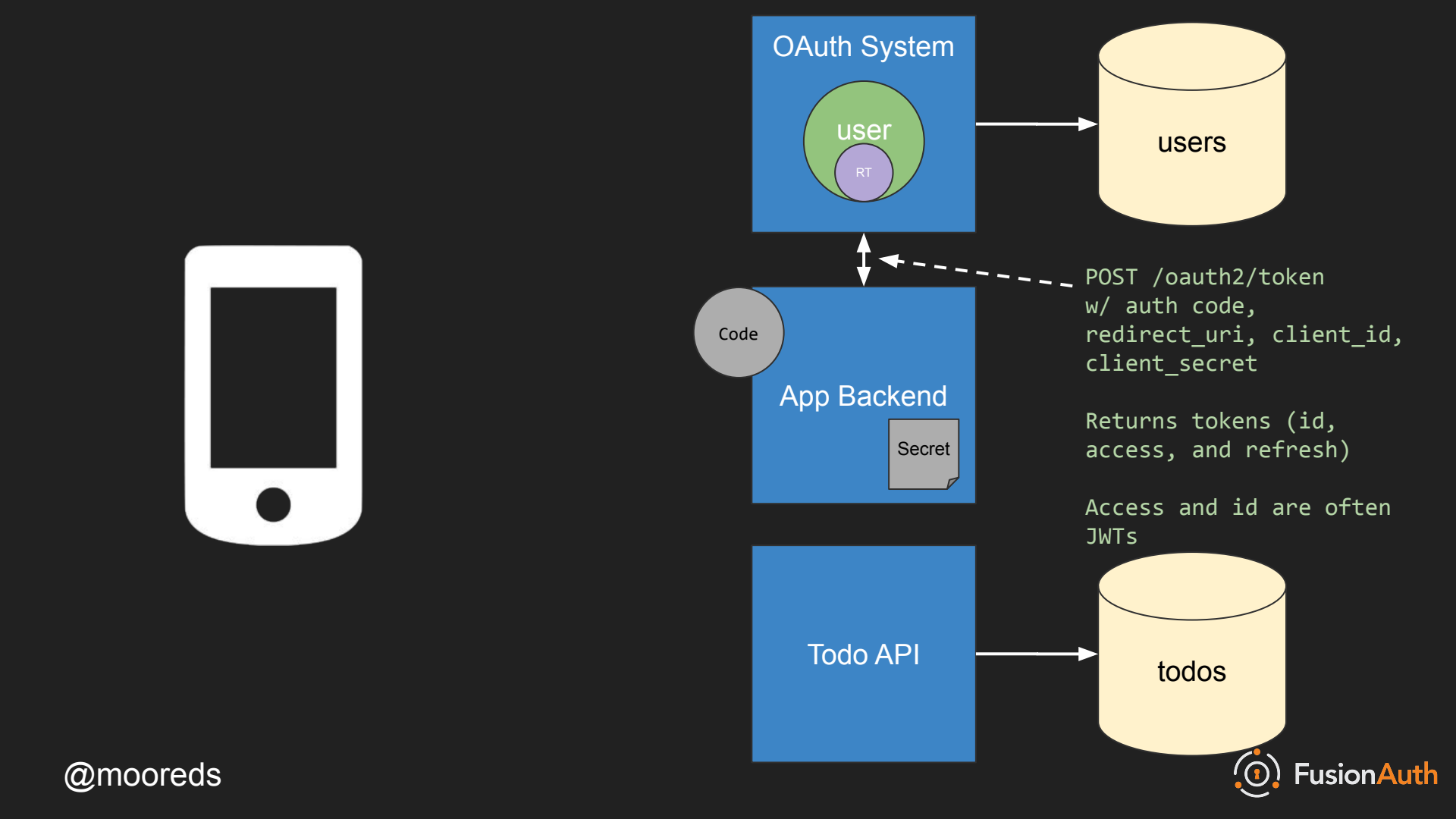
@mooreds

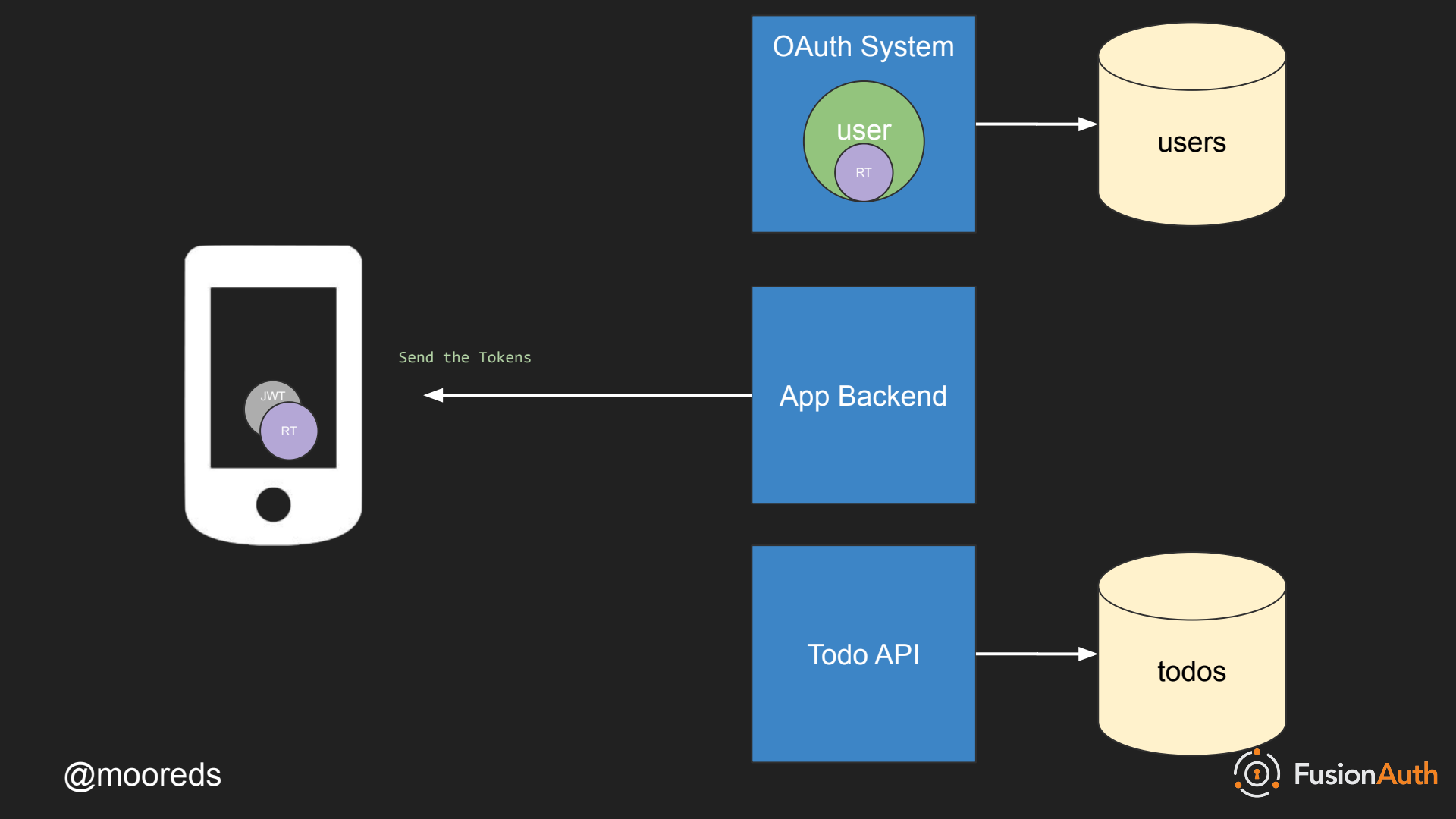
FusionAuth

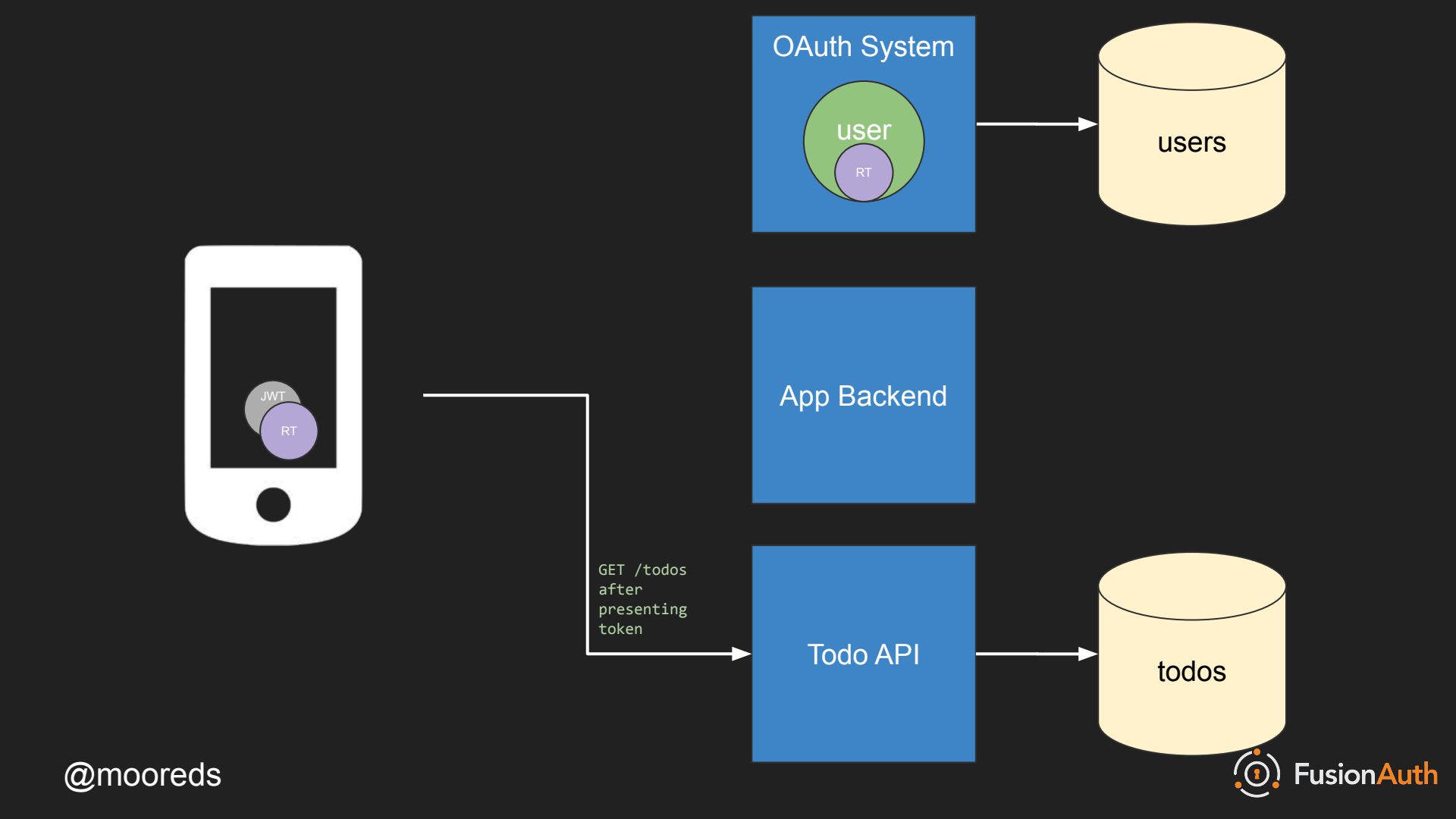

OAuth and User Management Platform
users
302 Redirect w/authorization code
Code
App Backend
Todo API
todos
@mooreds
FusionAuth

