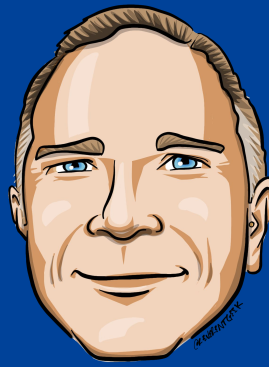
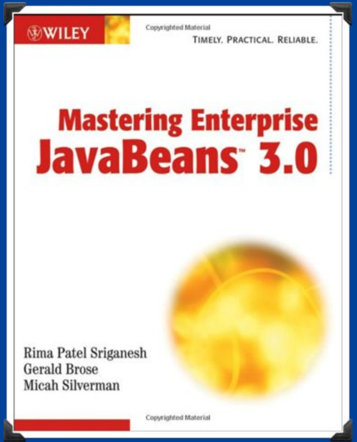
Hacking OAuth

Pitfalls and Remedies



MICAH SILVERMAN
Lead Developer Advocate



WILEY

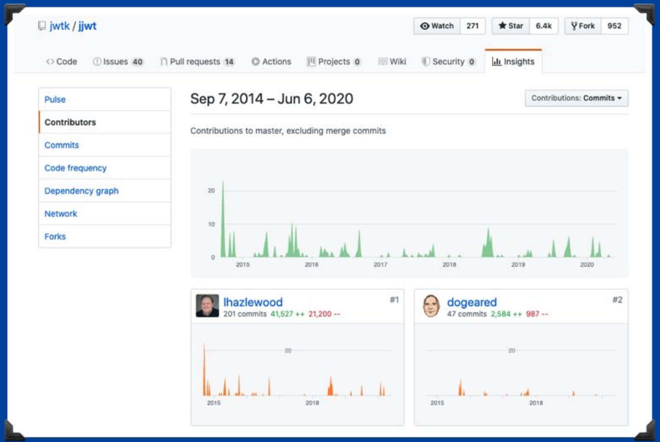
Copyrighted Material

TIMELY. PRACTICAL. RELIABLE.

Mastering Enterprise JavaBeans™ 3.0

Rima Patel Sriganesh
Gerald Brose
Micah Silverman

Copyrighted Material



jwtk / jjwt

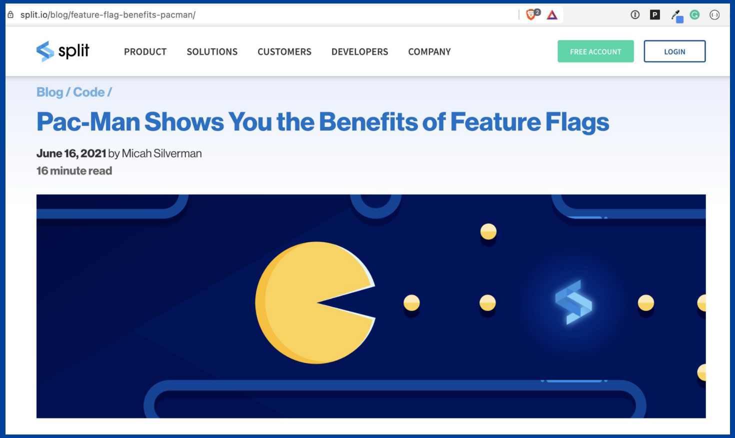
Sep 7, 2014 – Jun 6, 2020

Contributions to master, excluding merge commits

lhazlewood #1
201 commits 41,527 ++ 21,200 --

dogeared #2
47 commits 2,584 ++ 987 --

https://github.com/jwtk/jjwt



split.io/blog/feature-flag-benefits-pacman/

Blog / Code /

Pac-Man Shows You the Benefits of Feature Flags

June 16, 2021 by Micah Silverman
16 minute read

https://www.split.io/blog

split

Identity use cases (circa 2007)

