

Resources

- [RFC 6749 - OAuth 2.0](#)
- [RFC 6750 - Bearer Tokens](#)
- [RFC 7636 - Proof Key for Code Exchange](#)
- [OpenID Connect Specifications](#)
- [The OpenID Connect Handbook - Auth0](#)
- [Learn Identity Video Series - Auth0](#)

OAuth2 & OIDC

Stephanie Chamblee
@stephchamblee



Stephanie Chamblee

Software Developer at BrightLink (we're hiring!)
Auth0 Ambassador

stephaniechamblee.com

schamblee@thebrightlink.com

 [@stephchamblee](https://twitter.com/stephchamblee)





Overview

1

CONTEXT

Open Standards

Brief History of Identity

2

FOUNDATION

Four roles in OAuth

Tokens

Authorization Flows

3

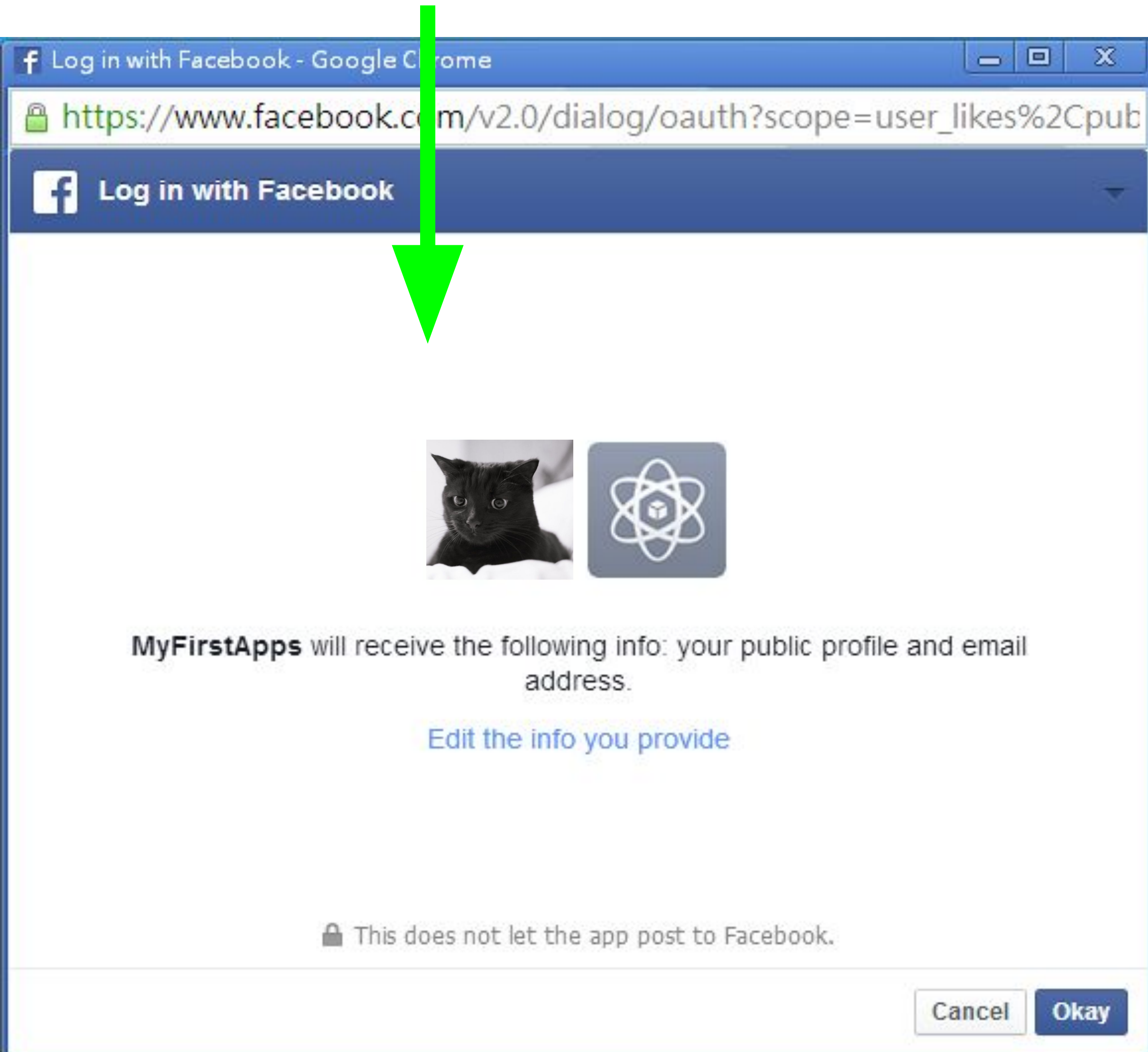
OAUTH & OIDC AUTHORIZATION CODE FLOW

OAuth 2.0 & OpenID Connect (OIDC) Walkthrough

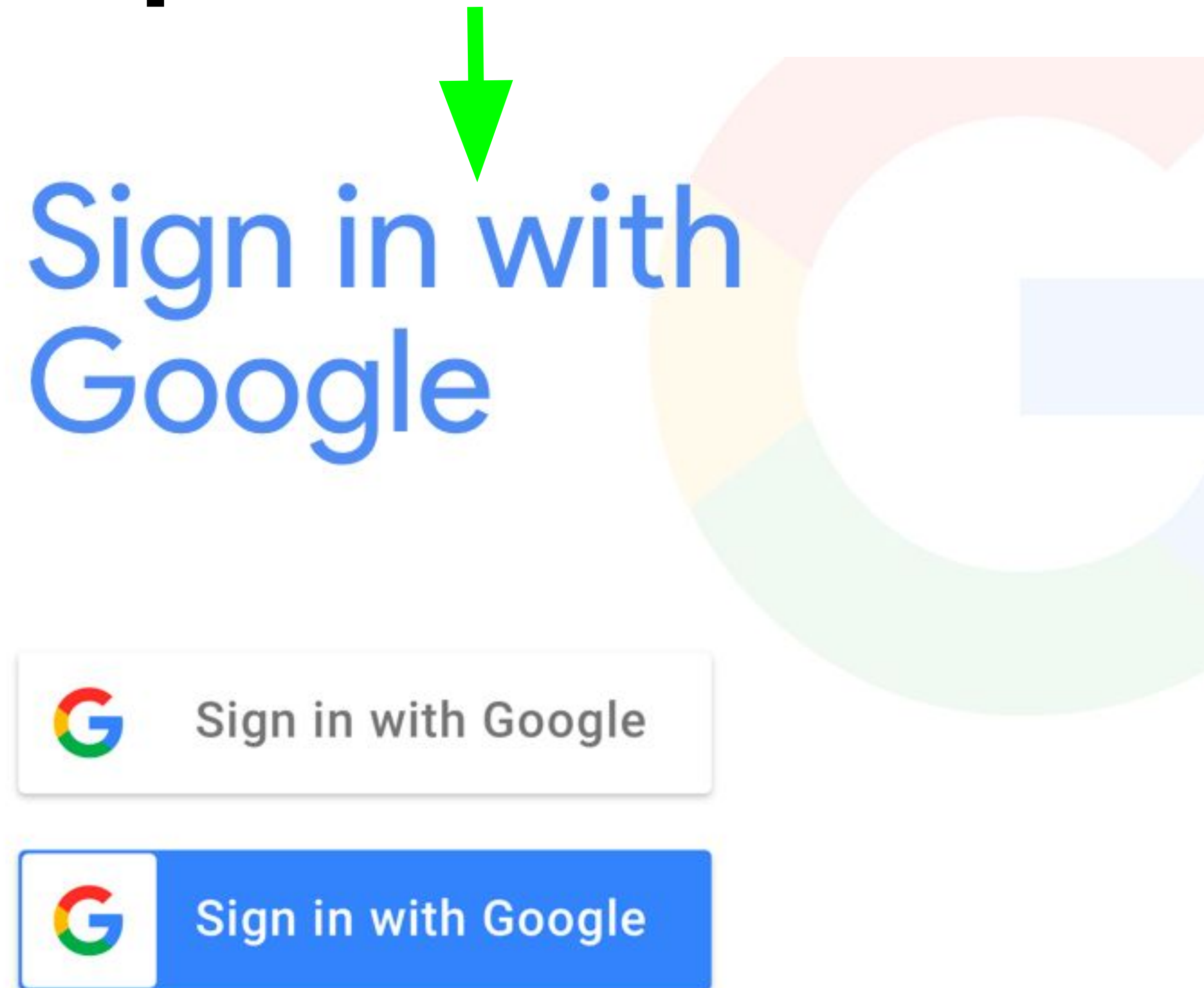


CONTEXT

OAuth2



OpenID Connect





Thanks!