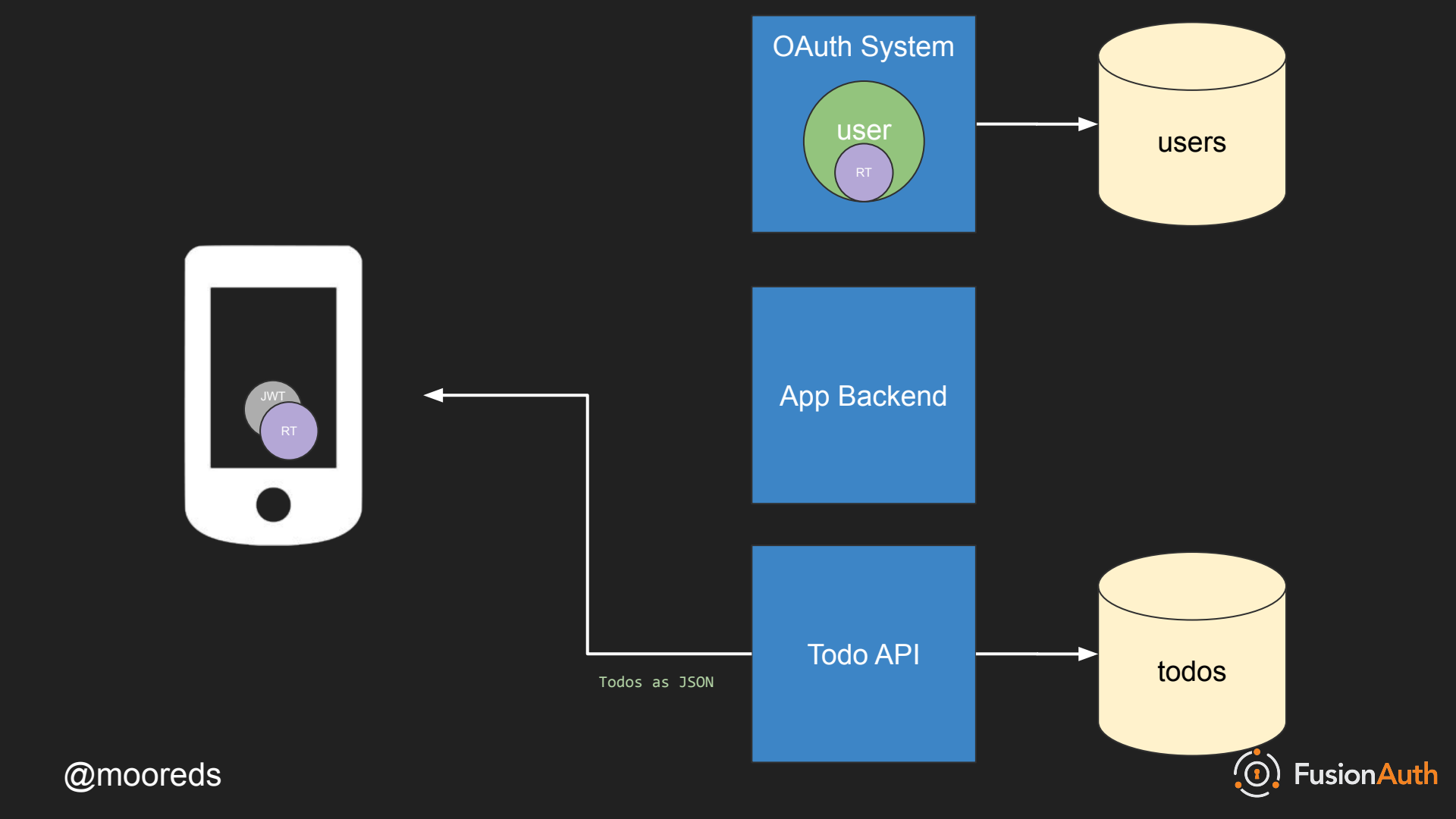
SplxlOBeZQQYbYS6WxSbIA


OAuth System
user
RT
users
GET /oauth2/callback
w/ auth code
302 redirect
to the app
(or the app itself)
Code
App Backend
Secret
Todo API
todos
@mooreds
FusionAuth

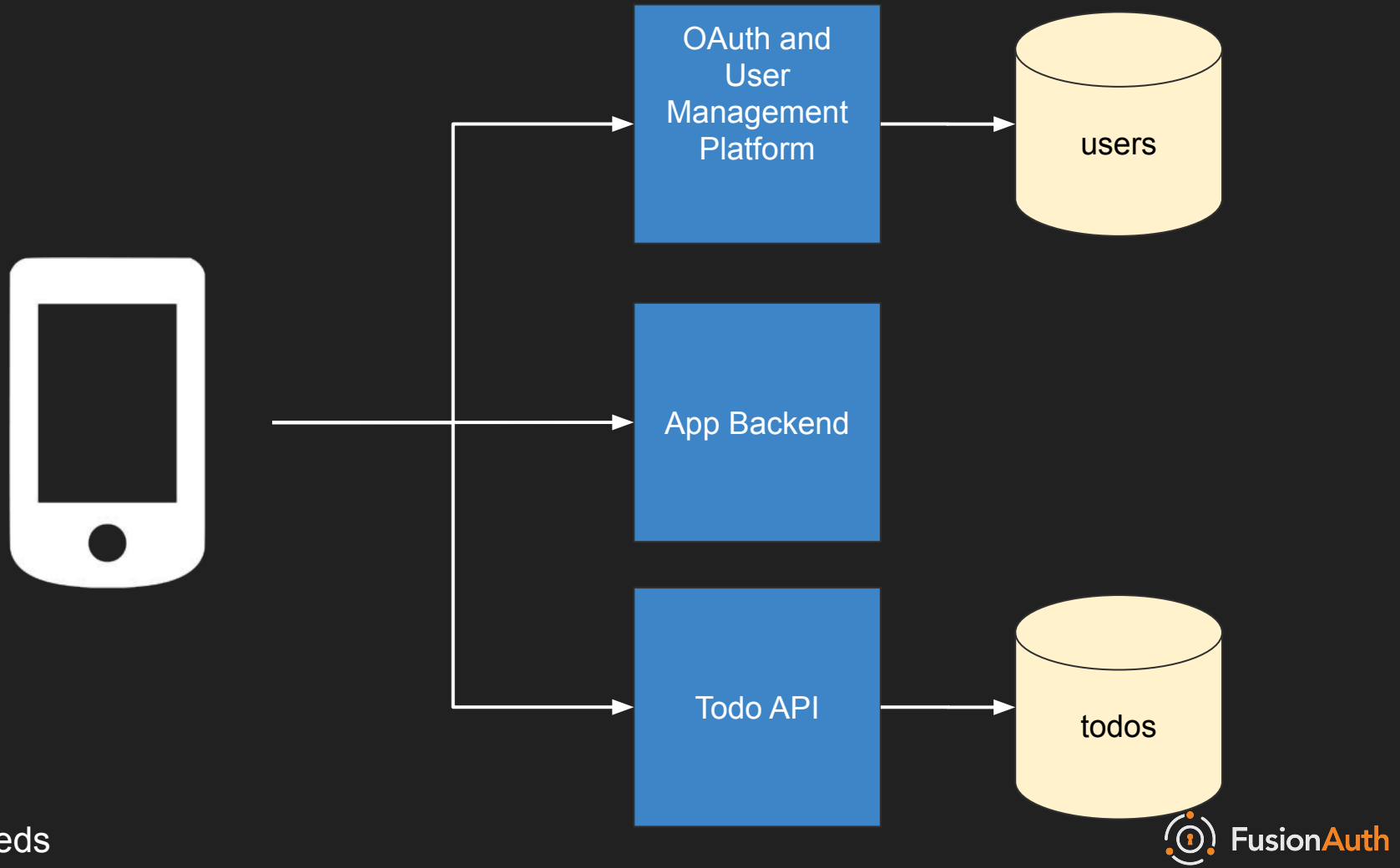

OAuth System
user
RT
users
POST /oauth2/token
w/ auth code,
redirect_uri, client_id,
client_secret
Returns tokens (id,
access, and refresh)
Access and id are often
JWTs
Code
App Backend
Secret
Todo API
todos
@mooreds
FusionAuth


OAuth System
user
RT
users
JWT
RT
Send the Tokens
App Backend
Todo API
todos
@mooreds
FusionAuth


OAuth System
user
RT
users
JWT
RT
App Backend
GET /todos
after
presenting
token
Todo API
todos
@mooreds
FusionAuth


OAuth System
user
RT
users
JWT
RT
App Backend
Todo API
Todos as JSON
todos
@mooreds
FusionAuth