01 Simple login – forms and cookies

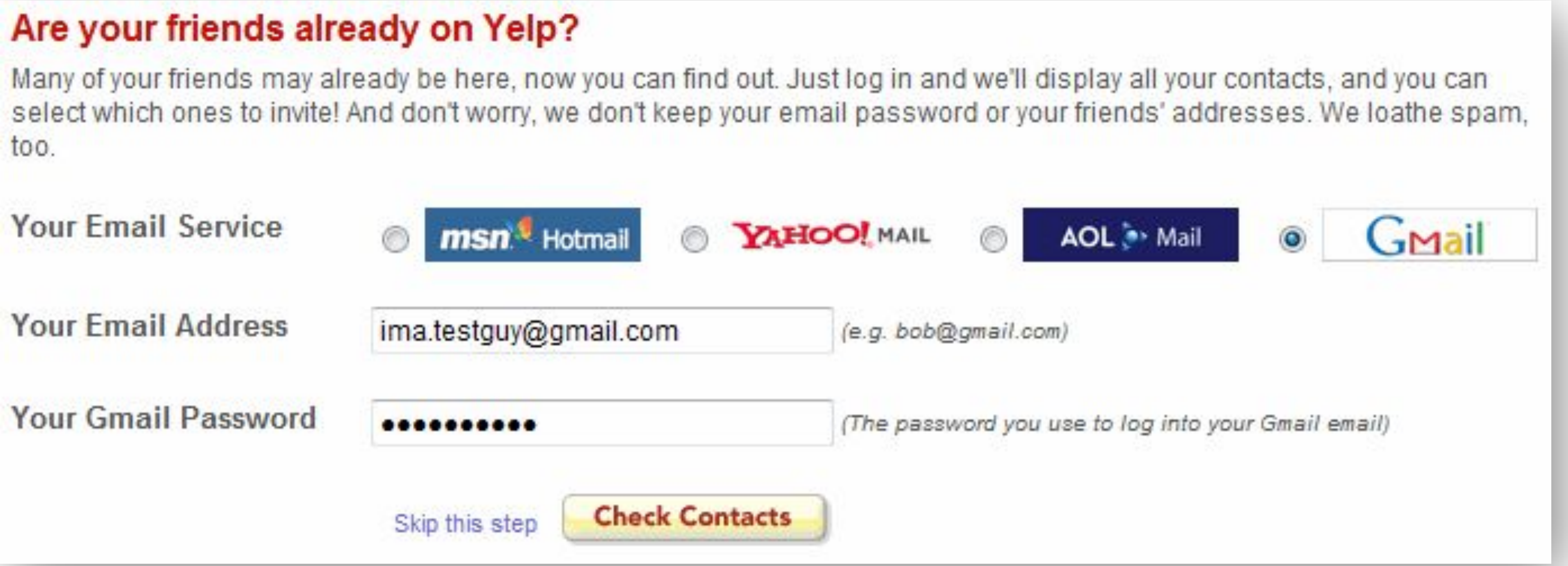
02 Single sign-on across sites – SAML

03 Mobile app login – ???

04 Delegated authorization – ???

Don’t do it This Way!

(Password Anti-Pattern)



Are your friends already on Yelp?

Many of your friends may already be here, now you can find out. Just log in and we'll display all your contacts, and you can select which ones to invite! And don't worry, we don't keep your email password or your friends' addresses. We loathe spam, too.