Stephanie Chamblee
@stephchamblee

OPEN STANDARDS

Open Standards in Identity

* **SAML**

* **JWT**

* **OAuth2**

* **OIDC**

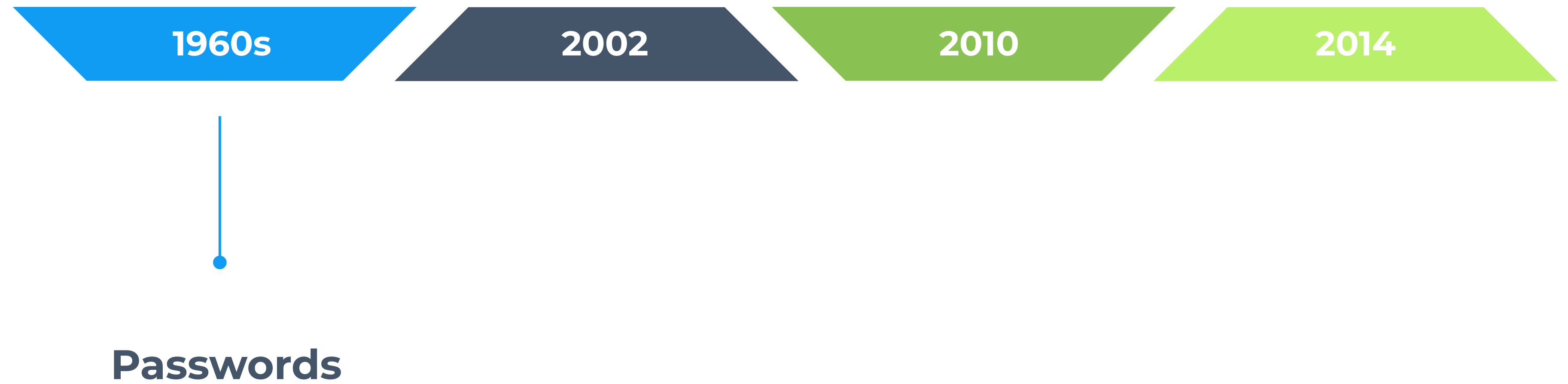
HISTORY OF IDENTITY

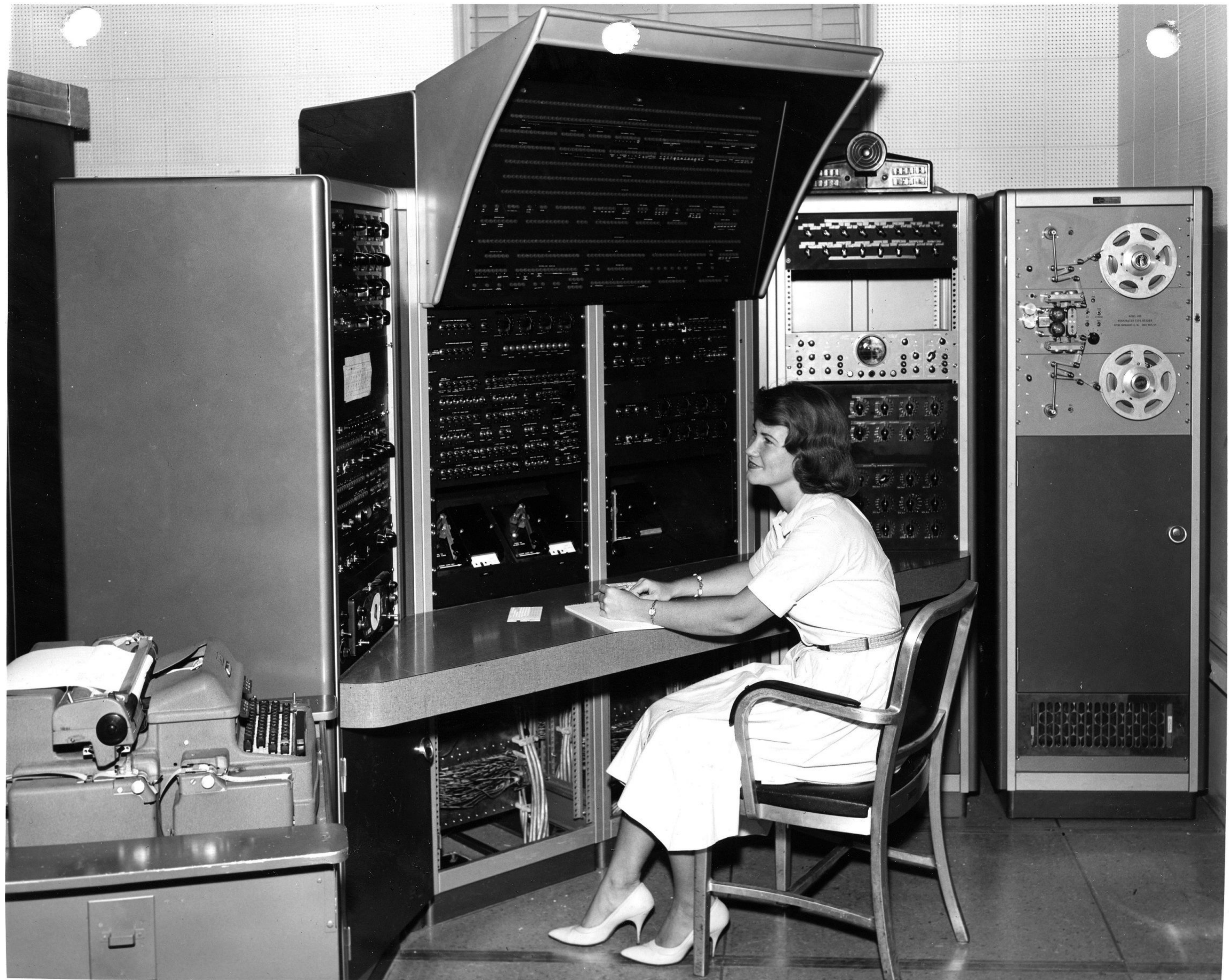






HISTORY OF IDENTITY






```
schamblée@sjc-ThinkPad-P52s:~/steph$ sudo vim cat_jokes.txt  
[sudo] password for schamblée: 
```

1



User

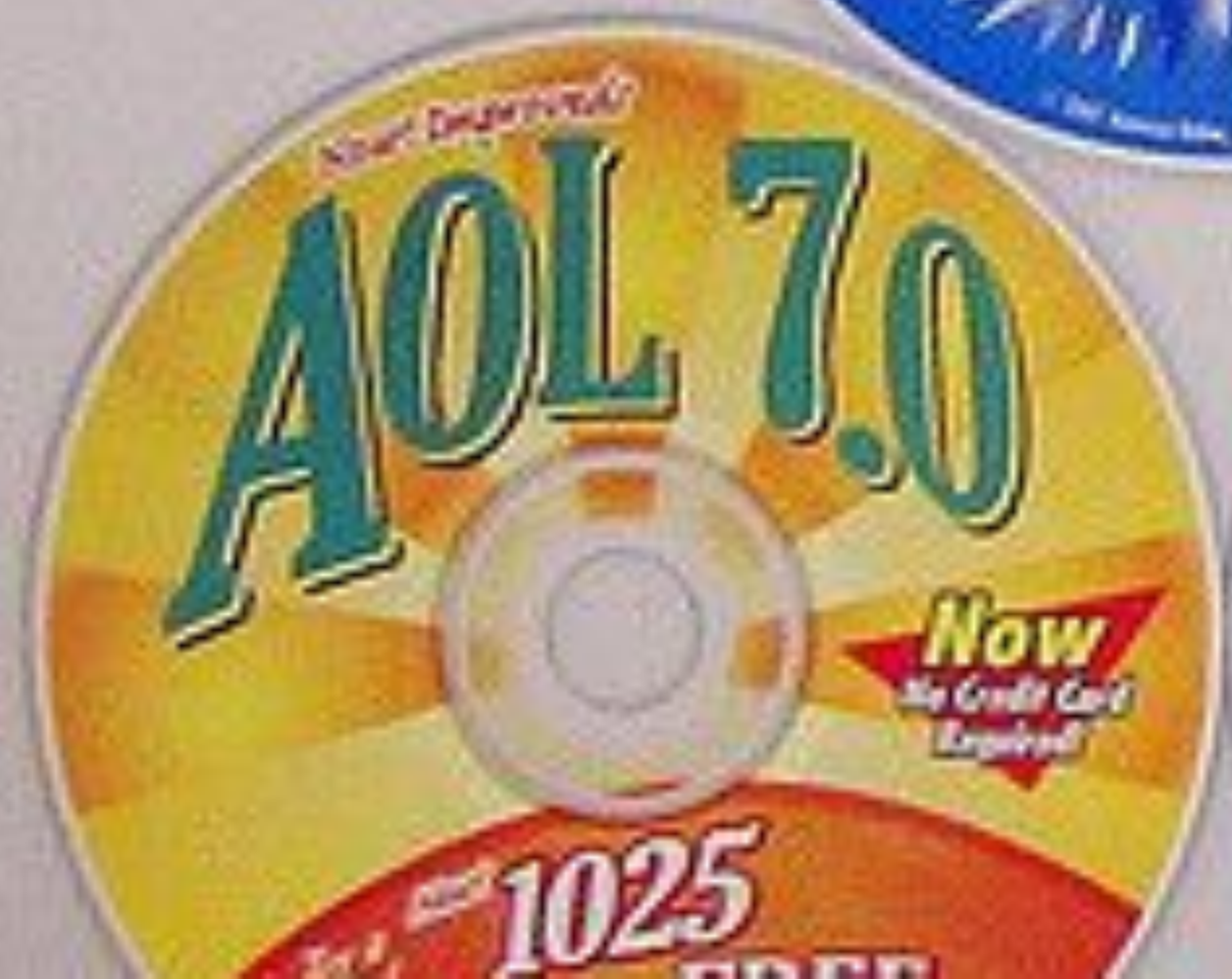


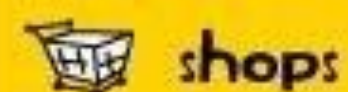
Password



Computer







login!

logout!

Search Neopets:

Go!



login.to.neopets

NeoClassicLogin

To log-in, please enter your username & password below.

Once you hit **Log In to Neopets**, you will be able to enjoy classic neopets.

Username

Your password

(It's cAsE sEnSitIvE!):

Log Into Neopets!

Don't have a Neopets Account?

Click the Grundo below to create one! Creating an account is easy and shouldn't take more than a couple of minutes.



Sign Up!

Simple Login Security

- * **Password Strength Requirements**
- * **Password Hashing**
- * **Two-Factor Authentication**

[Home](#)[Notify me](#)[Domain search](#)[Who's been pwned](#)[Passwords](#)[API](#)[About](#)[Donate ₿ !\[\]\(17413706fd4997a1a4bdf85c6864eee1_img.jpg\)](#)

Pwned Passwords

Pwned Passwords are 555,278,657 real world passwords previously exposed in data breaches. This exposure makes them unsuitable for ongoing use as they're at much greater risk of being used to take over other accounts. They're searchable online below as well as being downloadable for use in other online systems. [Read more about how HIBP protects the privacy of searched passwords.](#)

.....



pwned?

Oh no — pwned!

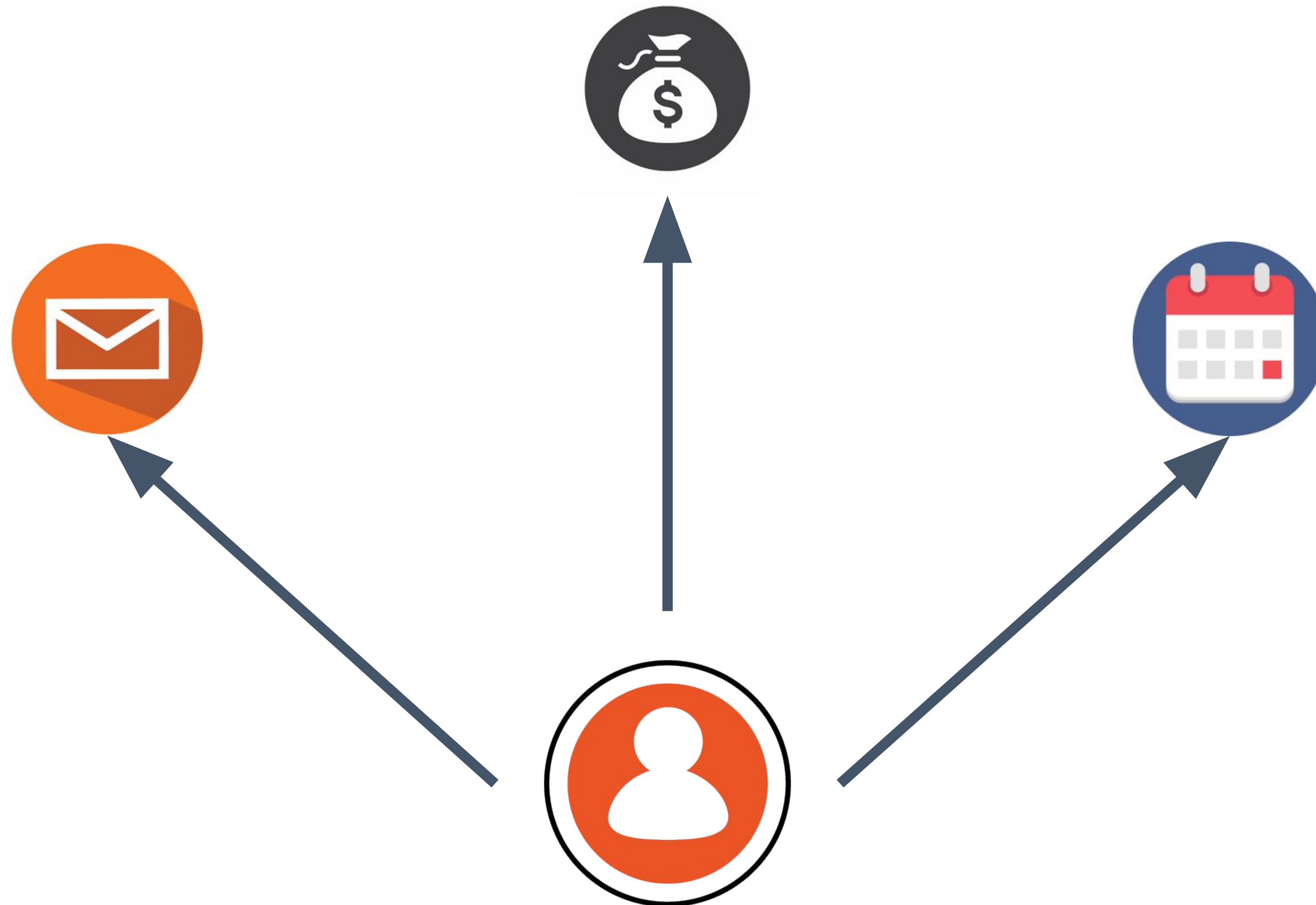
This password has been seen 3,730,471 times before

This password has previously appeared in a data breach and should never be used. If you've ever used it anywhere before, change it!