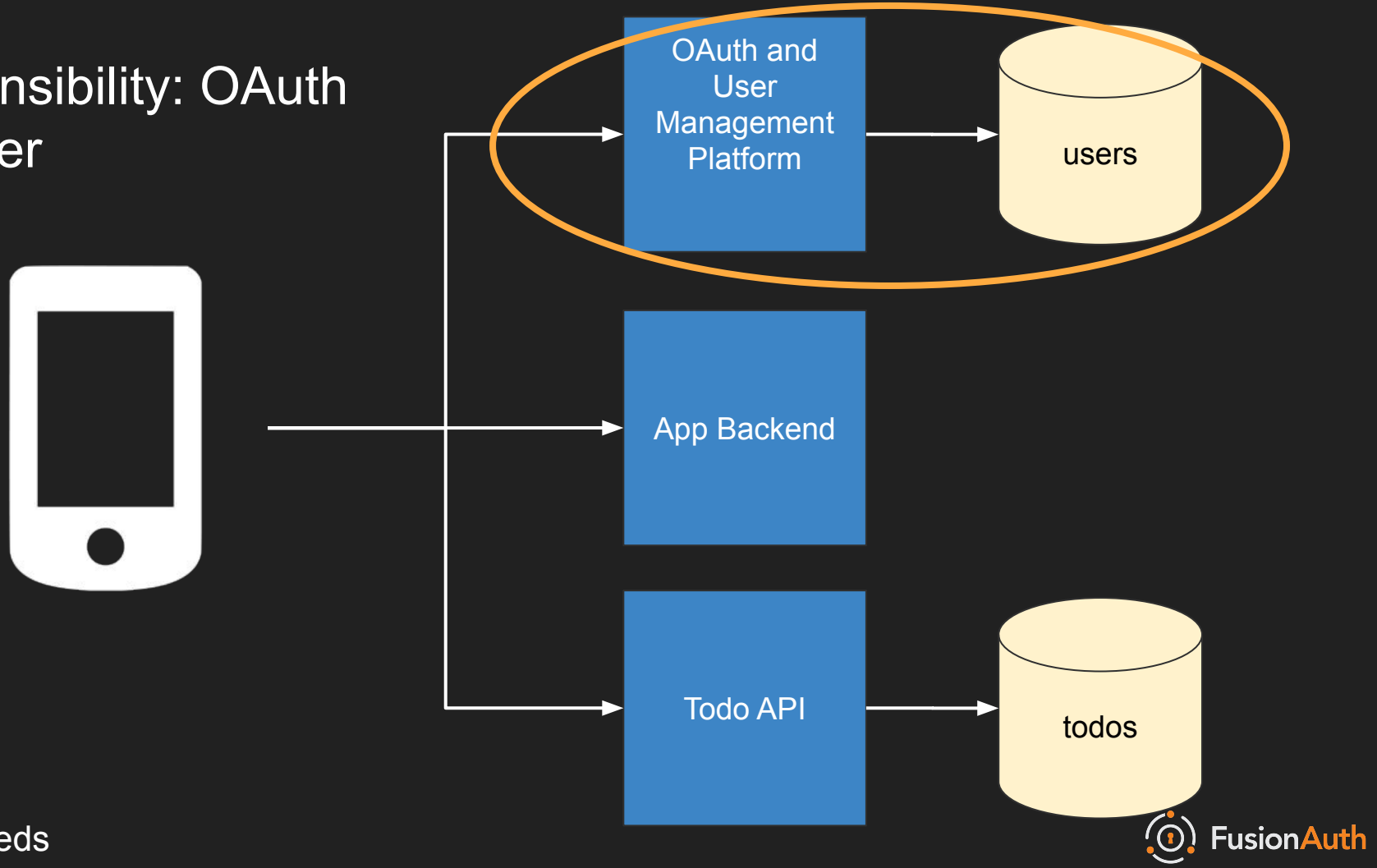
Responsibilities



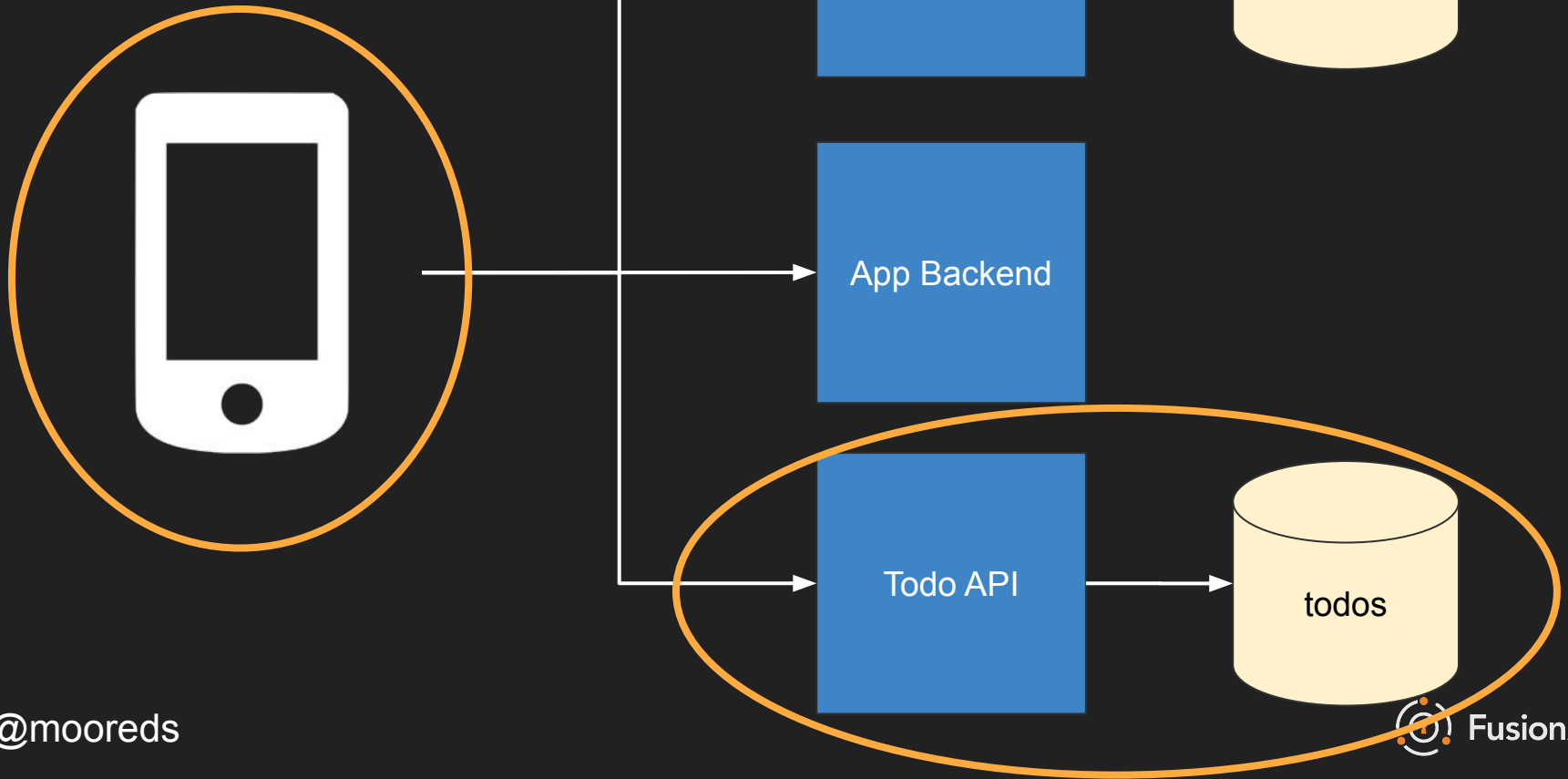
@mooreds

FusionAuth

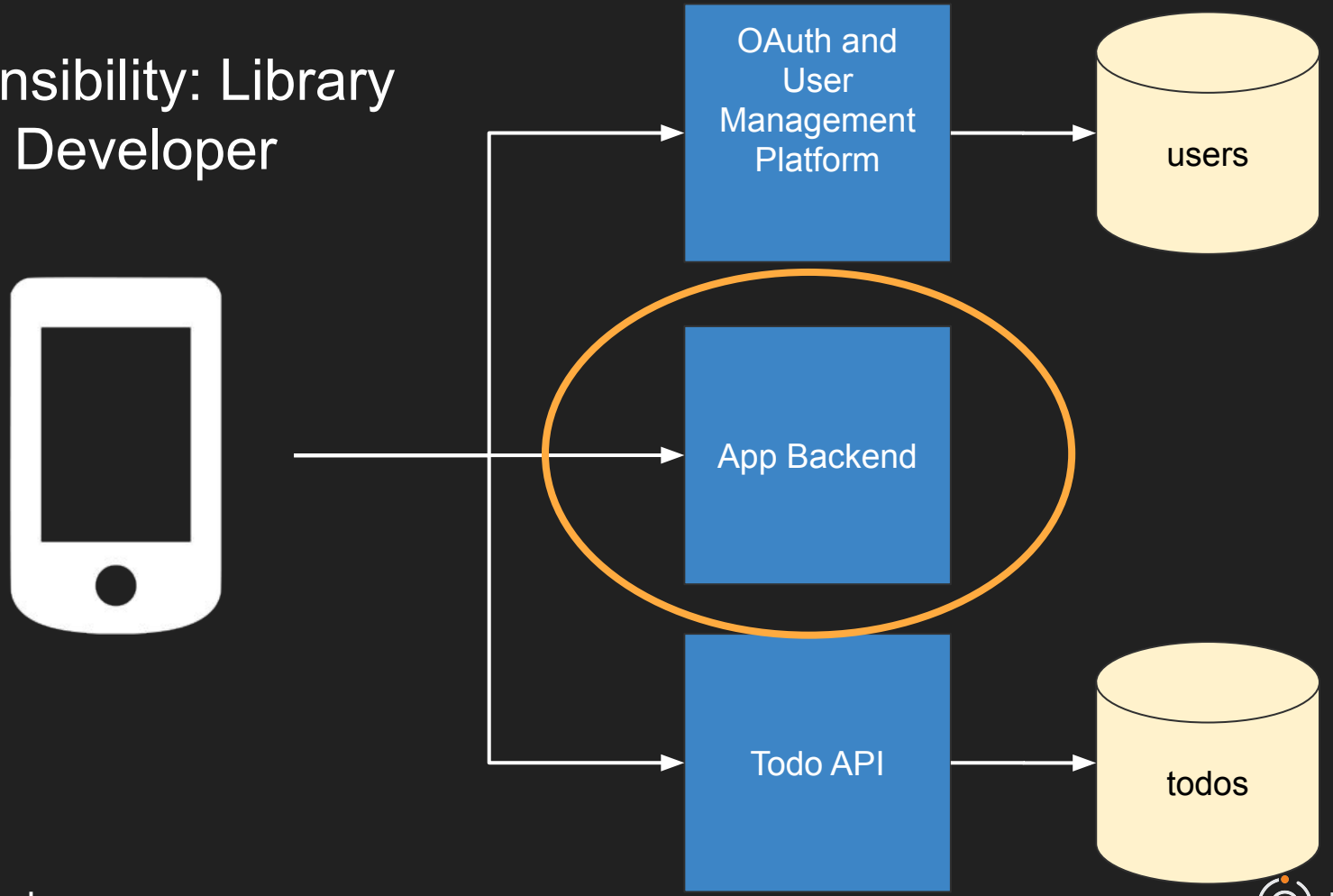
Responsibility: OAuth Provider



Responsibility: Application Developer



Responsibility: Library or App Developer



Questions?

Tokens

Tokens

- OAuth
  - Access tokens
  - Refresh tokens
- OIDC
  - Id tokens

Access Tokens

- Opaque per spec
- In practice, often JSON Web Tokens (JWTs)
- Presented to the consumer

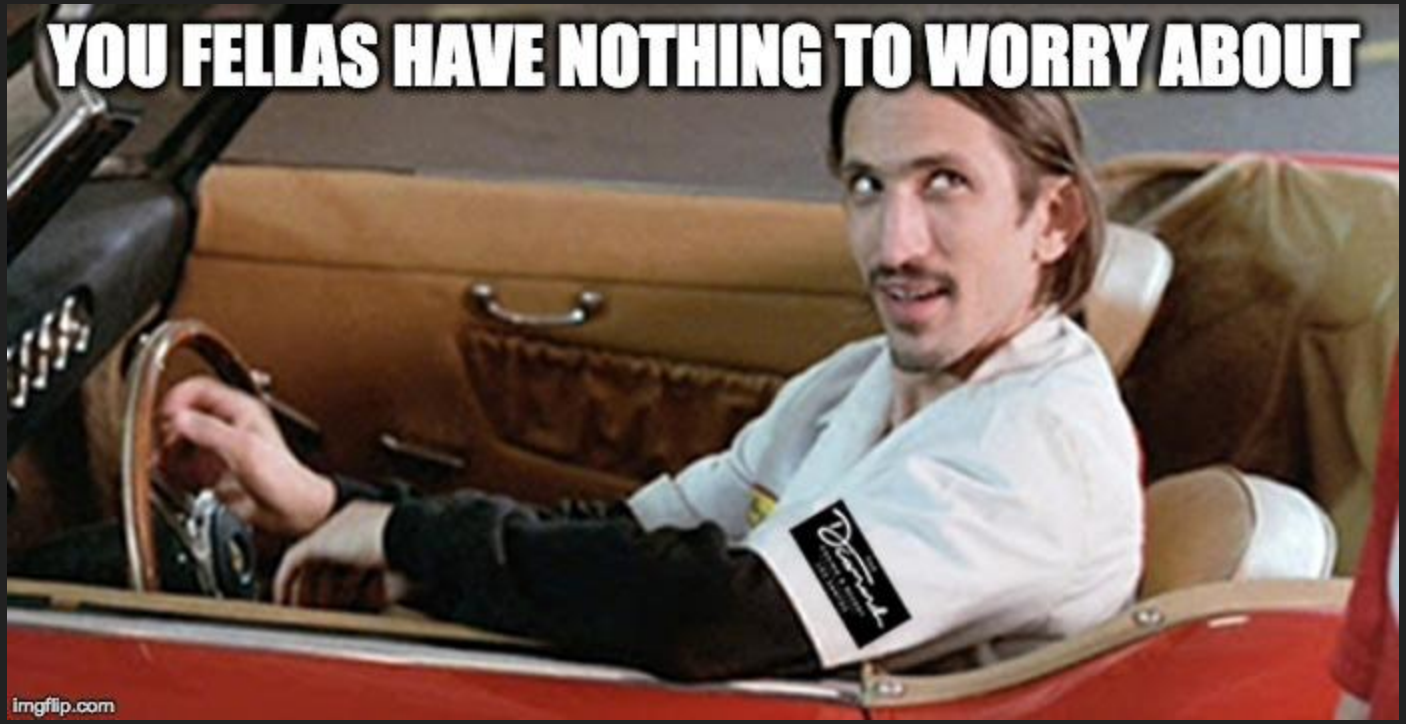
Refresh Tokens

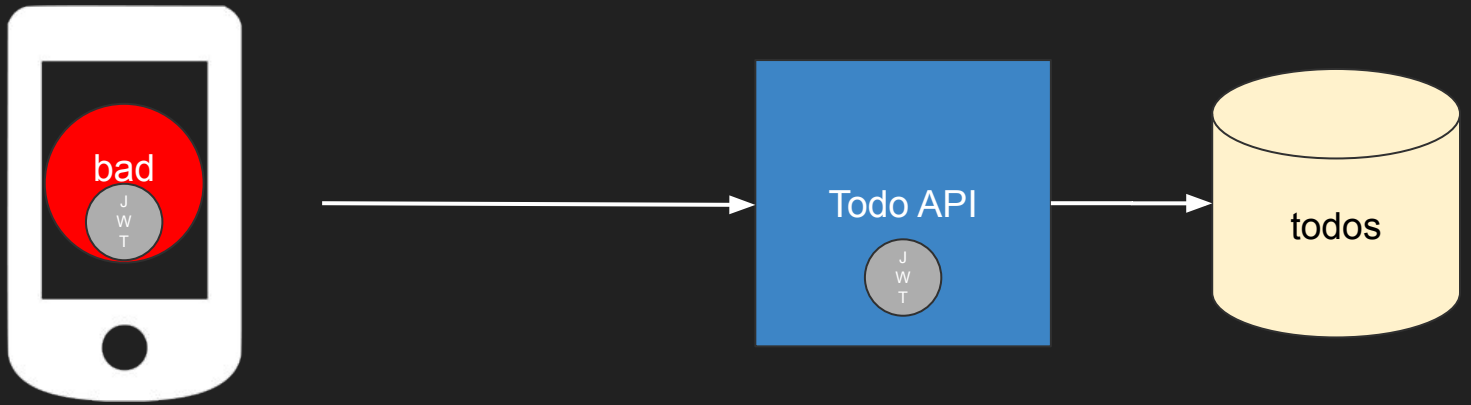
- Opaque
- Used to get new access tokens
- Presented to the OAuth server

Id Tokens

- Part of OIDC
- JWTs
- Authentication not authorization
- Can be used by the client

Bearer Tokens


YOU FELLAS HAVE NOTHING TO WORRY ABOUT
imgflip.com

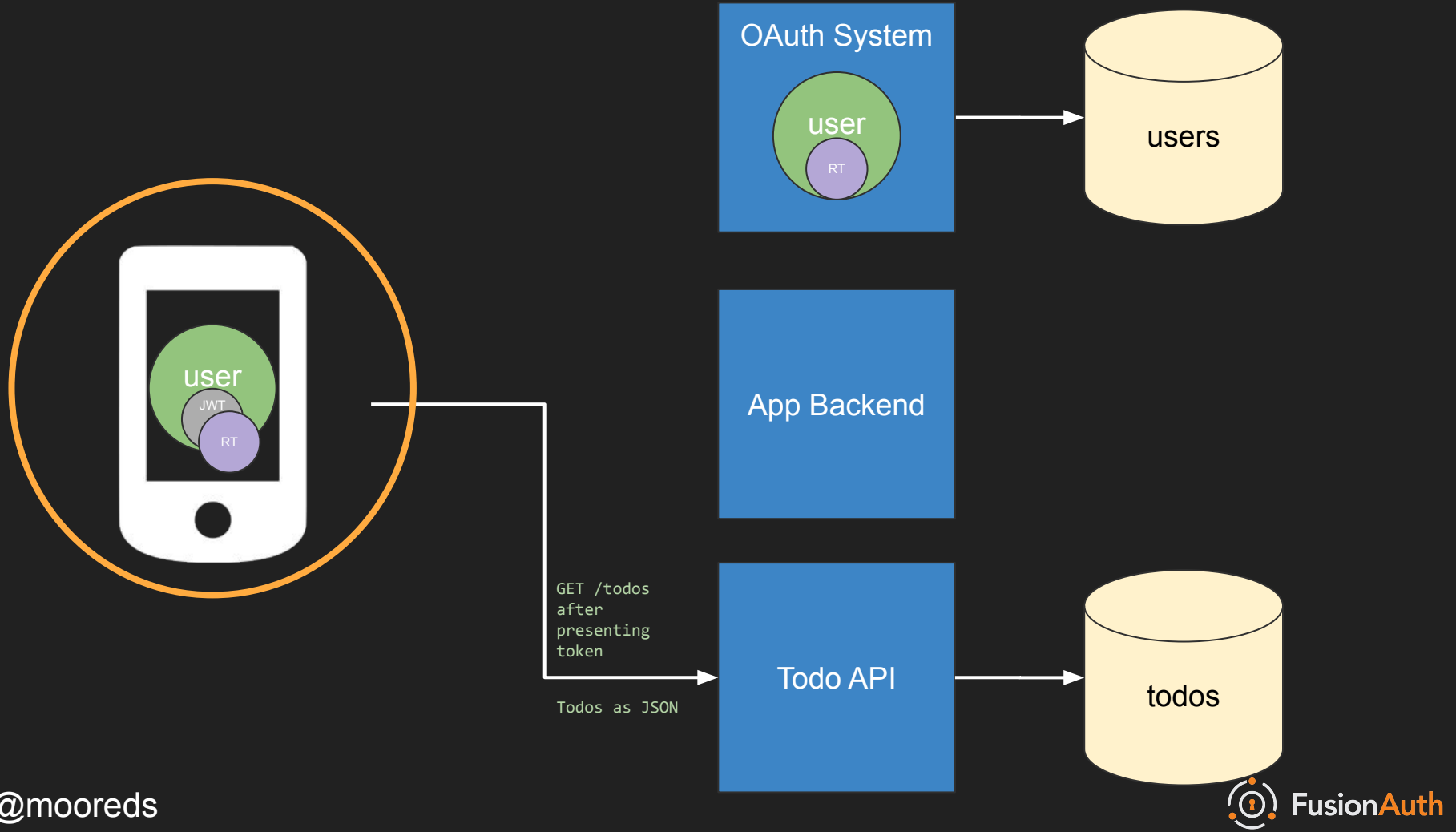

bad
J
W
T
Todo API
J
W
T
todos

Bearer Token Won’t Work?