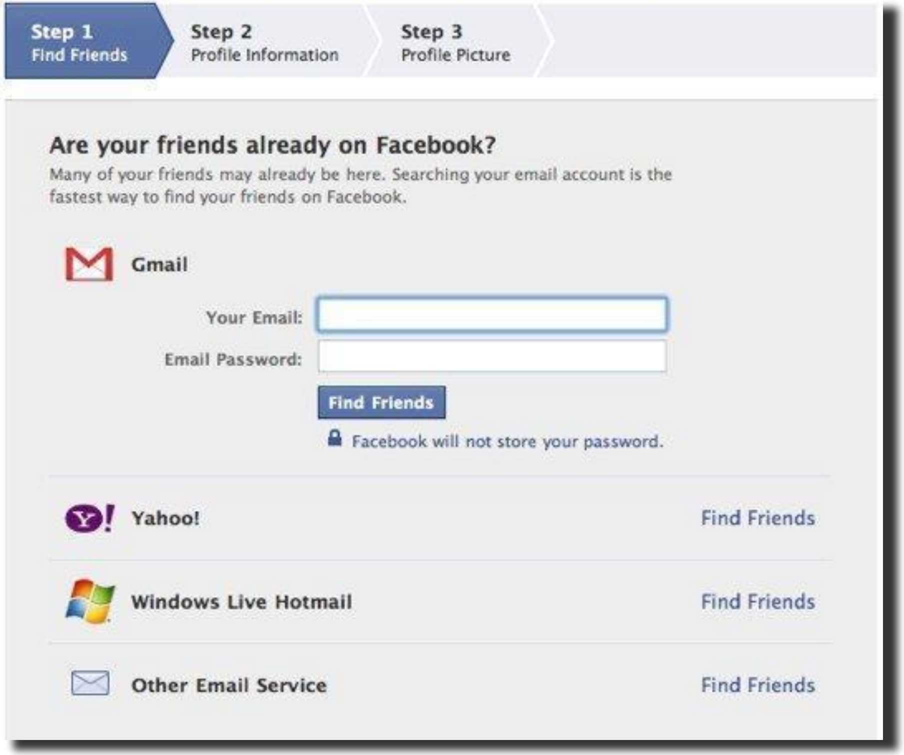
Your Email Service ○ msn Hotmail ○ YAHOO! MAIL ○ AOL Mail ● GMail

Your Email Address: ima.testguy@gmail.com (e.g. bob@gmail.com)

Your Gmail Password: ●●●●●●●●●●● (The password you use to log into your Gmail email)

Don’t do it This Way!

(Password Anti-Pattern)



Step 1 Find Friends

Step 2 Profile Information

Step 3 Profile Picture

Are your friends already on Facebook?

Many of your friends may already be here. Searching your email account is the fastest way to find your friends on Facebook.

Gmail

Your Email:

Email Password:

Find Friends

Facebook will not store your password.