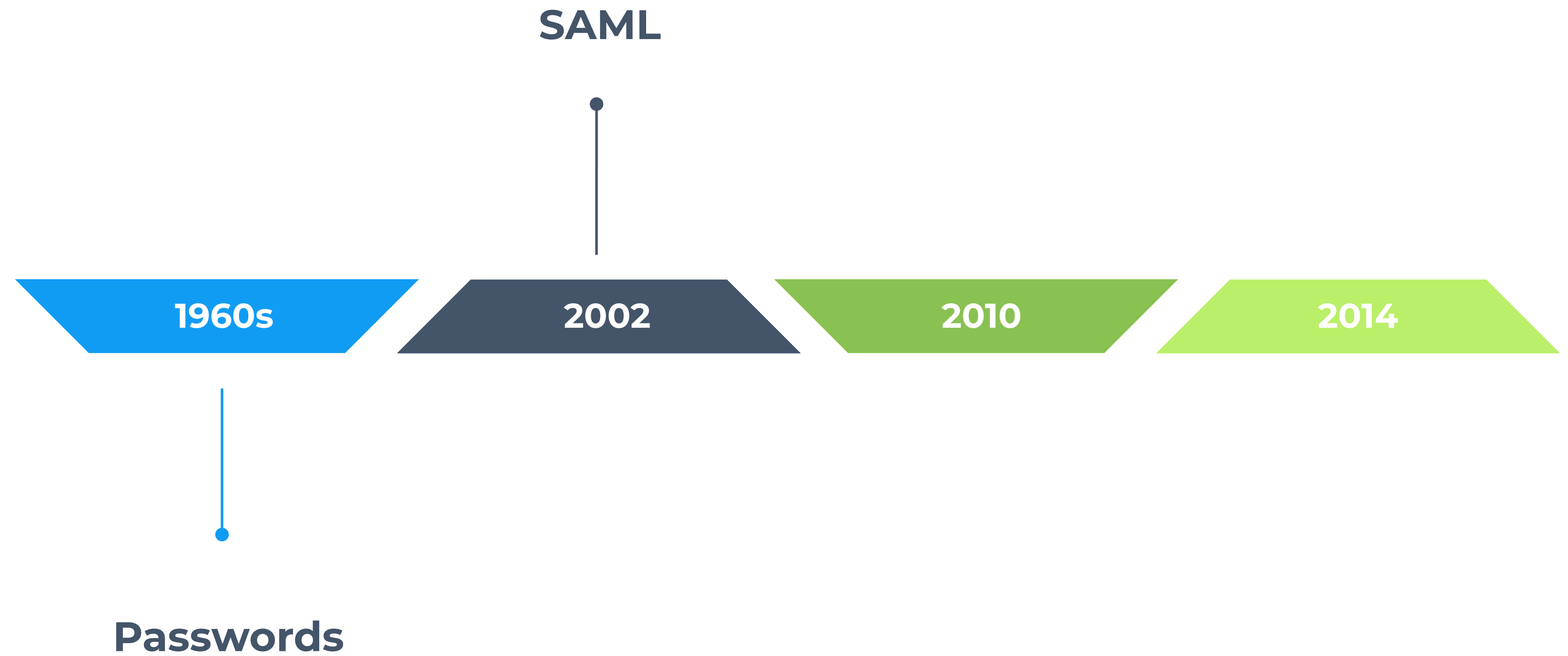
haveibeenpwned.com



ABC Company



HISTORY OF IDENTITY






home search global social net invite faq logout

[quick search](#)

[My Profile \[edit\]](#)
[My Groups](#)
[My Friends](#)
[My Messages](#)
[My Away Message](#)
[My Mobile Info](#)
[My Account](#)
[My Privacy](#)

Profile (This is you)

Scranton

Picture [edit]



[Visualize My Friends](#)
[Edit My Profile](#)
[My Account Preferences](#)
[My Privacy Preferences](#)

Information [edit]

Account Info:

Name:
Member Since: January 12, 2005
Last Update: February 3, 2005

Basic Info: [edit]

Email:
Status: Alumnus/Alumna
Sex: Male
Year: 2004
Concentration: Computing Sciences
Mathematics

Extended Info: [edit]

Phone:
High School:
Screenname:
Looking For: Friendship
Partner

Find your friends on Tagged!

Check your Gmail address book:

Gmail email address: @gmail.com

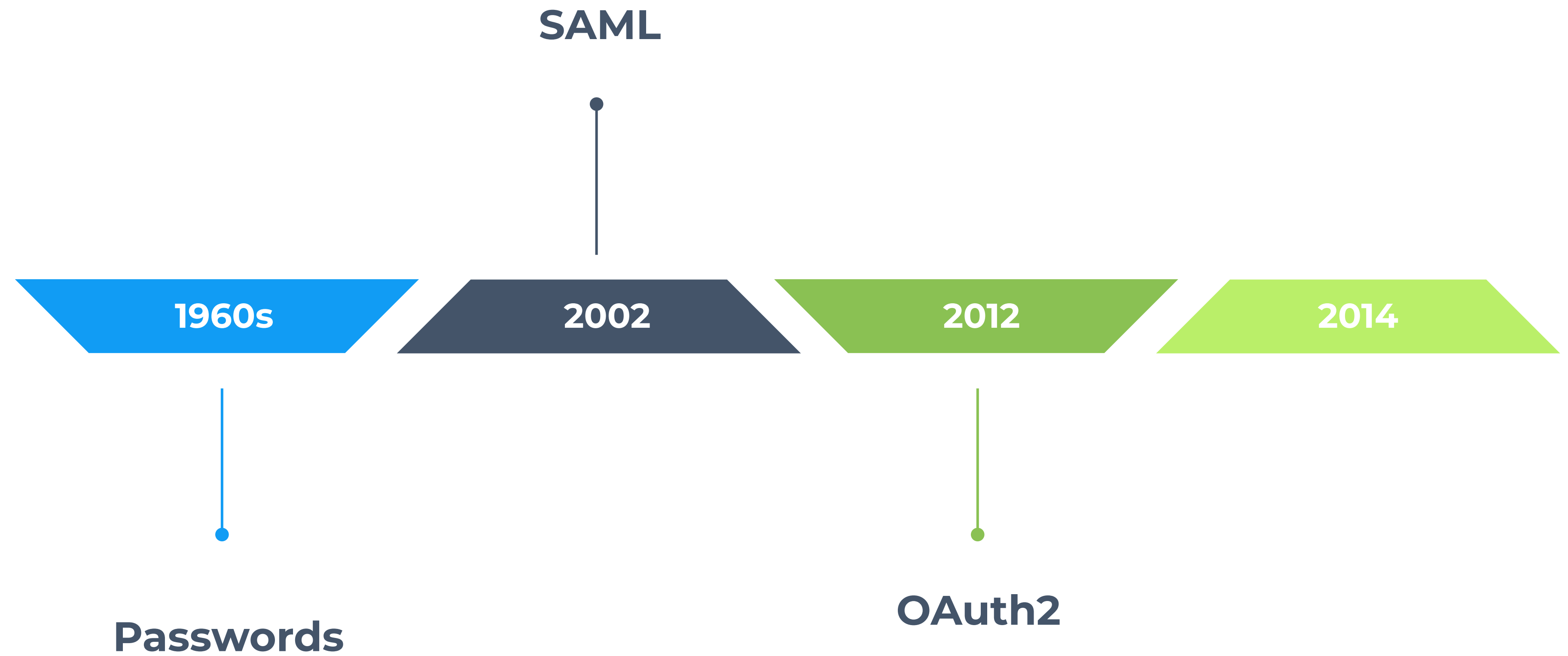
Gmail password:

Next >



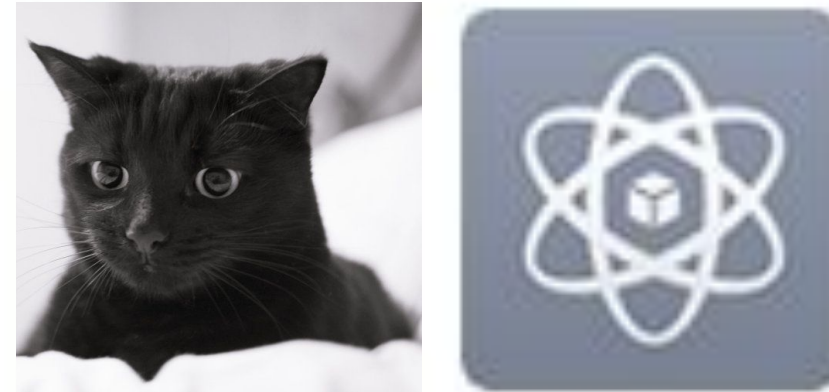
Enter your Gmail account name and password so we can match you up with your friends.

HISTORY OF IDENTITY



DELEGATED AUTHORIZATION

f Log in with Facebook



MyFirstApps will receive the following info: your public profile and email address.

[Edit the info you provide](#)

🔒 This does not let the app post to Facebook.

Cancel

Okay

Login to my account

If you have an account with us, please log in.

Email Address*

Password*

[Forgot your password?](#)

SIGN IN

OR



Sign in With Facebook



Sign in with Google

20% OFF First Order

New user registration

Email Address*

tracyyan12345@gmail.com

Re-enter password*

☐ Please send me emails about special offers, exclusives and promotions monthly.

☐ I agree to Rosewe.com [Terms and Conditions](#).

REGISTER

Authorization *vs.* Authentication

Authorization

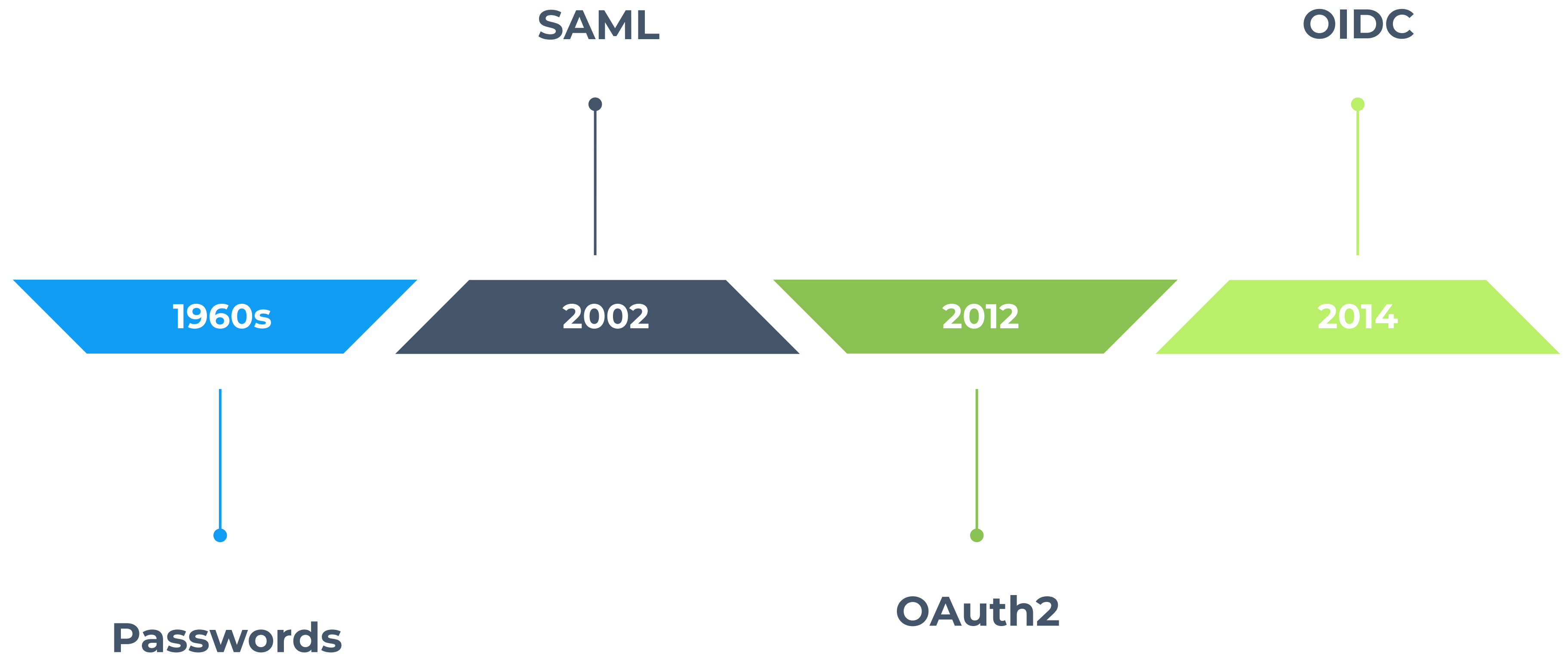


Authentication





HISTORY OF IDENTITY





CONTEXT SUMMARY

A large cruise ship is shown from a low angle, sailing on the ocean. The ship is white with multiple decks and a prominent funnel. The sky is a deep blue, and the water is calm, reflecting the ship and the sky. The text "CONTEXT SUMMARY" is overlaid in a large, white, sans-serif font across the middle of the image.