- Look into DPoP or MTLS
  - DPoP: https://datatracker.ietf.org/doc/html/draft-ietf-oauth-dpop-04
  - MTLS: https://datatracker.ietf.org/doc/html/rfc8705

Questions?

Client Concerns


OAuth System
user
RT
users
user
JWT
RT
App Backend
GET /todos
after
presenting
token
Todos as JSON
Todo API
todos
@mooreds
FusionAuth

Over the Wire

- HTTPS, please

Over the Wire

- HTTPS, please
- Hide it from caches

Over the Wire

- HTTPS, please
- Hide it from caches
  - Don’t put in URL

Token Storage On the Client

- Mobile clients
- Browsers
- Side-step with Sessions

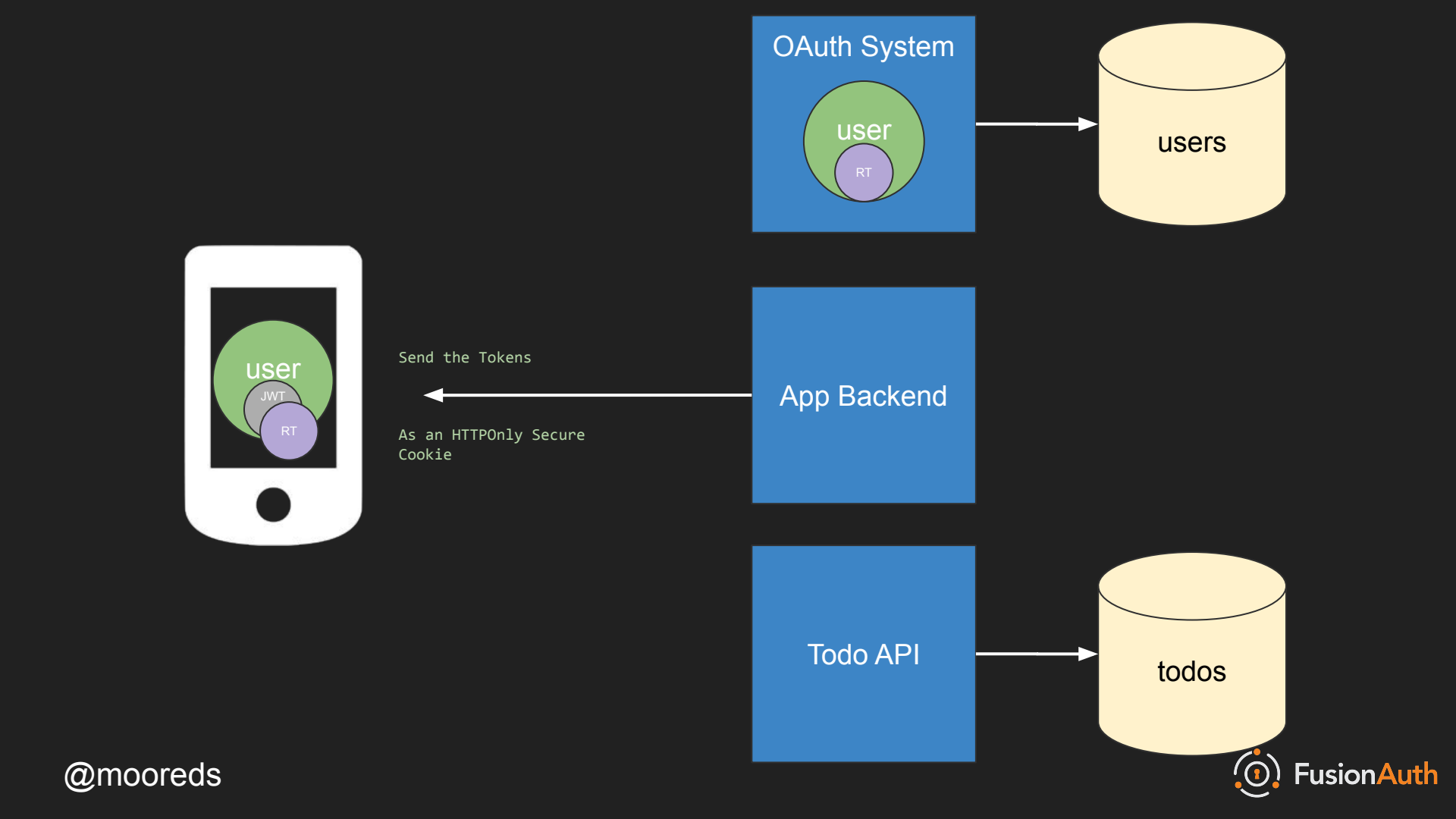
Mobile Application Token Storage

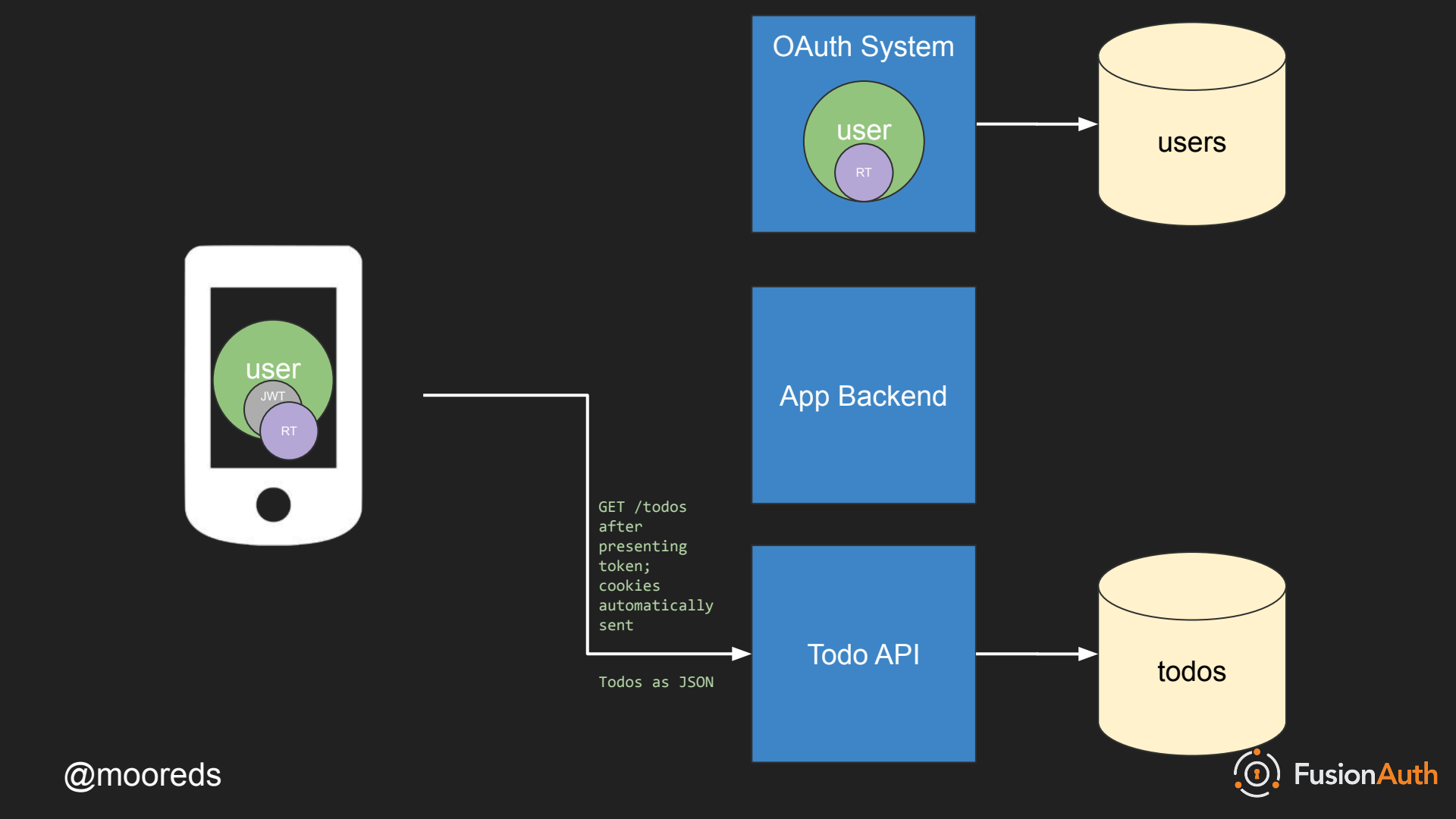
- iOS:
  - Use encrypted storage or the keychain
- Android:
  - Use encrypted storage or app preferences
- Some auth libraries take care of it for you

Browser Storage

Browser Storage

- Cookies


OAuth System
user
RT
users
user
JWT
RT
Send the Tokens
As an HTTPOnly Secure Cookie
App Backend
Todo API
todos
@mooreds
FusionAuth


OAuth System
user
RT
users
user
JWT
RT
App Backend
GET /todos
after
presenting
token;
cookies
automatically
sent
Todos as JSON
Todo API
todos
@mooreds
FusionAuth

Browser Storage

- Cookies
- In memory
  - Good for SPAs

Browser Storage

- Cookies
- In memory
  - Good for SPAs
- Web worker
  - https://gitlab.com/jimdigriz/oauth2-worker

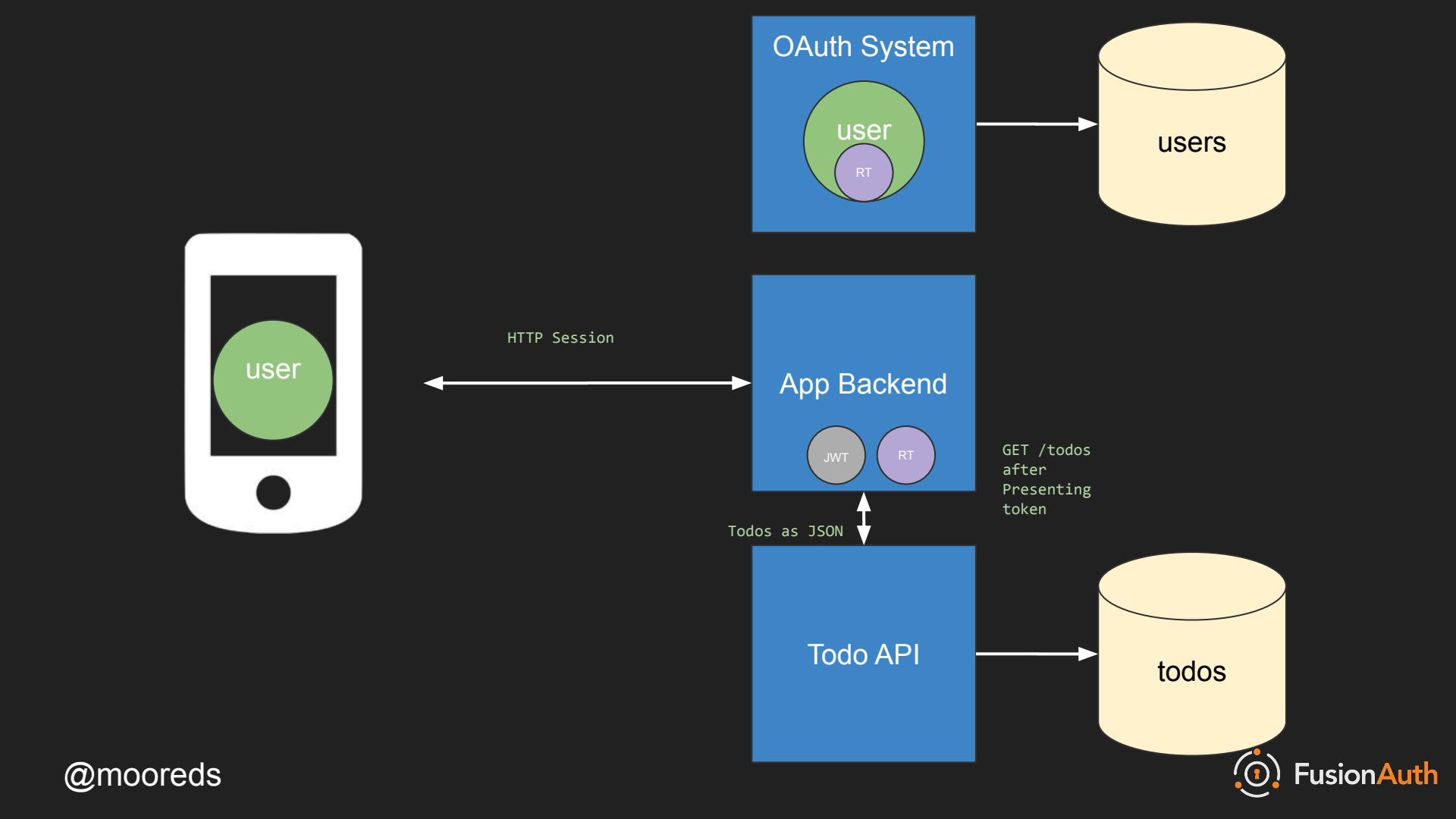
BFF

@mooreds

FusionAuth

BFF

- Back-end for front-end pattern


OAuth System
user
RT
users
user
HTTP Session
App Backend
JWT
RT
GET /todos
after
Presenting
token
Todos as JSON
Todo API
todos
@mooreds
FusionAuth

Questions?