Yahoo! Find Friends

Windows Live Hotmail Find Friends

Other Email Service Find Friends

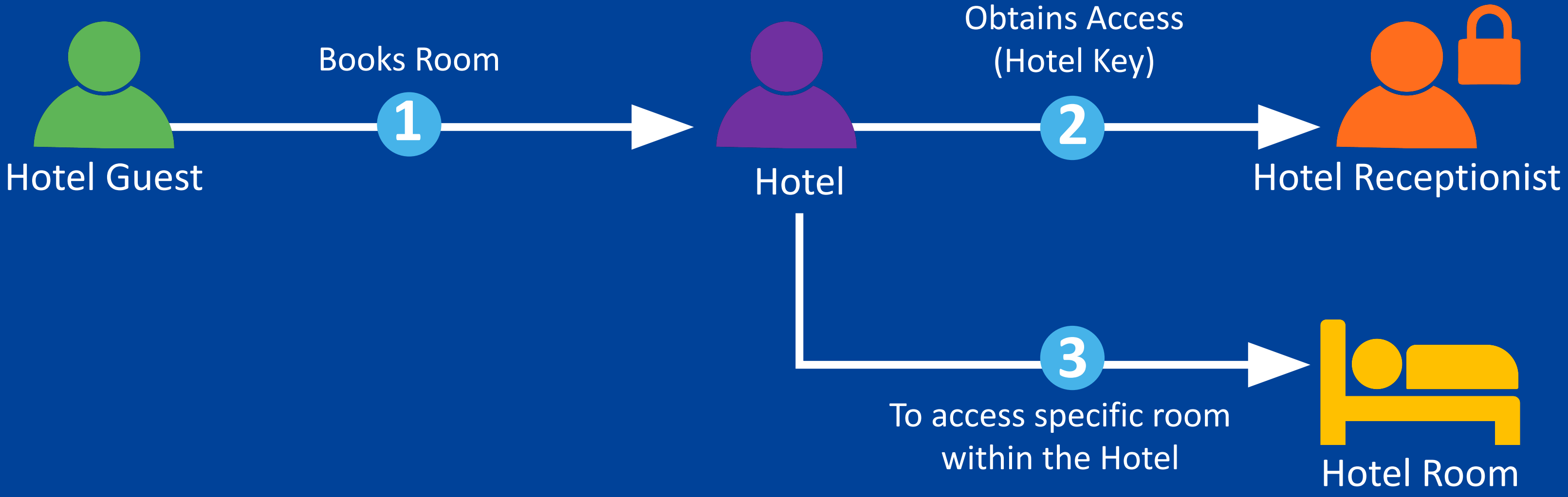
The Delegated Authorization Problem

HOW CAN I LET A WEBSITE ACCESS MY DATA, WITHOUT GIVING IT MY PASSWORD?

A lot of confusion around OAuth:

- ⇨ Terminology and jargon
- ⇨ Incorrect advice
- ⇨ Hard to find a life-like example

The Hotel Analogy



Hotel Guest

1 Books Room

Hotel

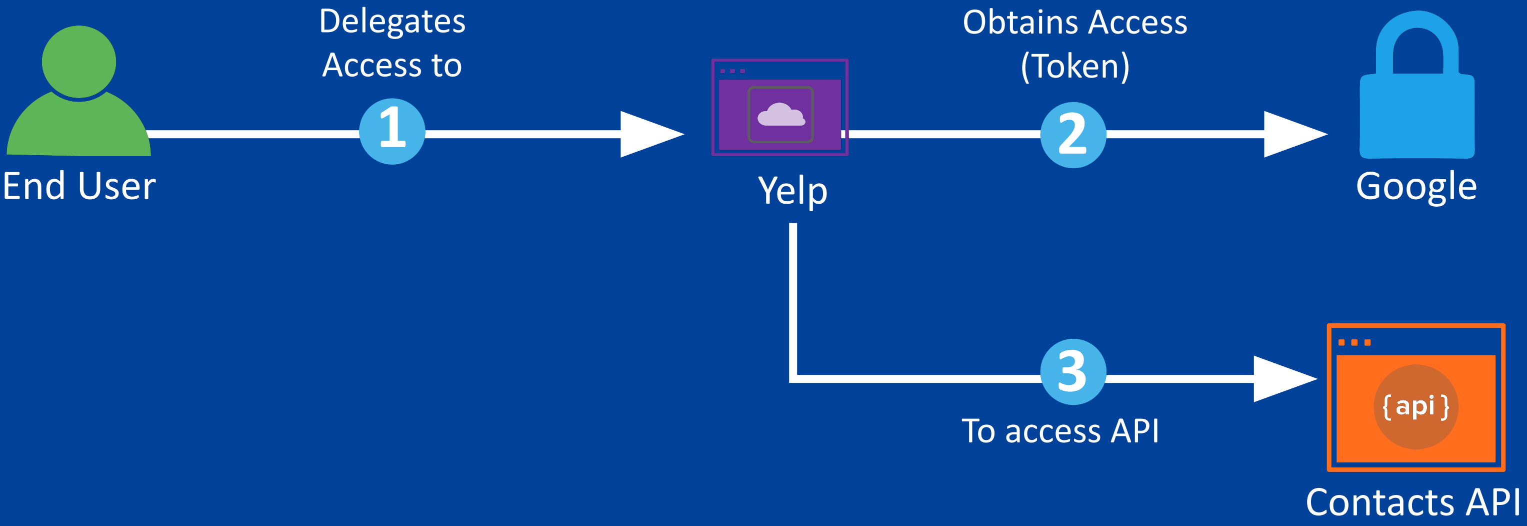
2 Obtains Access (Hotel Key)