FOUNDATION

Four Roles Defined by OAuth2

OAuth 2.0 Roles

End User who
provides
consent for
scopes



Resource Owner (RO)

end user
scopes



OAuth 2.0 Roles

**End User who
provides
consent for
scopes**



**Resource
Owner (RO)**

**Resource
Server (RS)**



**API or
Application
controlling
the data**



Resource Server (RS)

Application Controlling
the data



▼ LinkedIn would like to:



View your email address



View your basic profile info



Manage your contacts



By clicking Accept, you allow this app and Google to use your information in accordance with their respective terms of service and privacy policies. You can change this and other [Account Permissions](#) at any time.

Cancel

Accept

OAuth 2.0 Roles

End User who
provides
consent for
scopes

Application
handling
delegated
authorization
decisions



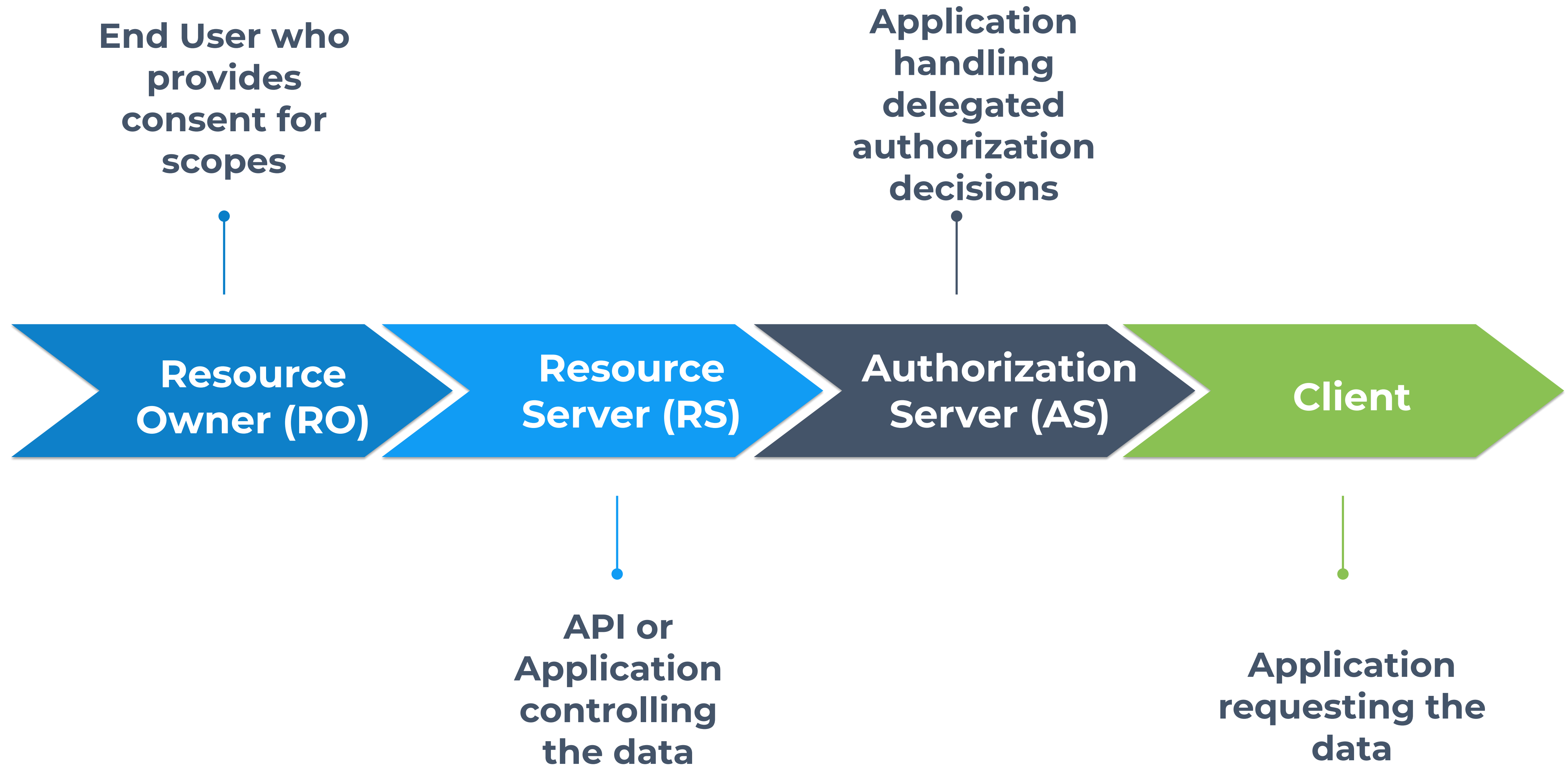
Resource
Owner (RO)

Resource
Server (RS)

Authorization
Server (AS)

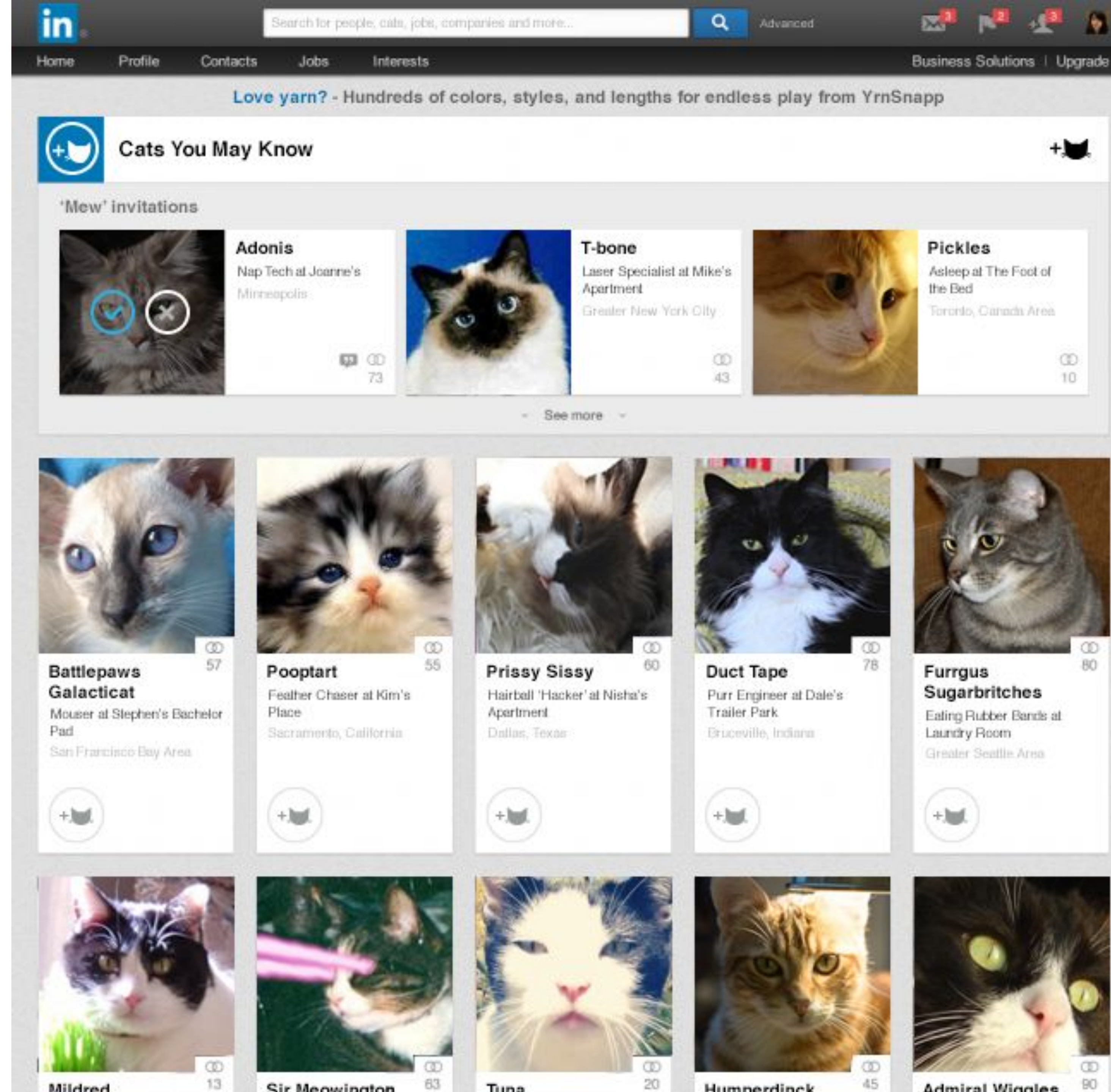
API or
Application
controlling
the data

OAuth 2.0 Roles



Client

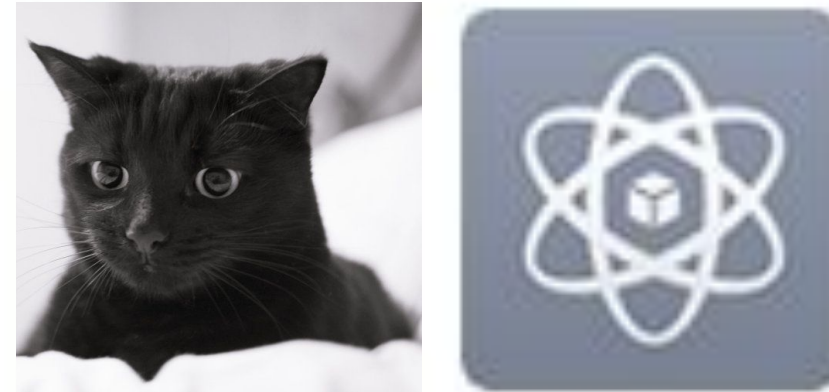
Application requesting
the data



A man in a dark suit, white shirt, and patterned tie is shown from the chest up, looking off to the right. He is holding a small, red and yellow wrapped candy in his right hand, near his mouth. The background is a solid, muted grey.

Let's review.

f Log in with Facebook



SomeApp will receive the following info: your public profile and email address.

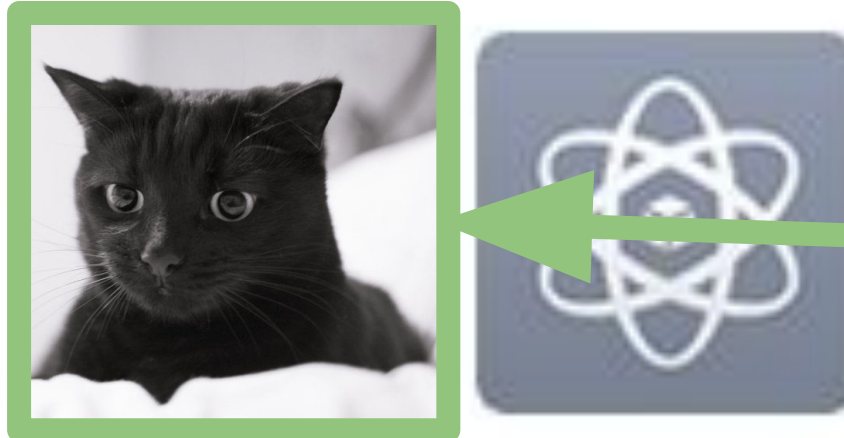
[Edit the info you provide](#)

🔒 This does not let the app post to Facebook.