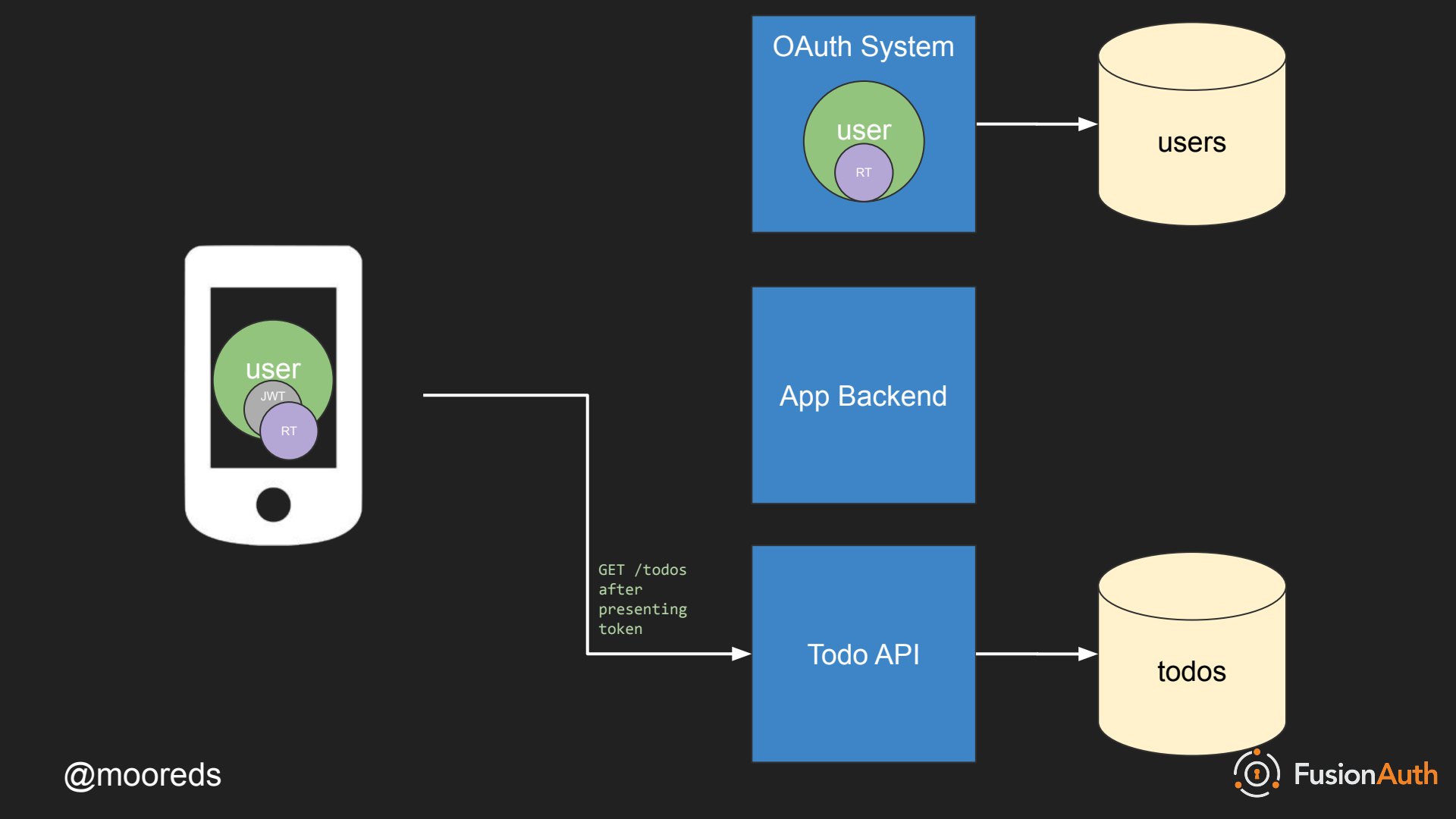
Refresh Tokens

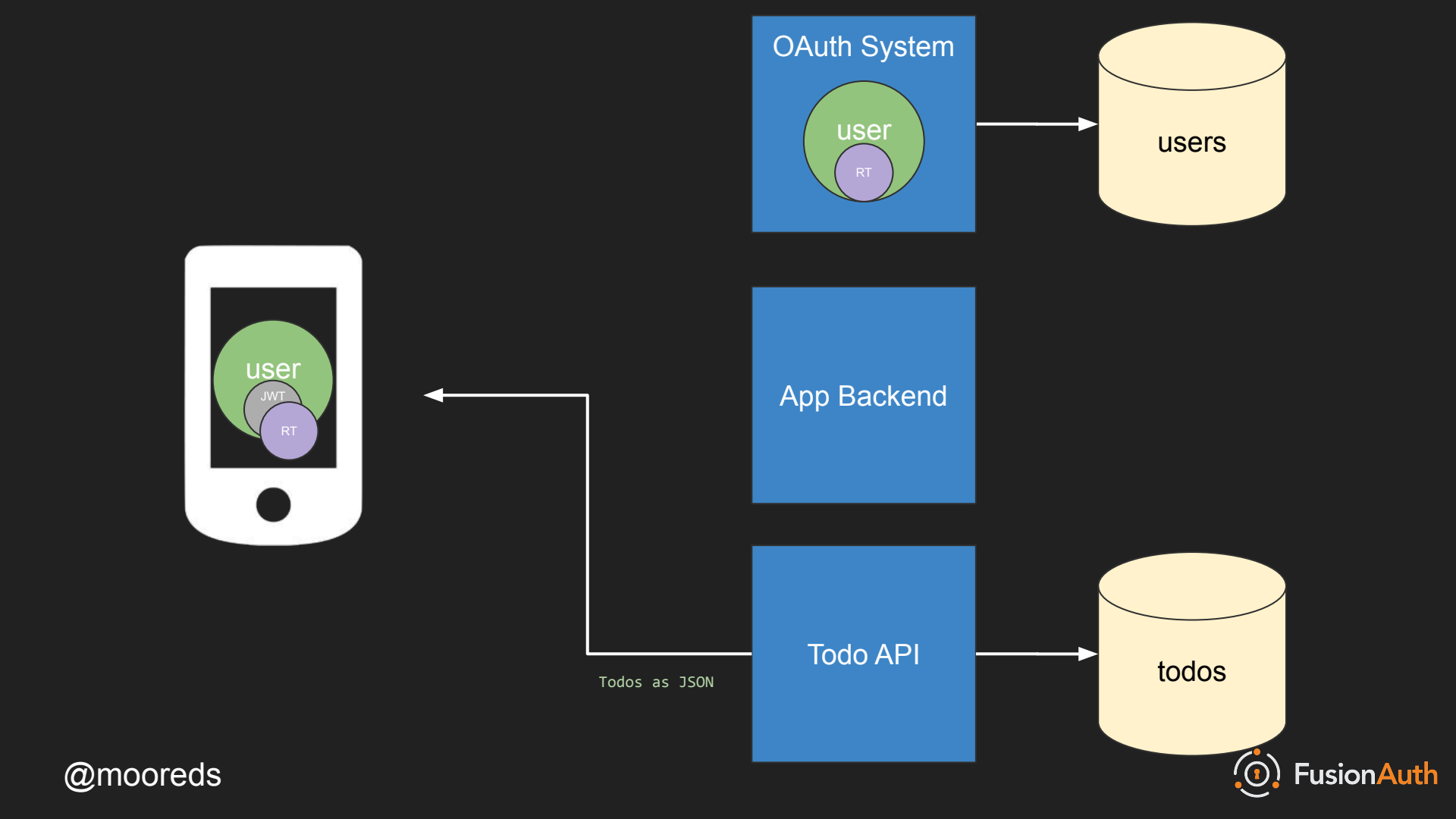
Refresh Tokens

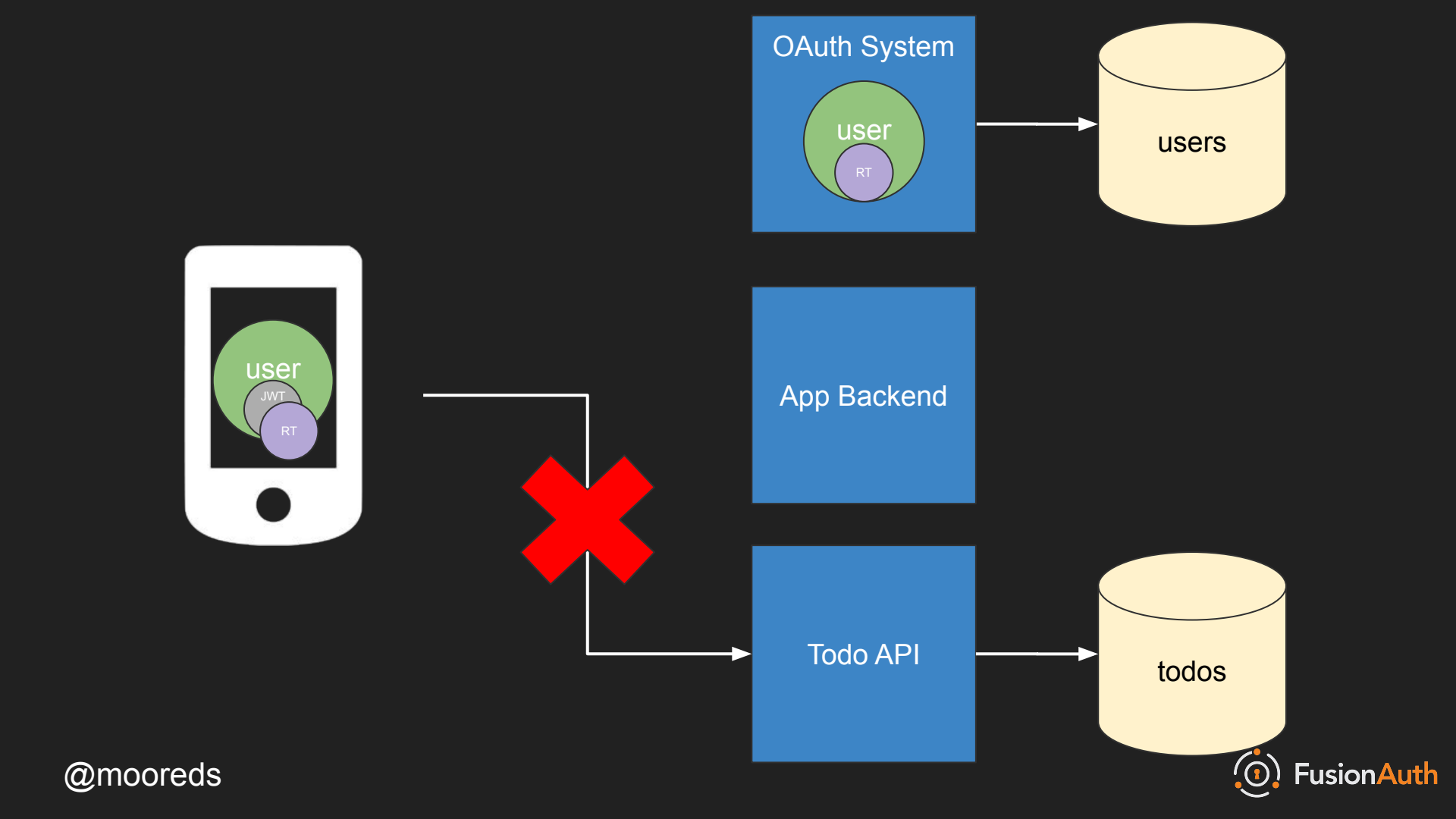
- Access tokens are meant to be short lived (minutes)
- Refresh tokens are long lived (days or months)
- Refresh tokens can be used to create new access tokens