Hotel Receptionist

3 To access specific room within the Hotel

Hotel Room

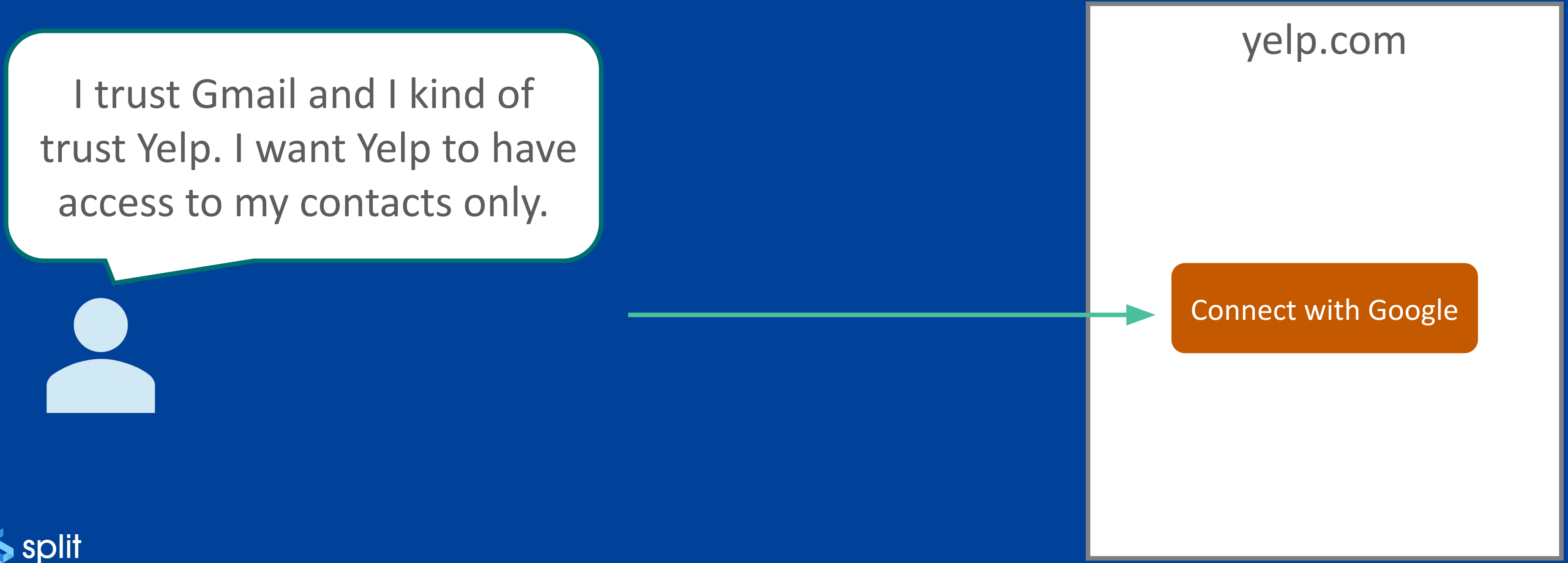
OAuth 2.0: App Scenario



OAuth 2.0 Terminology

Hotel	App	OAuth	Description
Hotel Guest	End User	Resource Owner	Wants a Client App to do something on their behalf
Hotel	Yelp	Client Application	Needs authorization to interact with an API on behalf of a user
Hotel Receptionist	Google	Authorization Server	Grants access (in the form of tokens) to an app
Hotel Room	Contacts API	Resource Server	Has an API that an app can use if presented with a token