Cancel

Okay

f Log in with Facebook



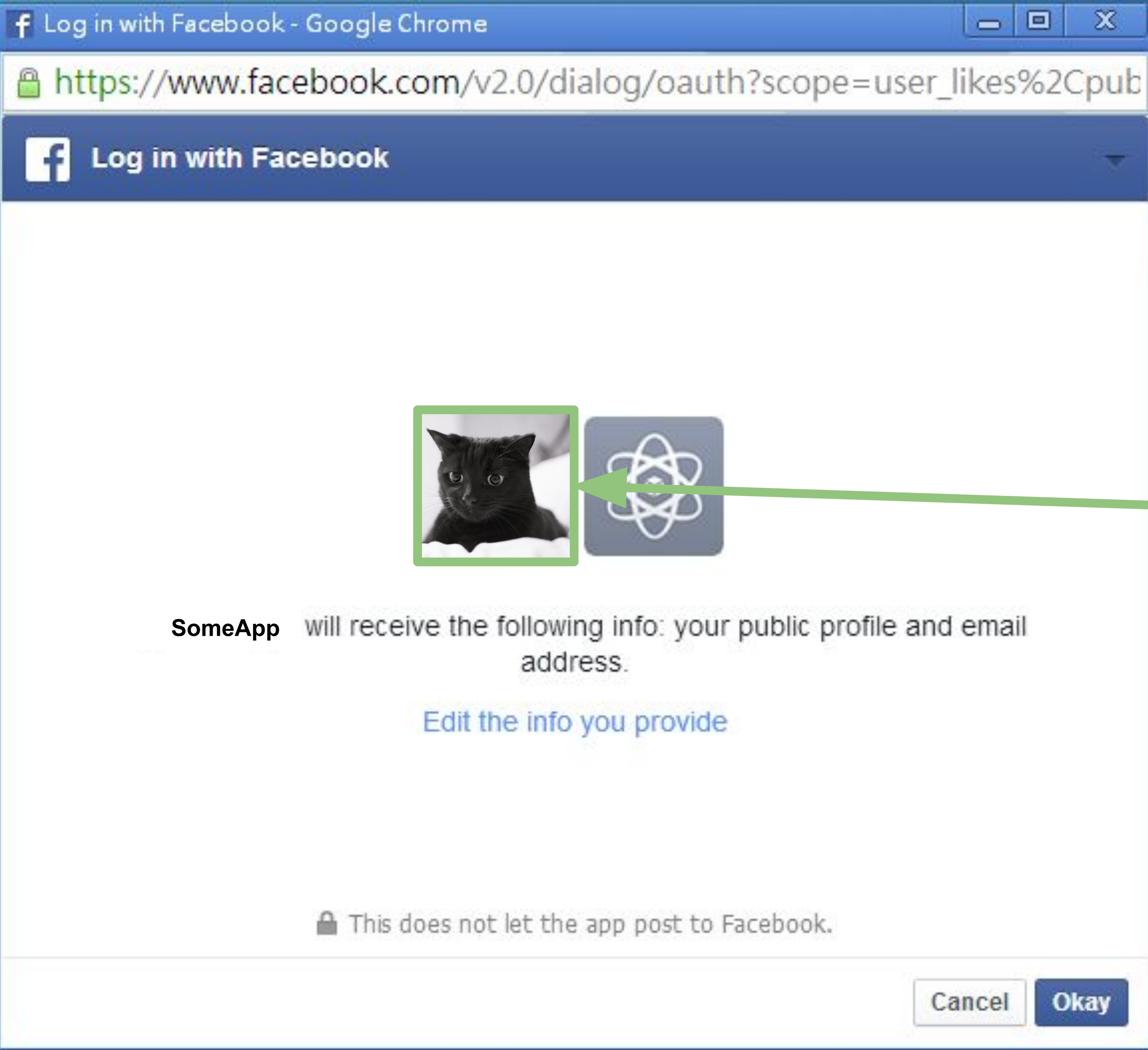
SomeApp will receive the following info: your public profile and email address.

[Edit the info you provide](#)


🔒 This does not let the app post to Facebook.

Cancel

Okay



**Resource
Owner**

 Log in with Facebook



SomeApp will receive the following info: your public profile and email address.

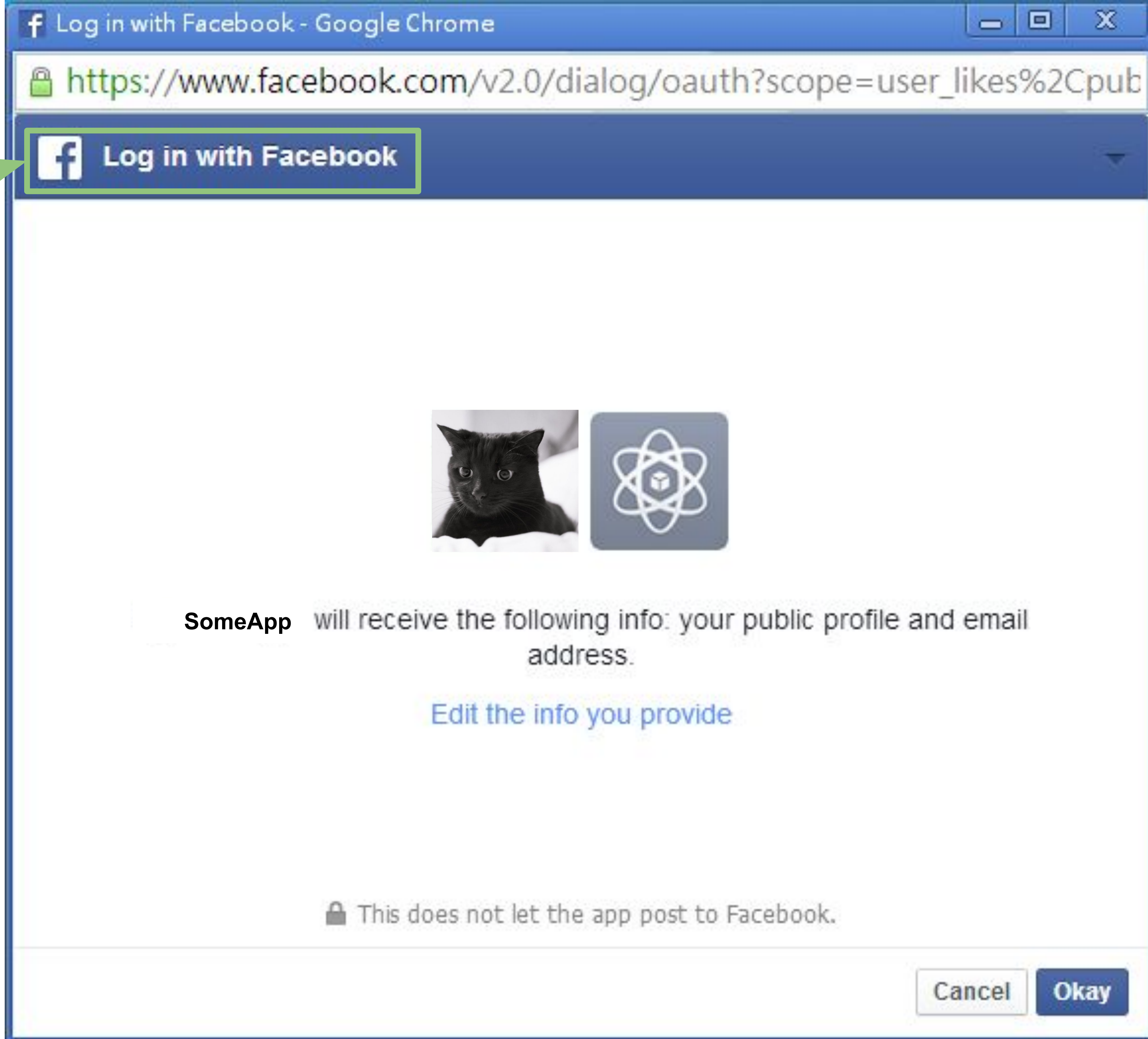
[Edit the info you provide](#)

 This does not let the app post to Facebook.

Cancel

Okay

**Resource
Server +
Authorization
Server**



f Log in with Facebook



SomeApp

will receive the following info: your public profile and email address.

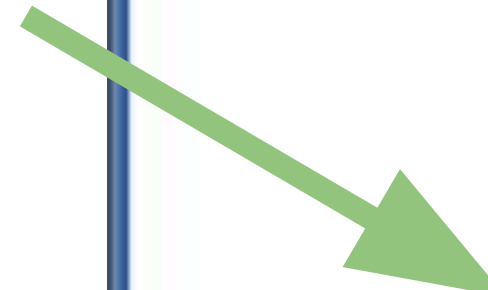
[Edit the info you provide](#)

🔒 This does not let the app post to Facebook.

Cancel

Okay

Client





SomeApp

Log in with Facebook - Google Chrome

https://www.facebook.com/v2.0/dialog/oauth?scope=user_likes%2Cpub


Log in with Facebook



SomeApp

will receive the following info: your public profile and email address.

Edit the info you provide

 This does not let the app post to Facebook.

Cancel

Okay

f Log in with Facebook



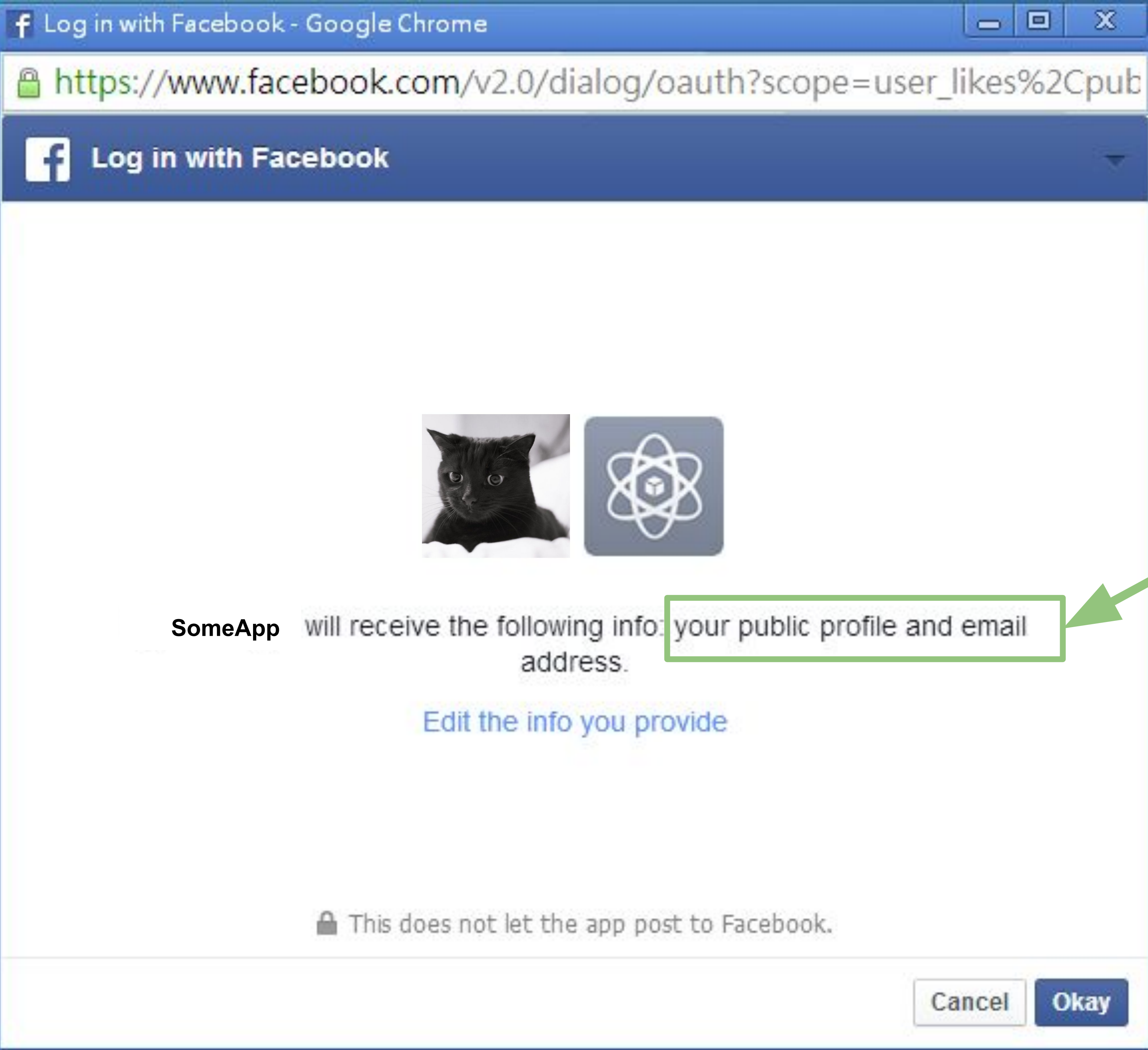
SomeApp will receive the following info: your public profile and email address.

[Edit the info you provide](#)

🔒 This does not let the app post to Facebook.