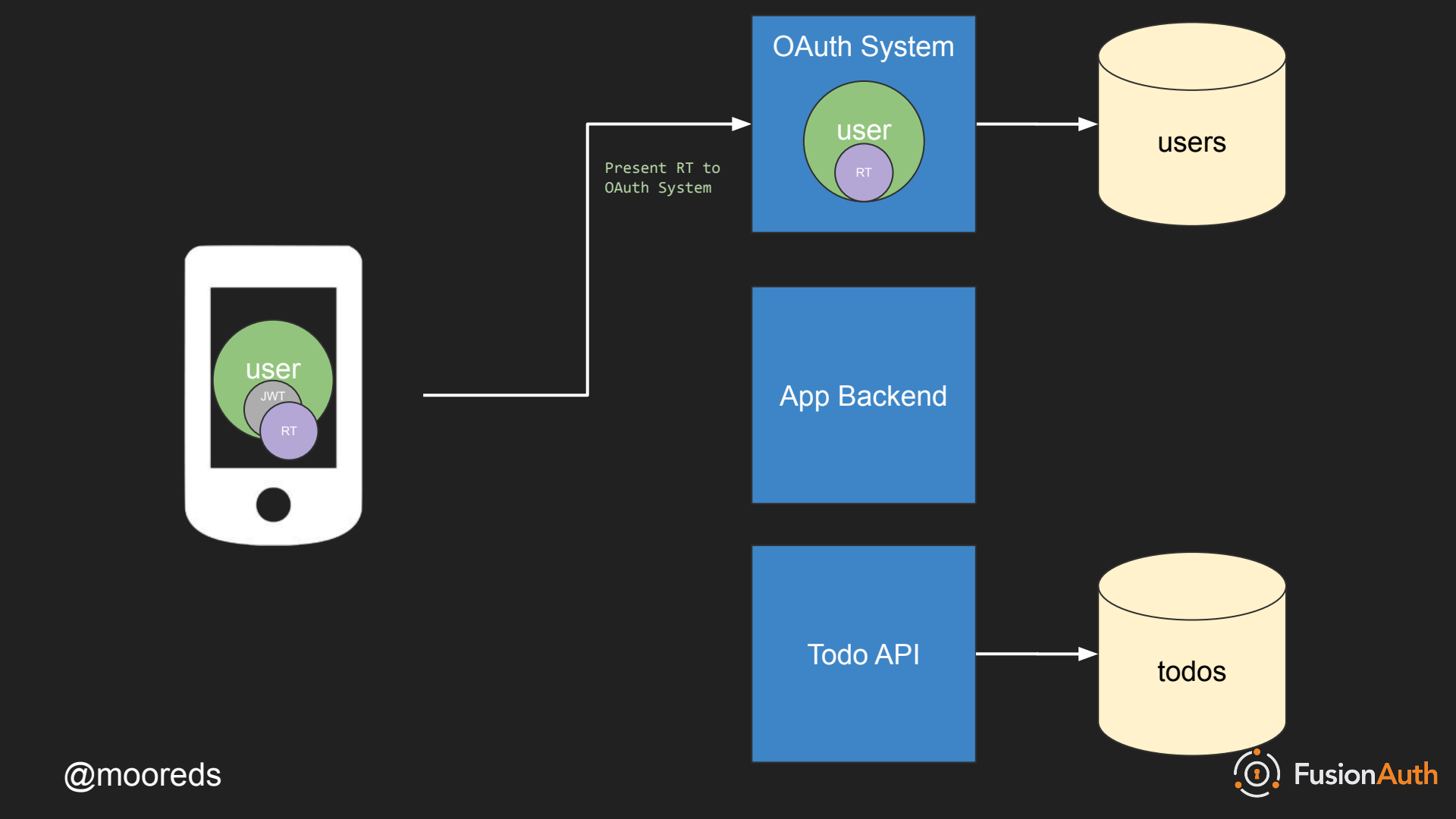
Why Refresh Tokens

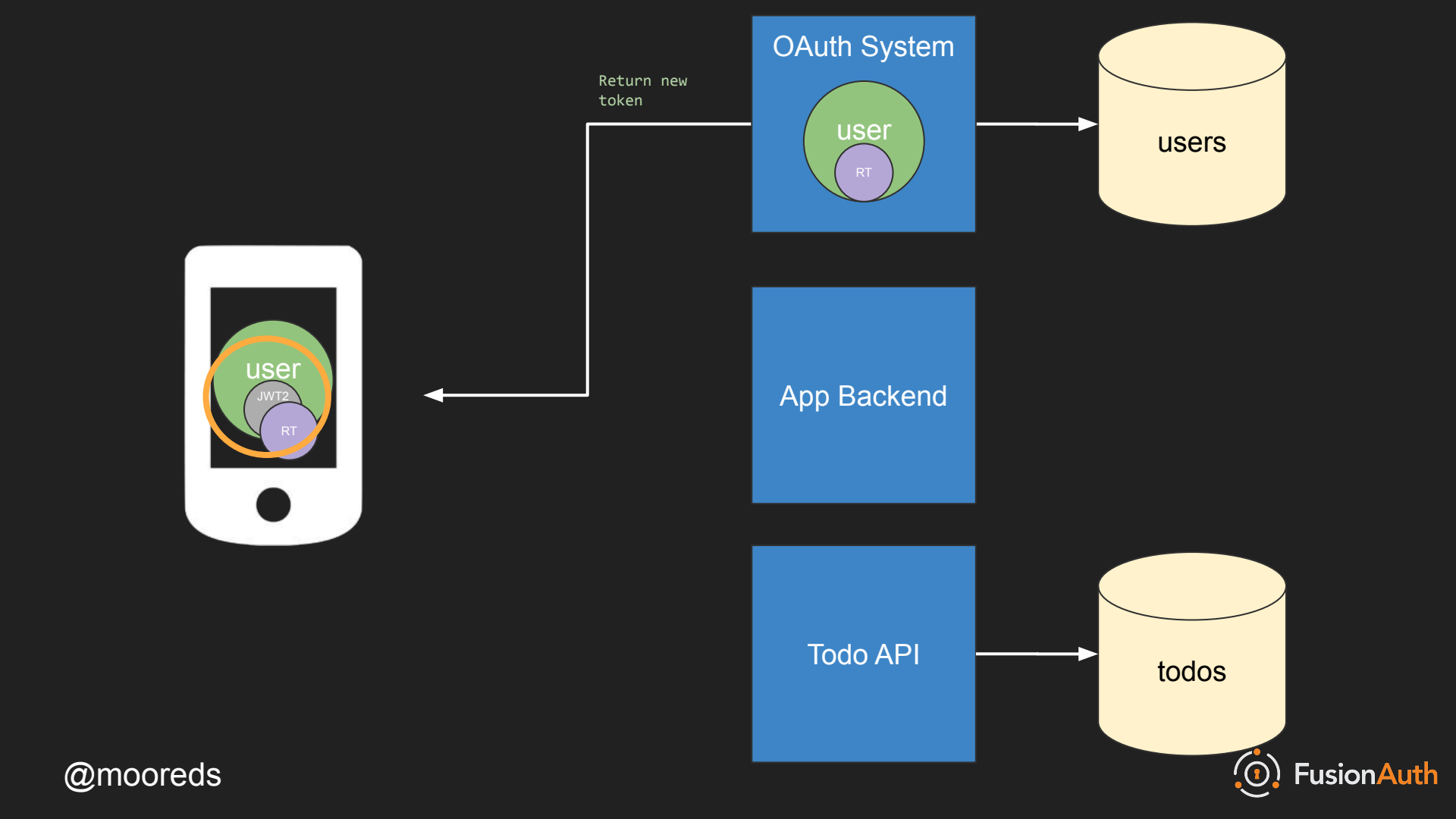
- Compromise between
  - User authenticating regularly
    - User hassle
  - Long lived tokens
    - Security risk

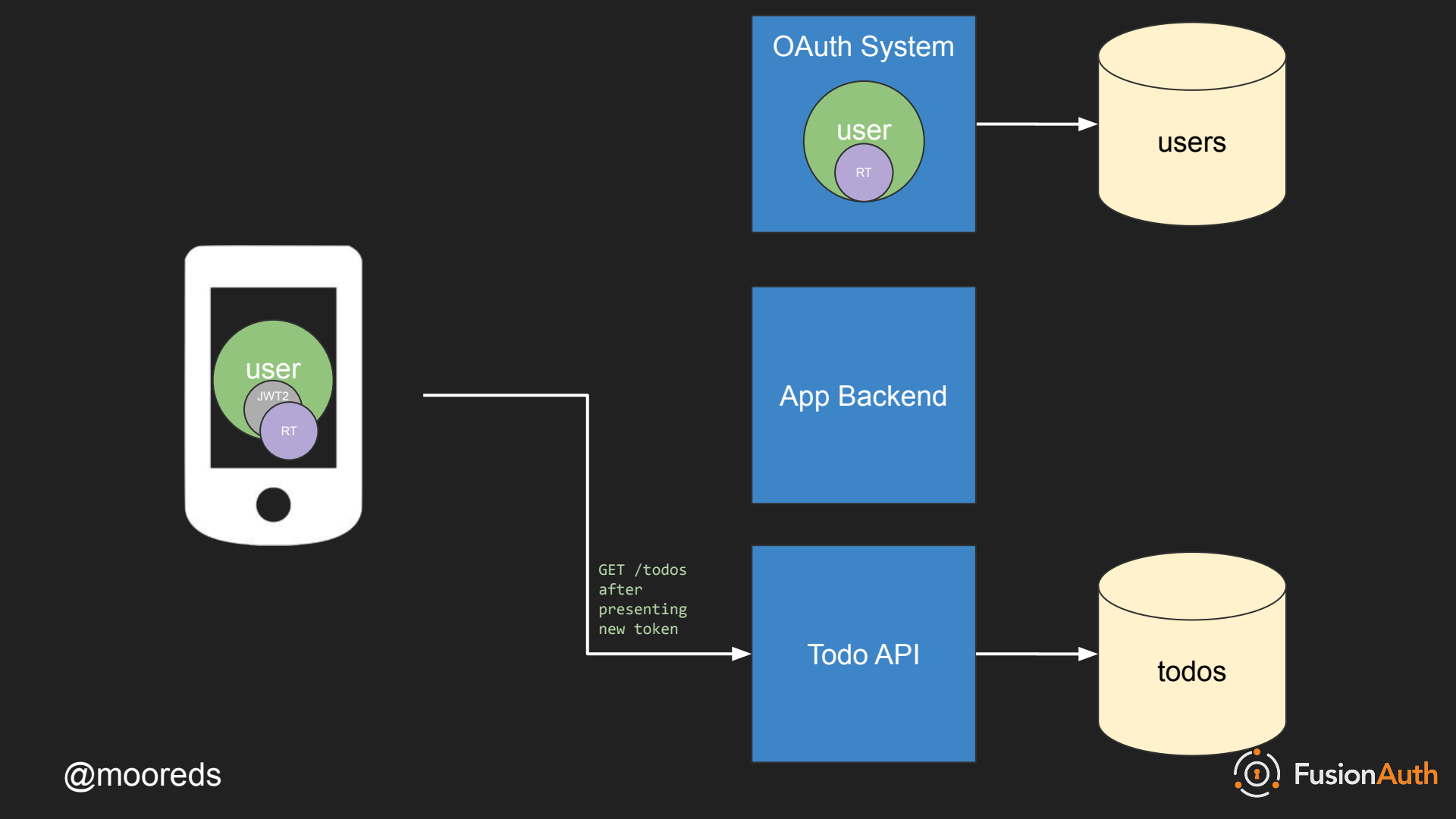

OAuth System
user
RT
users
user
JWT
RT
App Backend
GET /todos
after
presenting
token
Todo API
todos
@mooreds
FusionAuth


OAuth System
user
RT
users
user
JWT
RT
App Backend
Todo API
Todos as JSON
todos
@mooreds
FusionAuth


OAuth System
user
RT
users
user
JWT
RT
App Backend
Todo API
todos
@mooreds
FusionAuth


OAuth System
user
RT
users
Present RT to
OAuth System
user
JWT
RT
App Backend
Todo API
todos
@mooreds
FusionAuth


OAuth System
user
RT
users
Return new
token
user
JWT2
RT
App Backend
Todo API
todos
@mooreds
FusionAuth


OAuth System
user
RT
users
user
JWT2
RT
App Backend
GET /todos
after
presenting
new token
Todo API
todos
@mooreds
FusionAuth

Refresh Token Takeaway

- Client needs to know how to use

Refresh Token Takeaway

- Client needs to know how to use
  - Store it

Refresh Token Takeaway

- Client needs to know how to use
  - Store it
  - Catch the access denied

Refresh Token Takeaway

- Client needs to know how to use
  - Store it
  - Catch the access denied
  - Present it to the OAuth server