Delegated Authorization with OAuth 2.0

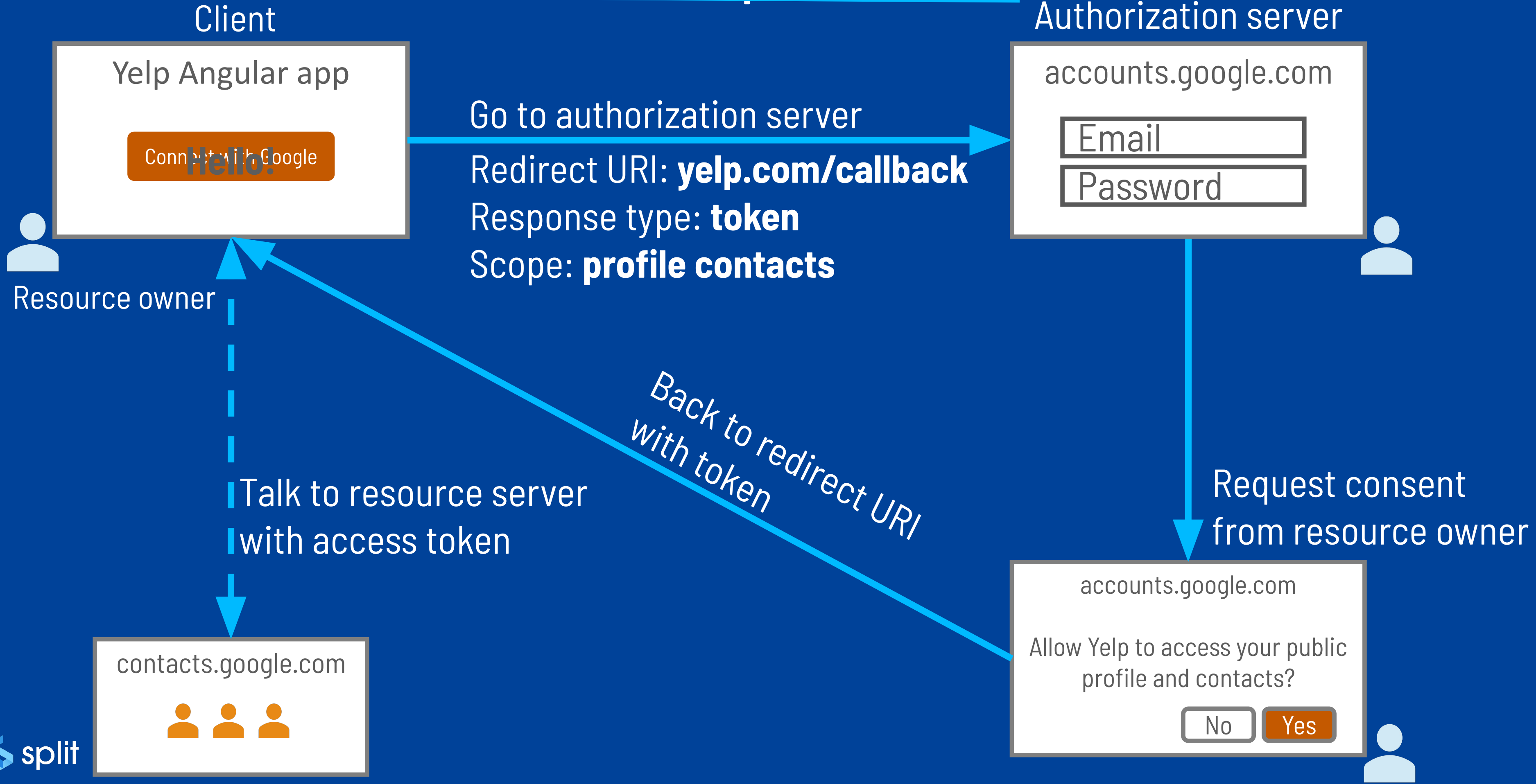


I trust Gmail and I kind of trust Yelp. I want Yelp to have access to my contacts only.