Cancel

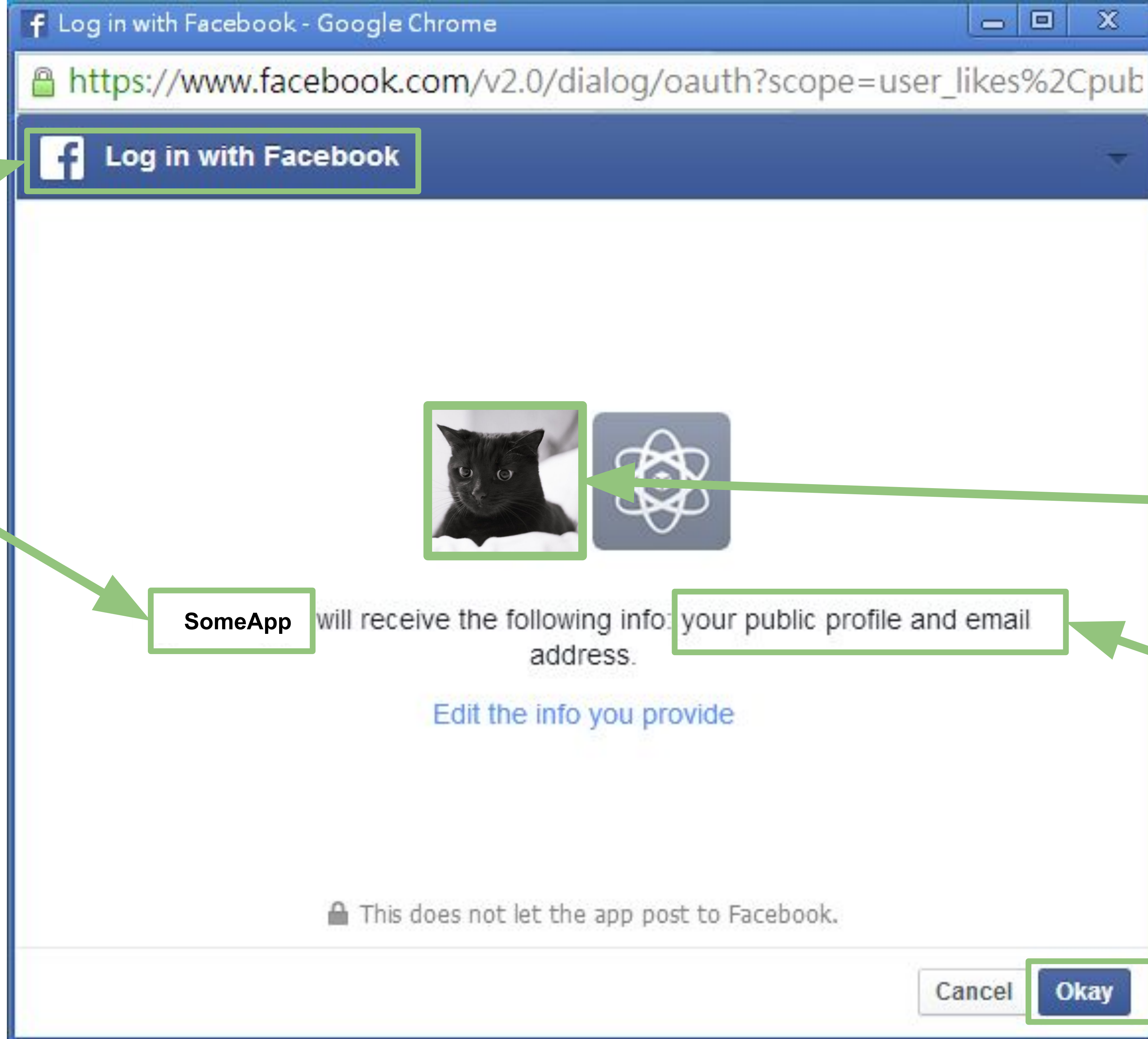
Okay



Scopes

**Resource
Server +
Authorization
Server**

Client



**Resource
Owner**

Scopes

Consent

Tokens



Tokens

Access Token

Refresh Token

ID Token

Access Token



Refresh Token



ID Token



ID Number

Document
Number

ID Token

JSON Web Token (JWT)



ID Number

Document
Number

JSON Web Token (JWT)

- **Encoded Claims (user data)**
 - **Stateless validation**
- **Signed for authenticity**

3 parts of JWT

Header

Payload

Signature

hhhhhhhhhh . ppppppppppppppppppp . sssssssssssssssssss

3 parts of JWT

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6ImlnZmVybnQ0IiwiaWF0IjoxNTE2MzI1ODUyLCJ1aWQiOiJhbnR5bWVudCJ9.rMAPKVY
oWb1hBAKQ1-qrZgMksb6FoS-ajM4b5p1bKUg

hhhhhhhhhh . ppppppppppppppppppppp . sssssssssssssssssssss

Header

Payload

Signature

HEADER

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


PAYLOAD

```
{  
  "iss": "https://example.com",  
  "sub": "123",  
  "aud": "some-client-id",  
  "exp": 1311281970,  
  "iat": 1311280970  
}
```


SIGNATURE

```
HMACSHA256(  
    base64UrlEncode(header) + "." +  
    base64UrlEncode(payload),  
    your-256-bit-secret  
)
```

☐ secret base64 encoded

Parts of a JWT Summary

Header - alg (algorithm) & type (JWT)

Payload - claims (data about the user)

Signature - uses payload, header and secret and specified algorithm verify that a token is authentic

Authorization Grants

Authorization Grants

methods for a client application to acquire an access token which represents a user's permission for the client to access their data

Authorization Grant Flows

- * **Authorization Code**
- * **Authorization Code + PKCE**
- * **Client Credentials**