Refresh Token Takeaway

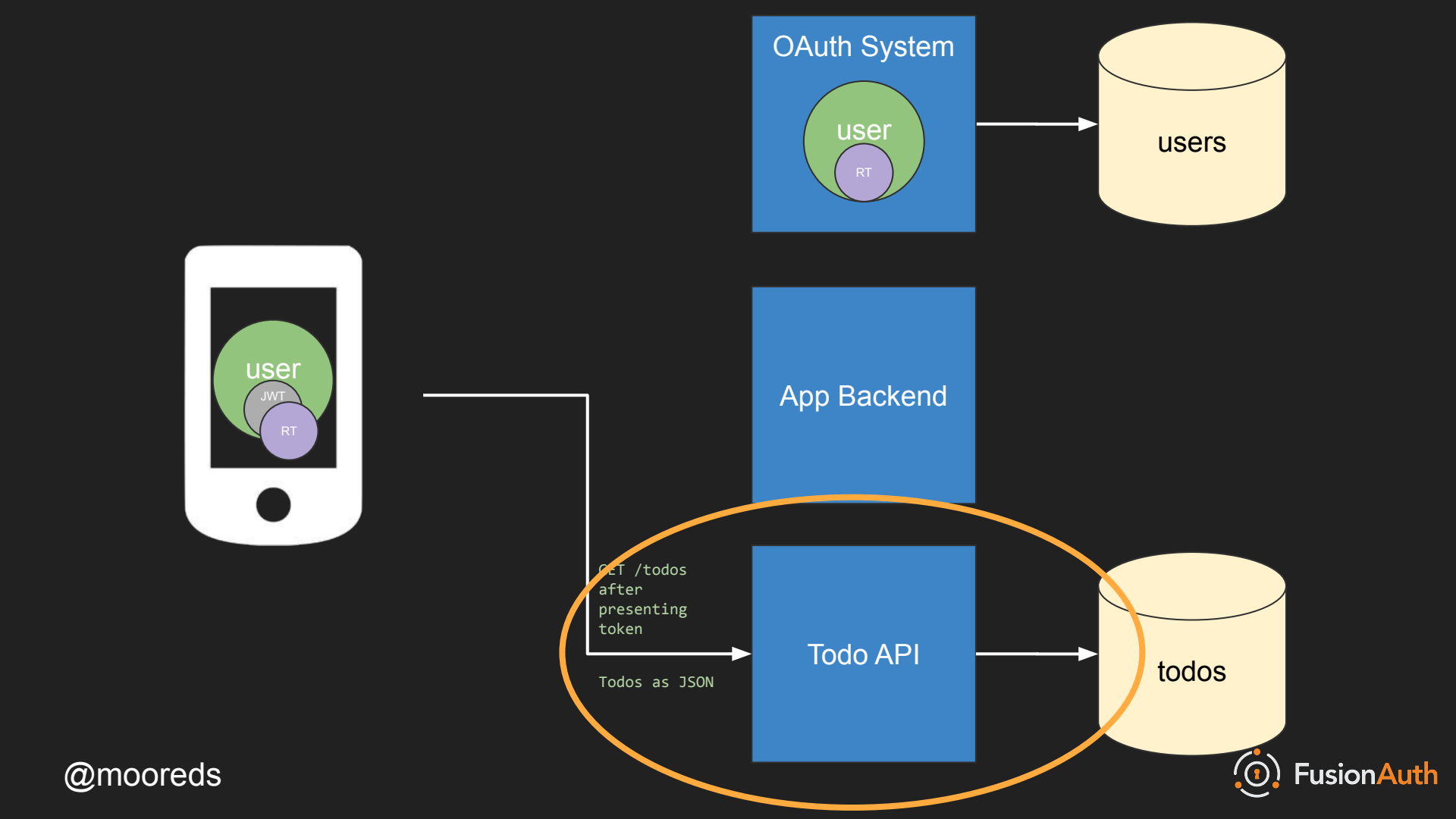
- Client needs to know how to use
  - Store it
  - Catch the access denied
  - Present it to the OAuth server
  - Store new access token

Refresh Token Takeaway

- Client needs to know how to use
  - Store it
  - Catch the access denied
  - Present it to the OAuth server
  - Store new access token
- No effort required of the consumer

Questions?

Consumer Concerns


OAuth System
user
RT
users
user
JWT
RT
App Backend
GET /todos
after
presenting
token
Todos as JSON
Todo API
todos
@mooreds
FusionAuth

What Is a Consumer

What Is a Consumer

- Accepts the access token, returns valuable data
  - Needs to validate the token

What Is a Consumer

- Accepts the access token, returns valuable data
  - Needs to validate the token
- The Todo API

What Is a Consumer

- Accepts the access token, returns valuable data
  - Needs to validate the token
- The Todo API
- Also called
  - Resource Server
  - RS

Two Options

- Examine access token
- Introspect access token

Examine JWT

- Only works if your access token is a JWT

Examine JWT

- Only works if your access token is a JWT
- Validate signature

Examine JWT

- Only works if your access token is a JWT
- Validate signature
- Validate claims

JWT

eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJmdXNpb25hdXRoLml
vIiwiZXhwIjoxNjE5NTU1MDE4LCJhdWQiOiIyMzhkNDc5My03MGRlLTQxODMtO
TcwNy00OGVkOGVjZDE5ZDkiLCJzdWIiOiIxOTAxNmI3My0zZmZhLTRiMjYtODB
kOC1hYTkyODc3Mzg2NzciLCJuYW1lIjoiRGFuIE1vb3JlIiwibWVtYmVyc2hpc
EV4cGlyZWQiOmZhbHNlLCJyb2xlcyI6WyJSRVRSSUVWRV9UT0RPUyIsIkFETUl
OIl19.cPL36Al_8eT7YQVowIOruitxXb0n8w4DKaWVthfEwfc

JWT Header

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9
```

=

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

JWT Body

eyJpc3MiOiJmdXNpb25hdXRoLmlvIiwiZXhwIjoxNjE5NTU1MDE4LCJhdWQiOiIyMzhkNDc5My03MGRlLTQxODMtOTcwNy00OGVkOGVjZDE5ZDkiLCJzdWIiOiIxOTAxNmI3My0zZmZhLTRiMjYtODBkOC1hYTkyODc3Mzg2NzciLCJuYW1lIjoiRGFuIE1vb3JlIiwibWVtYmVyc2hpcEV4cGlyZWQiOmZhbHNlLCJyb2xlcyI6WyJSRVRSSUVWRV9UT0RPUyIsIkFETUlOIl19

=

```
{
  "iss": "fusionauth.io",
  "exp": 1619555018,
  "aud": "238d4793-70de-4183-9707-48ed8ecd19d9",
  "sub": "19016b73-3ffa-4b26-80d8-aa9287738677",
  "name": "Dan Moore",
  "roles": ["RETRIEVE_TODOS", "ADMIN"]
}
```

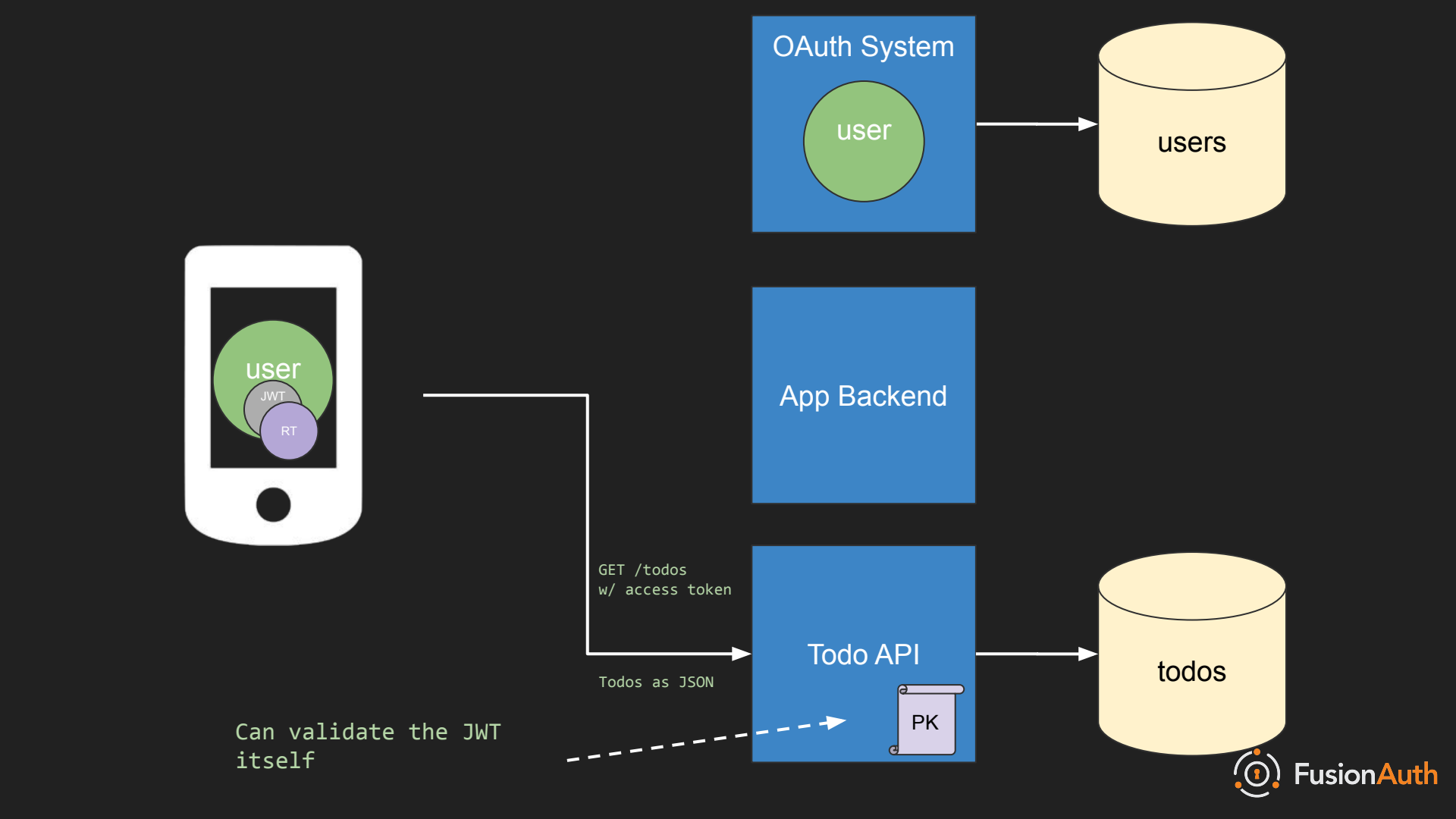
JWT Signature

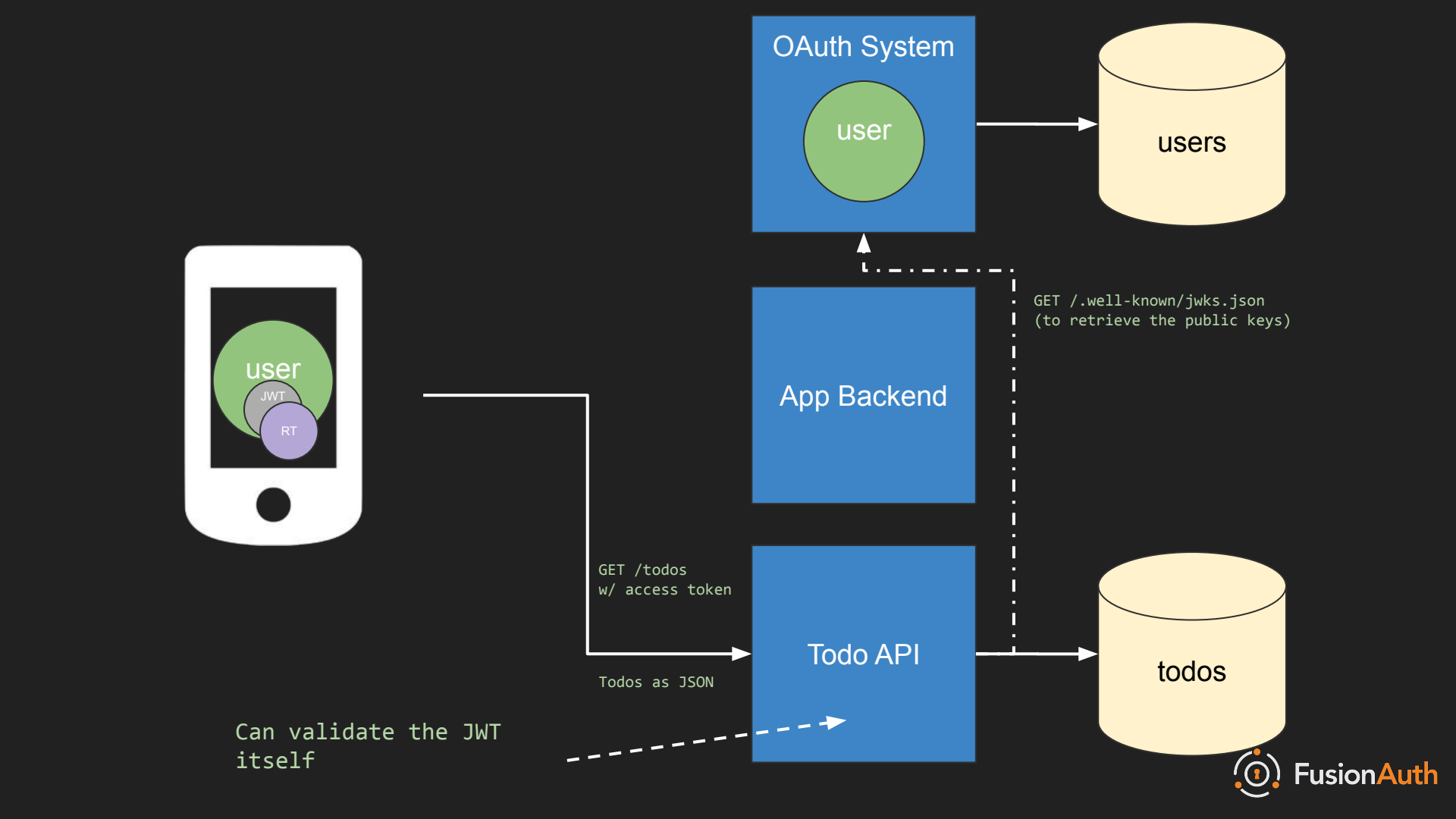
```
cPL36Al_8eT7YQVowIOruitxXb0n8w4DKaWVthfEwfc
```

=

Signature

Validate the Signature


OAuth System
user
users
user
JWT
RT
App Backend
GET /todos
w/ access token
Todo API
Todos as JSON
PK
todos
Can validate the JWT
itself
FusionAuth


OAuth System
user
users
user
JWT
RT
App Backend
GET /.well-known/jwks.json
(to retrieve the public keys)
GET /todos
w/ access token
Todo API
todos
Todos as JSON
Can validate the JWT
itself
FusionAuth