Front-Channel

Browser to API

Not-so secure

Back-Channel

Server to API

Very Secure



Authorization Code Flow

Back Channel + Front Channel



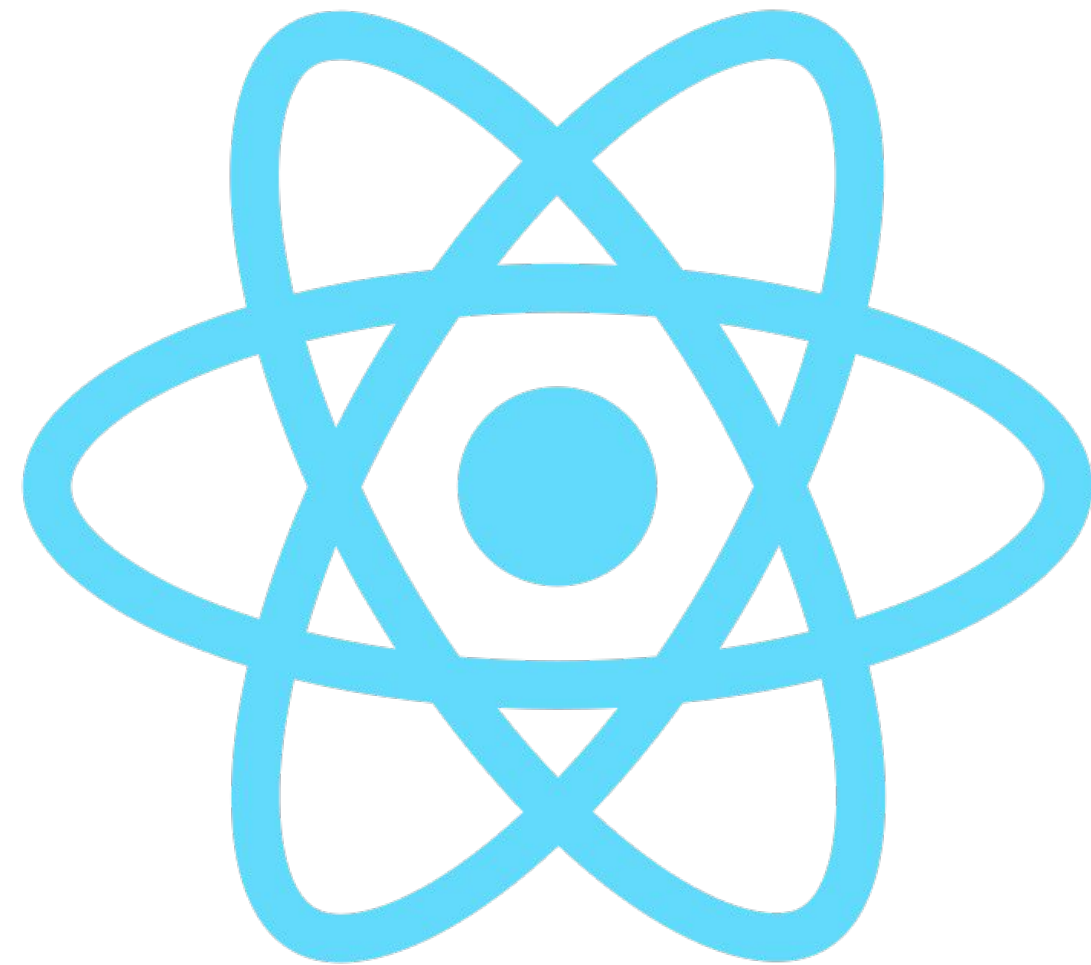
Implicit

Authorization Code + PKCE

Front Channel Only



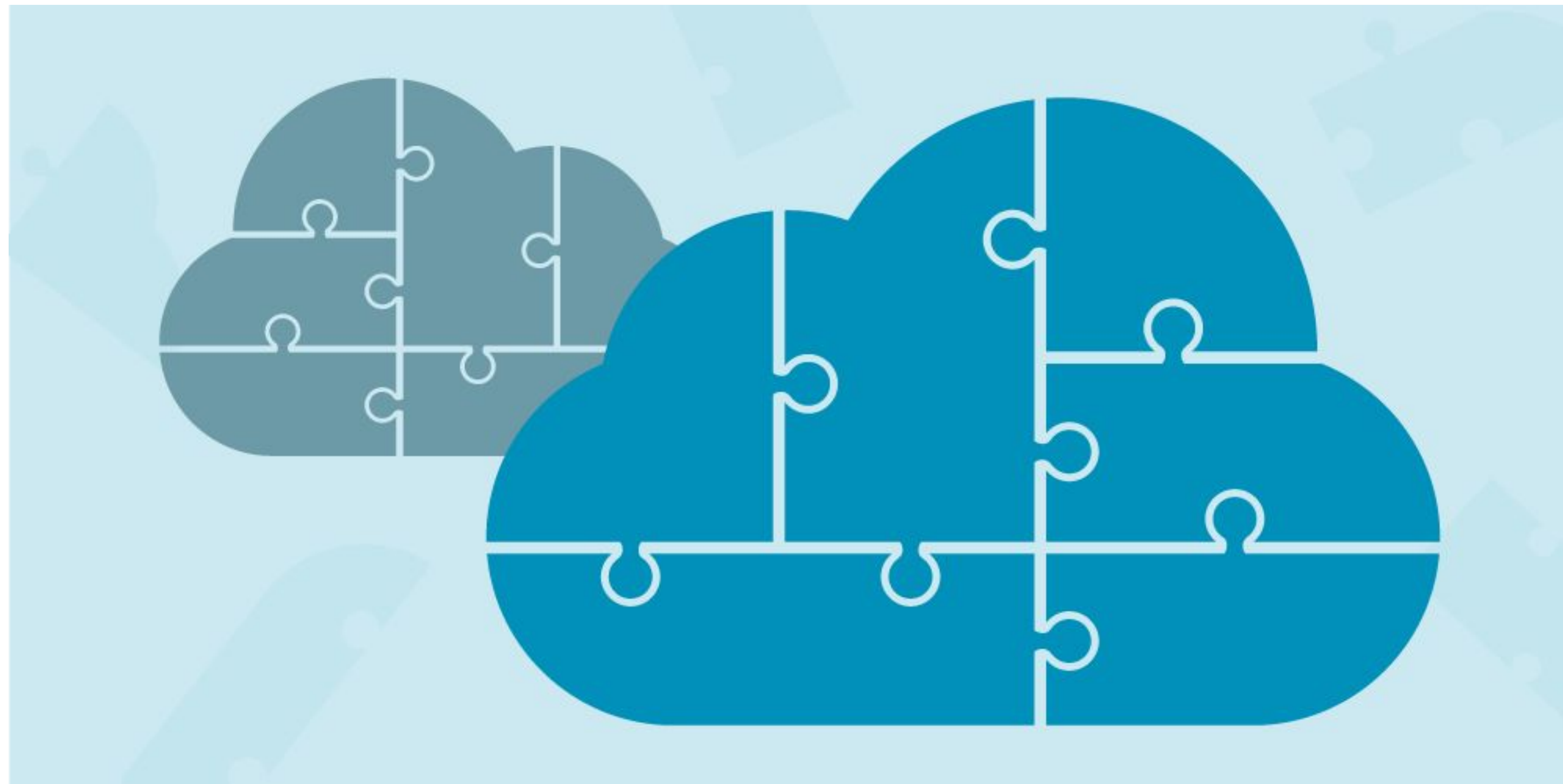
SPA/Mobile



Client Credentials Flow

Back Channel Only

Machine-to-Machine
example: microservices



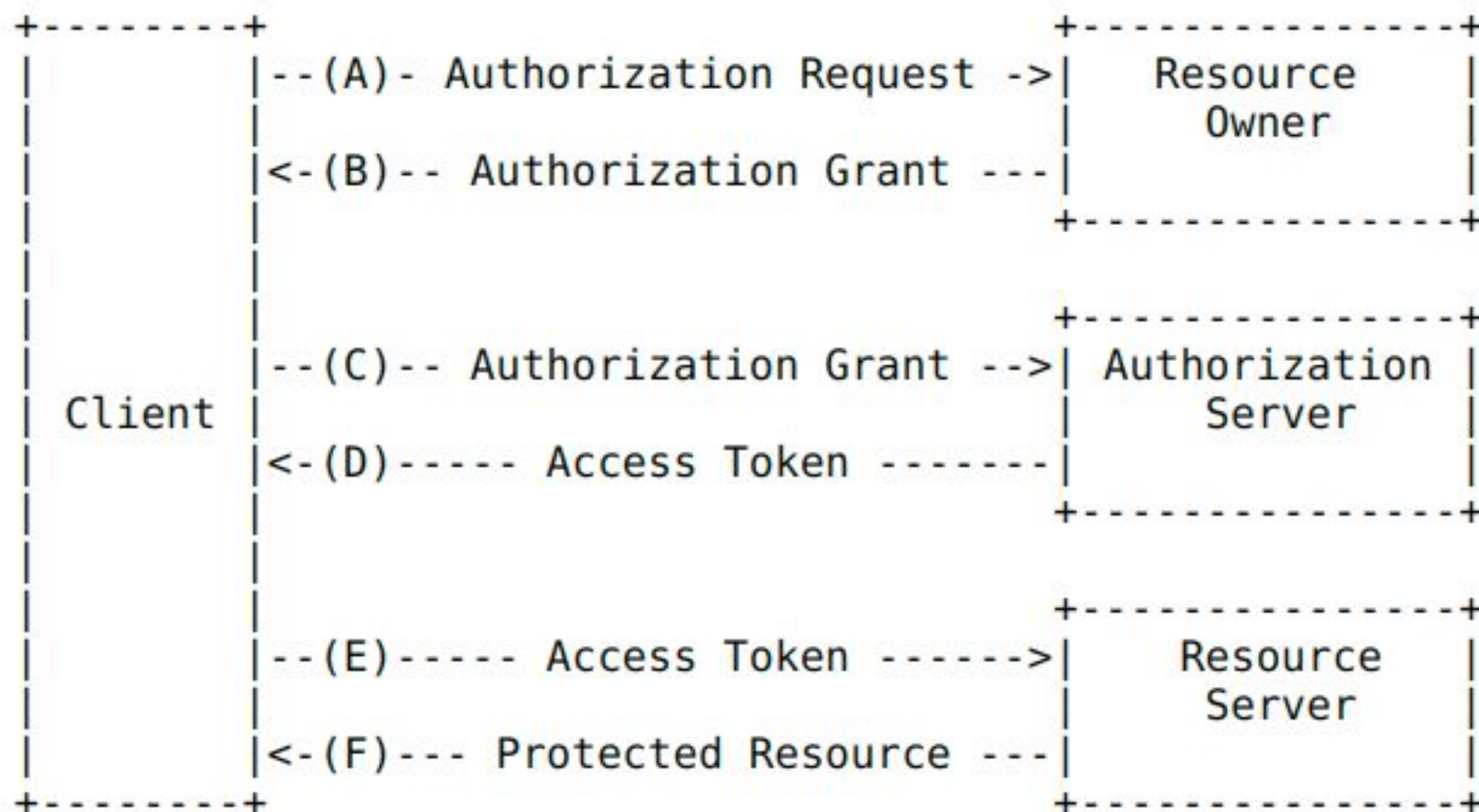
FOUNDATION SUMMARY





OAUTH & OIDC FLOW

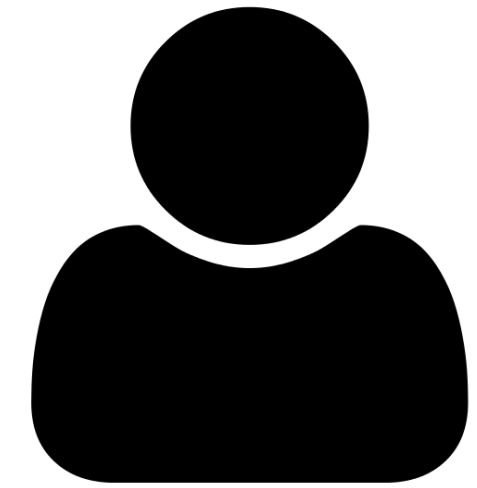
1.2. Protocol Flow



CLIENT

my app

RO User



AS Auth0



RS Google

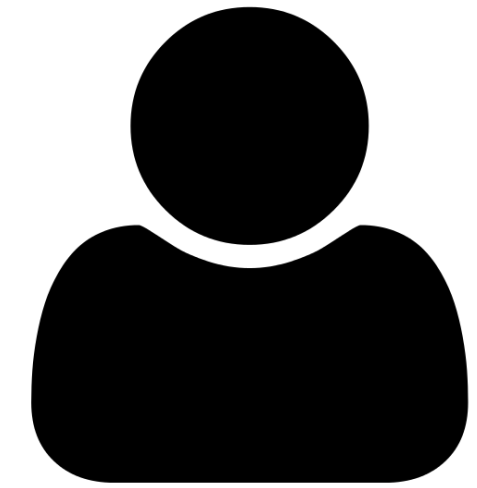


CLIENT

my app



RO User



AS Auth0



RS Google



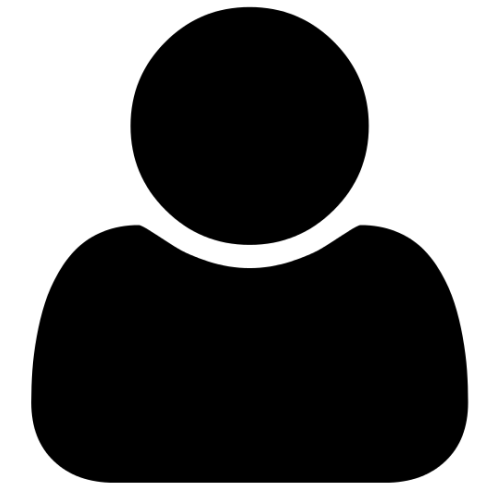
CLIENT

my app



Can I have authorization?

RO User



AS Auth0



RS Google



https://auth-server.com/authorize?

response_type=code&

client_id=client_id123&

redirect_uri=https://example.com/callback&

scope=openid+profile+email&

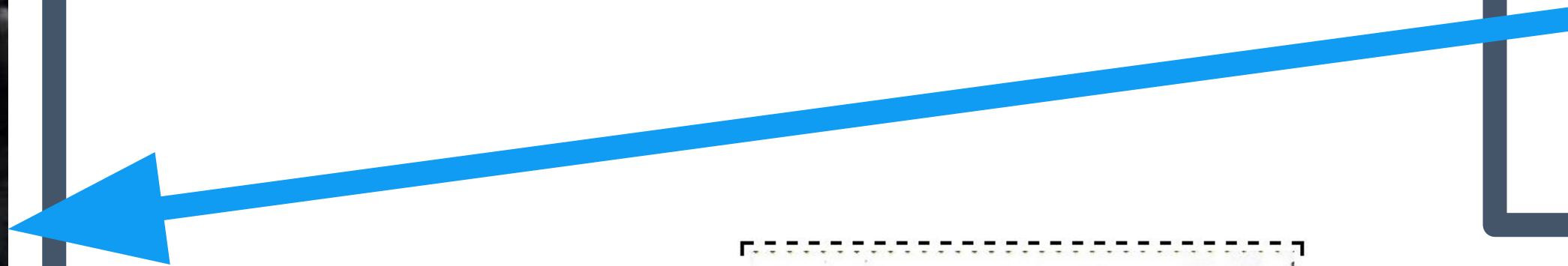
state=some_random_string

CLIENT

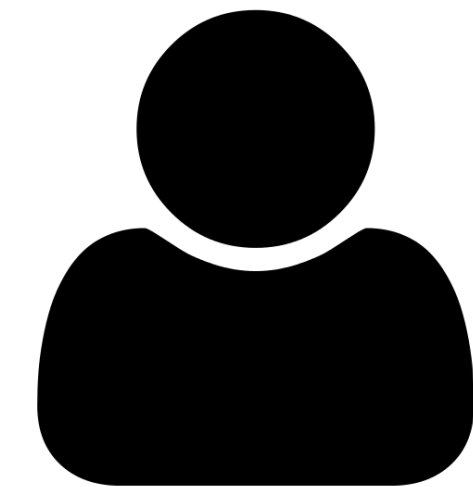
my app



Here's your auth code!



RO User



AS Auth0



RS Google



https://example.com/callback?

code=123&

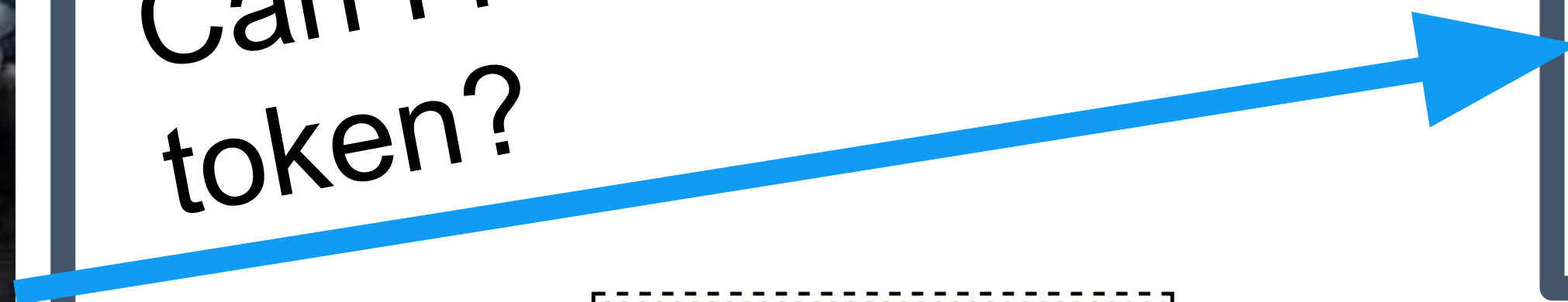
state=some_random_string

CLIENT

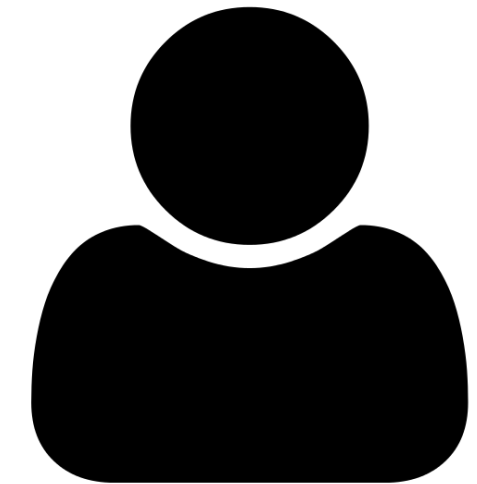
my app



Can I have an access token?



RO User



AS Auth0



RS Google




```
POST https://auth-server.com/token
  grant_type=authorization_code&
code=123&
redirect_uri=https://example.com/callback&
client_id=client_id123&
```


CLIENT

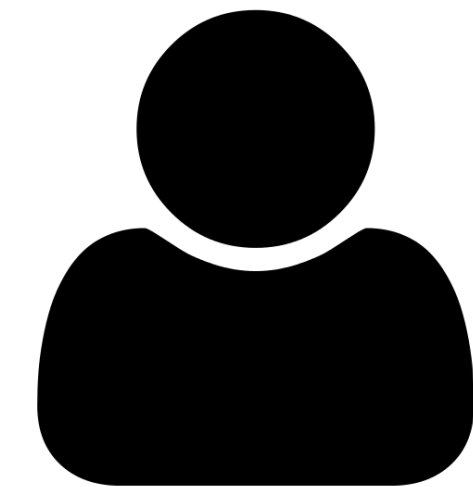
my app



Here's your access
token, refresh token and
ID token!



RO User



AS Auth0



RS Google





HTTP/1.1 200 OK

Content-Type: application/json

Cache-Control: no-store

Pragma: no-cache

{

 "access_token": "LSDLFJSDFKLK123",

 "token_type": "bearer",

 "expires_in": 3600,

 "refresh_token": "DSLFSCLKDF12321",

 "id_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.

 eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6Ikpva

 G4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.

 SflKxwRJSMeKKF2QT4fwpMeJf36P0k6yJV_adQssw5c"

}

CLIENT

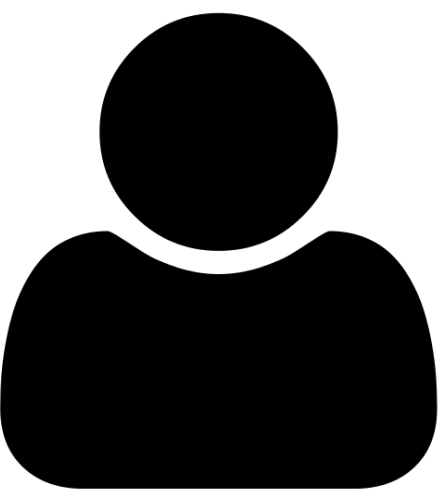
my app



Can I have user data?



RO User



AS Auth0



RS Google





```
POST /user/123 HTTP/1.1
```

```
Authorization: Bearer asdasdlq323342"
```

```
Host: google.com
```


CLIENT

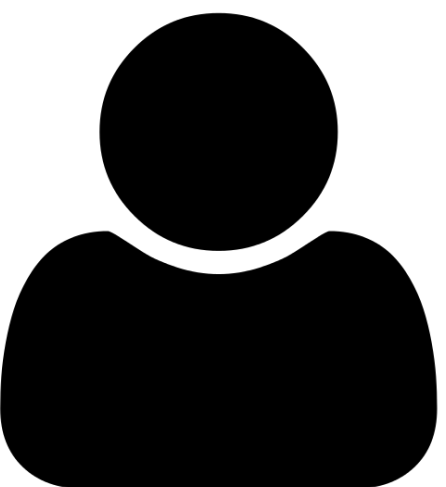
my app



Here's the user data!



RO User



AS Auth0



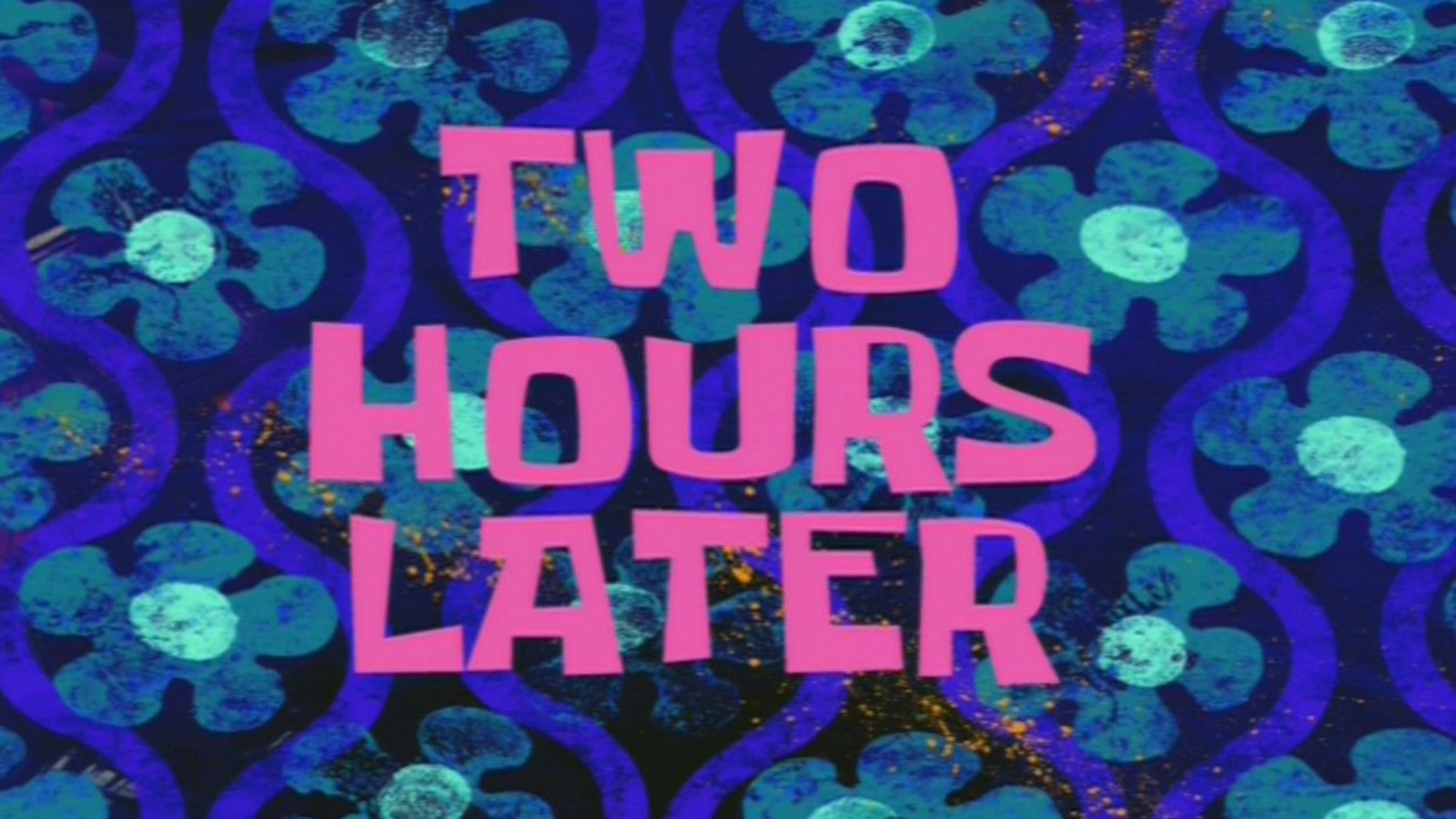
RS Google





```
{
  "displayName": "Stephanie Chamblee",
  "id": "0200107020277127",
  "user_id": "00007117",
  "name": {
    "familyName": "Chamblee",
    "givenName": "Stephanie"
  },
  "emails": [
    {
      "value": "stephaniejoychamblee@gmail.com"
    }
  ],
  "picture": "https://lh3.googleusercontent.com/a-/AAuE7mD0BfvvtIJgdy7FqJecZ0bGjUhCqUWcX-_Jw833HA",
  "locale": "en",
  "nickname": "stephaniejoychamblee"
}
```



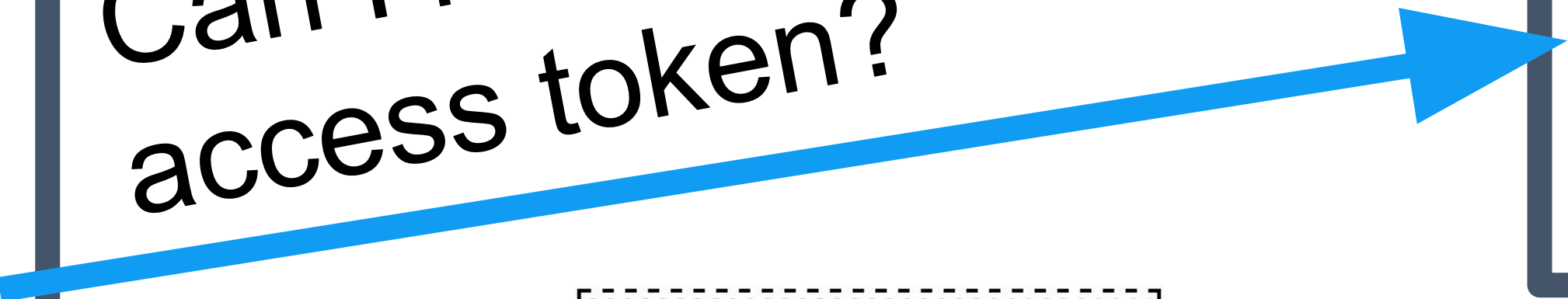
**TWO
HOURS
LATER**

CLIENT

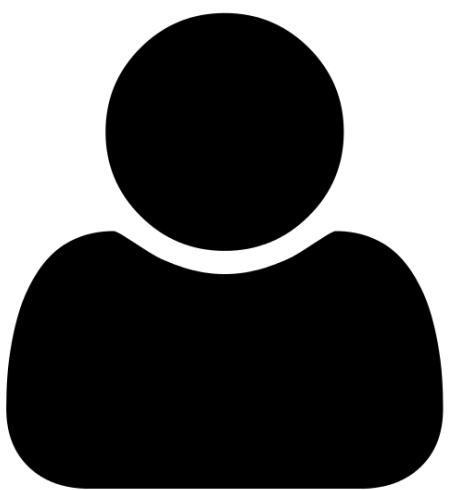
my app



Can I have another access token?



RO User



AS Auth0

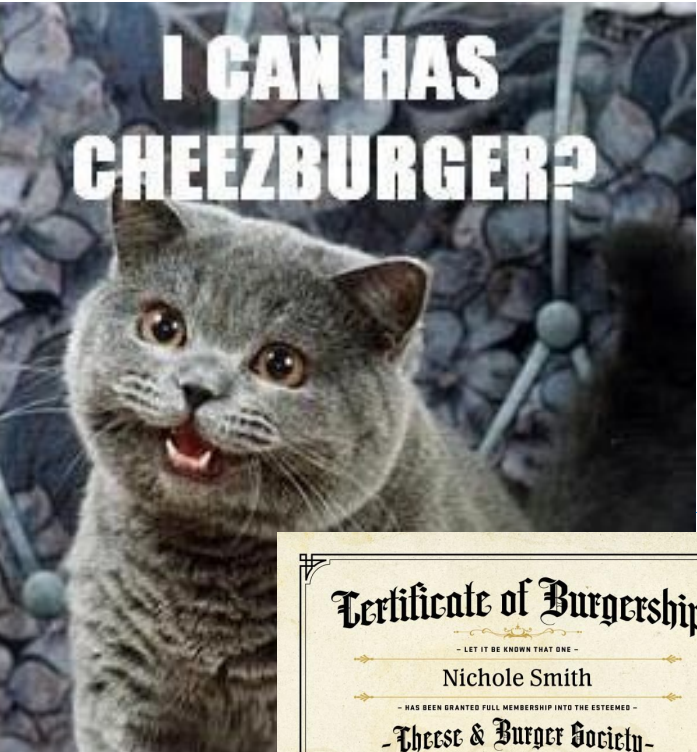


RS Google



CLIENT

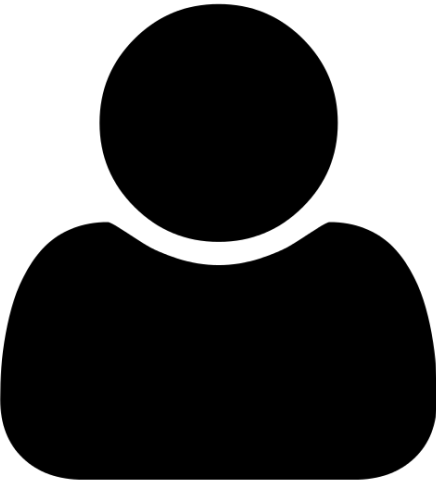
my app



Here's another
access token!



RO User



AS Auth0



RS Google



CLIENT

my app



Can I have user data



RS Google



AS Auth0



RS Google



CLIENT

my app



YEAH!



RS Google



AS Auth0



RS Google



DEMO

Summary

1

CONTEXT

Open Standards

Brief History of Identity

2

FOUNDATION

Four roles in OAuth

Tokens

Authorization Flows

3

OAUTH & OIDC AUTHORIZATION CODE FLOW

OAuth 2.0 & OpenID Connect (OIDC) Walkthrough

Resources

- [RFC 6749 - OAuth 2.0](#)
- [RFC 6750 - Bearer Tokens](#)
- [RFC 7636 - Proof Key for Code Exchange](#)
- [OpenID Connect Specifications](#)
- [The OpenID Connect Handbook - Auth0](#)
- [Learn Identity Video Series - Auth0](#)



Thanks!

Stephanie Chamblee
@stephchamblee