```
{
  "keys" : [
    {
      "alg" : "RS256",
      "e" : "AQAB",
      "kid" : "uk0PWf8KkTKiJqWwrNjn16QoKKI",
      "kty" : "RSA",
      "n" :
"2xzTUGNSpIeNcvICSlFlhver42dR9CWYePEoLk6ncuk1nKLzwOr-hy3W2rmkG-x_DaVMVBT5jimClL_k7Fu5xlscexpmNTo3lK_fqWv_qhn
OONSaX0ETqWSrS9MXWnJcPTZkA37ZAwhGKaz8zzSF3Jh_fULWnFHgJxCLBNYmopnvVAv_erJR0wjX9imMpMsBh3w8O6RyN8ghhlkJ0q4JKya
auF-xk8nLwIAvdWiPWNpzJ57oXfHUXesbLCfLIuM3f_suh_RaZn7uC0jEO1uG23htOqMb1M0TW5uk8pAxdhVKYbKsPR8CMOUc1je8wDo2w4y
4gY_1koST3xnA6IRhBQ",
      "use" : "sig",
      "x5c" : [
"MIICuDCCAaCgAwIBAQIQMcWR+VPwTieC06b3Fn6XbzANBgkqhkiG9w0BAQsFADAYMRYwFAYDVQQDEw1mdXNpb25hdXRoLmlvMB4XDTIxMDU
yNDE5NDU1OVoXDTMxMDUyNDE5NDU1OVowGDEWMBQGA1UEAxMNZnVzaW9uYXV0aC5pbzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggE
BANsc01BjUqSHjXLyAkpRZYb3q+NnUfQlmHjxKC5Op3LpNZyi88Dq/oct1tq5pBvsfw2lTFQU+Y4pgpS/5OxbucZbHHsaZjU6N5Sv36lr/6o
ZzjjUml9BE6lkq0vTF1pyXD02ZAN+2QMIRims/M80hdyYf31C1pxR4CcQiwTWJqKZ71QL/3qyUdMI1/YpjKTLAYd8PDukcjfIIYZZCdKuCSs
mmrhfsZPJy8CAL3Voj1jacyee6F3x1F3rGywnyyLjN3/7Lof0WmZ+7gtIxDtbhtt4bTqjG9TNE1ubpPKQMXYVSmGyrD0fAjDlHNY3vMA6NsO
MuIGP9ZKEk98ZwOiEYQUCAwEAATANBgkqhkiG9w0BAQsFAAOCAQEAcj/NIIltfhyP9zsleVn7N/QRavfKA1SBTwtlPMVezuRIX+S3jzxJb/o
t47TBD5WFNY5y5AOkWHQFNkVtuPjUYmKTAqJd0+kVur77tLKzour6wjOp2QgKzG3IGxQnK903JkFlyWF4vSJuOpH8WymJ1jq1gD5zJjz2NXq
ch+gBIp5Kscr2tj2hg2BGmq5v7+5pz2jYHosarj4sJwGsLqk1j479wK1iaMjdBVCMuq/QS9yF0sG0STsjbceyGzFWbknmZdfNup6C4a0m8pa
eb9mlgFfIx9qljuNInpc9QZWeRymNJ5spX0WYVRLu7ULrLDXr8xUupjCSMV95yI1XF6tMkw=="
      ],
      "x5t" : "uk0PWf8KkTKiJqWwrNjn16QoKKI",
      "x5t#S256" : "UW-4zdFS6YR9g4Vhl3T3xLwfQ_S1aLFh2x4VBvK1sbY"
    }
  ]
}
```

```
    {
        "alg" : "RS256",
        "kid" : "uk0PWf8KkTKiJqWwrNjn16QoKKI",
        "x5c" : [
"MIICuDCCAaCgAwIBAQIQMcWR+VPwTieC06b3Fn6XbzANBgkqhk
iG9w0BAQsFADAYMRYwFAYDVQQDEw1mdXNpb25hdX..."
        ]
        // ...
    }
```

JWT Header

```
{
  "typ": "JWT",
  "alg": "RS256",
  "kid": "uk0PWf8KkTKiJqWwrNjn16QoKKI"
}
```

If the Signature is Invalid

Stop

Validate Claims

Validating Claims

```
{
  "iss": "fusionauth.io",
  "exp": 1619555018,
  "aud": "238d4793-70de-4183-9707-48ed8ecd19d9",
  "sub": "19016b73-3ffa-4b26-80d8-aa9287738677",
  "name": "Dan Moore",
  "roles": ["RETRIEVE_TODOS", "ADMIN"]
}
```

Validating Claims

```
{
  "iss": "fusionauth.io",
  "exp": 1619555018,
  "aud": "238d4793-70de-4183-9707-48ed8ecd19d9",
  "sub": "19016b73-3ffa-4b26-80d8-aa9287738677",
  "name": "Dan Moore",
  "roles": ["RETRIEVE_TODOS", "ADMIN"]
}
```

Validating Claims

```
{
  "iss": "fusionauth.io",
  "exp": 1619555018,
  "aud": "238d4793-70de-4183-9707-48ed8ecd19d9",
  "sub": "19016b73-3ffa-4b26-80d8-aa9287738677",
  "name": "Dan Moore",
  "roles": ["RETRIEVE_TODOS", "ADMIN"]
}
```

Validating Claims

```
{
  "iss": "fusionauth.io",
  "exp": 1619555018,
  "aud": "238d4793-70de-4183-9707-48ed8ecd19d9",
  "sub": "19016b73-3ffa-4b26-80d8-aa9287738677",
  "name": "Dan Moore",
  "roles": ["RETRIEVE_TODOS", "ADMIN"]
}
```

Validating Claims

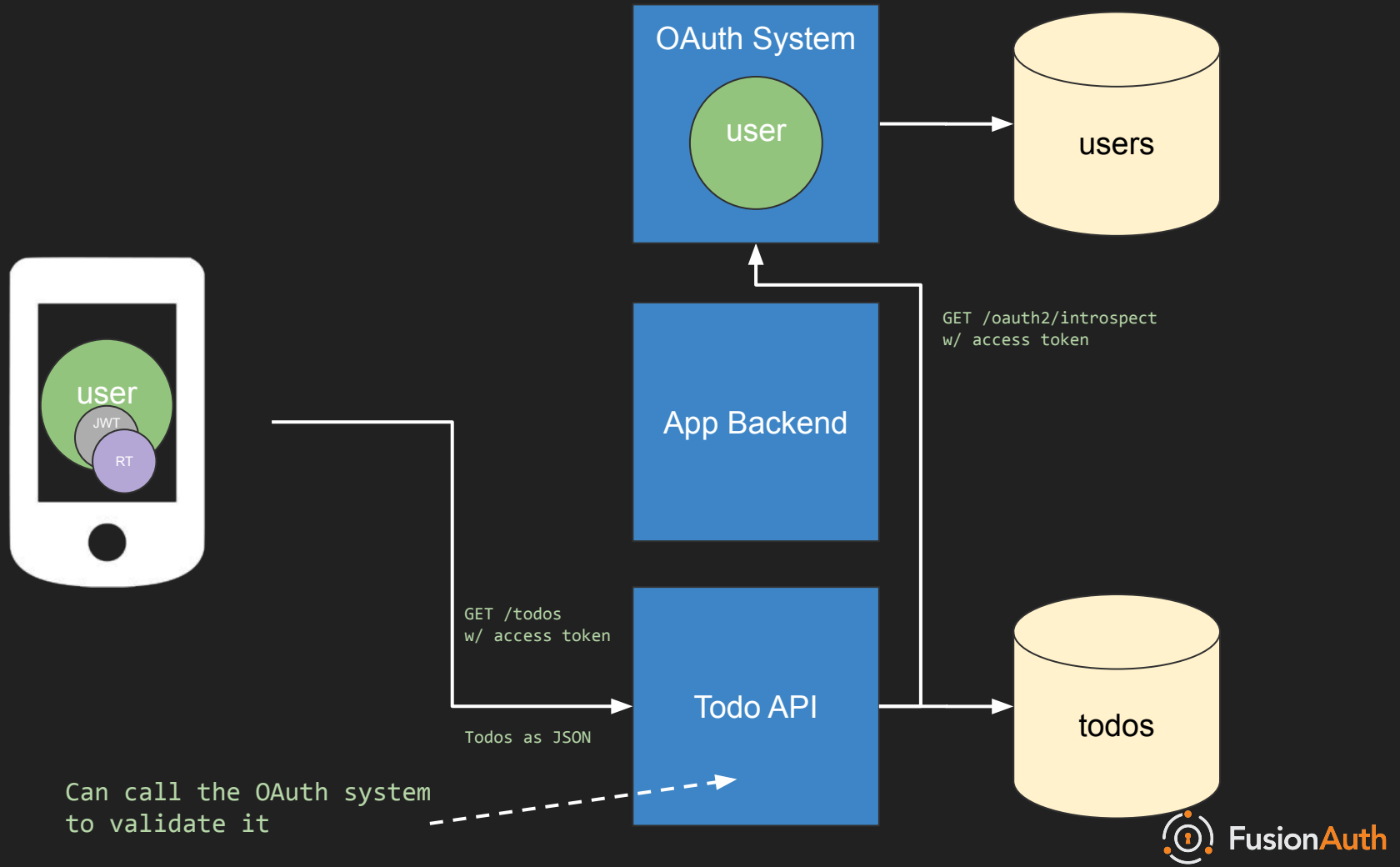
```
{
  "iss": "fusionauth.io",
  "exp": 1619555018,
  "aud": "238d4793-70de-4183-9707-48ed8ecd19d9",
  "sub": "19016b73-3ffa-4b26-80d8-aa9287738677",
  "name": "Dan Moore",
  "roles": ["RETRIEVE_TODOS", "ADMIN"]
}
```

Including "scp" claims

Validating Claims is Business Logic

Code!

Introspect


OAuth System
user
users
user
JWT
RT
App Backend
GET /oauth2/introspect
w/ access token
GET /todos
w/ access token
Todo API
todos
Todos as JSON
Can call the OAuth system
to validate it
FusionAuth

Introspect Results


```
{
  "iss": "fusionauth.io",
  "exp": 1619555018,
  "aud": "238d4793-70de-4183-9707-48ed8ecd19d9",
  "sub": "19016b73-3ffa-4b26-80d8-aa9287738677",
  "name": "Dan Moore",
  "roles": ["RETRIEVE_TODOS", "ADMIN"],
  "active": true
}
```

Processing Introspect Claims

- Defined in RFC7662; review your OAuth server docs

Processing Introspect Claims

- Defined in RFC7662; review your OAuth server docs
- Processed similar to JWT claims

Processing Introspect Claims

- Defined in RFC7662; review your OAuth server docs
- Processed similar to JWT claims
- Big difference:
  - Check “active” claim

Questions?

Common JWT Issues

JWT Footguns

- Lots of options
  - Use a library

JWT Footguns

- Lots of options
  - Use a library
- Can contain arbitrary JSON data
  - No secrets
  - Contents are base64 encoded


WHAT DO YOU NOT UNDERSTAND
THERE ARE NO SECRETS,
ONLY DELAYED REVELATIONS
imgflip.com

JWT Footguns

- Lots of options
  - Use a library
- Can contain arbitrary JSON data
  - No secrets
  - Contents are base64 encoded
- JWT specification allows a “none” algorithm
  - No signature is required

Unsanitized Credentials


OH WHY? THE HORROR!
THE HORROR!!

Simple Fix = Don’t Allow “none”

EVER!

JWT Tools

- https://fusionauth.io/learn/expert-advice/dev-tools/jwt-debugger
- https://fusionauth.io/docs/v1/tech/example-apps/
- https://fusionauth.io/learn/expert-advice/tokens/building-a-secure-jwt

Alternatives

API Keys

- Static
- Not necessarily time bound
- No encoded info

Proprietary Solutions

- AWS API Keys

Sessions

- No refresh
- Scale concerns

Conclusion

- On the client
  - Transmit tokens securely
  - Store tokens carefully
  - Handle refresh tokens
- In the consumer/resource
  - Validate your token
    - Check signature (or introspect)
    - Check claims

Thanks and More

- Mobile app feedback!
- fusionauth.io/ebook/
- Contact
  - Our booth
  - dan@fusionauth.io
  - FusionAuth.io
  - @mooreds