yelp.com

Connect with Google

OAuth 2.0 implicit flow



Client

Authorization server

Resource owner

Go to authorization server

Redirect URI: **yelp.com/callback**

Response type: **token**

Scope: **profile contacts**

Request consent from resource owner

Back to redirect URI with token

Talk to resource server with access token

{ DEMO Implicit Flow Detector }

split

Confidential Clients

Application running on a server

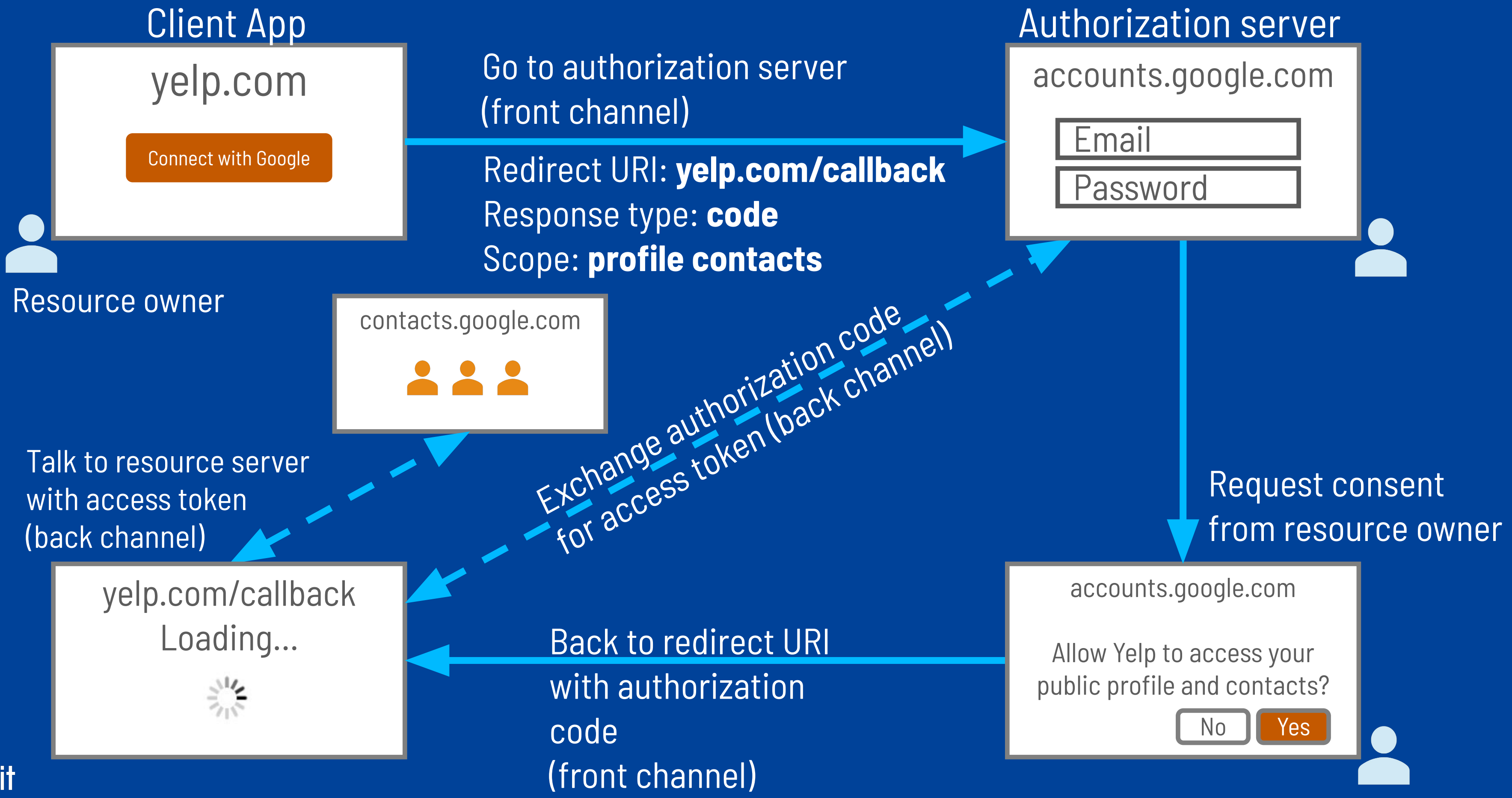
Has the ability to keep strings secret since code is running in a trusted environment

Public Clients

The application can’t keep strings secret

Javascript/Single-Page apps: “view source”
Native apps: decompile and extract strings

OAuth 2.0 authorization code flow



Client App

yelp.com

Connect with Google

Resource owner

Go to authorization server
(front channel)

Redirect URI: **yelp.com/callback**
Response type: **code**
Scope: **profile contacts**

Authorization server

accounts.google.com

Email

Password

Request consent
from resource owner

accounts.google.com

Allow Yelp to access your
public profile and contacts?

No Yes

Back to redirect URI
with authorization
code
(front channel)

yelp.com/callback
Loading...

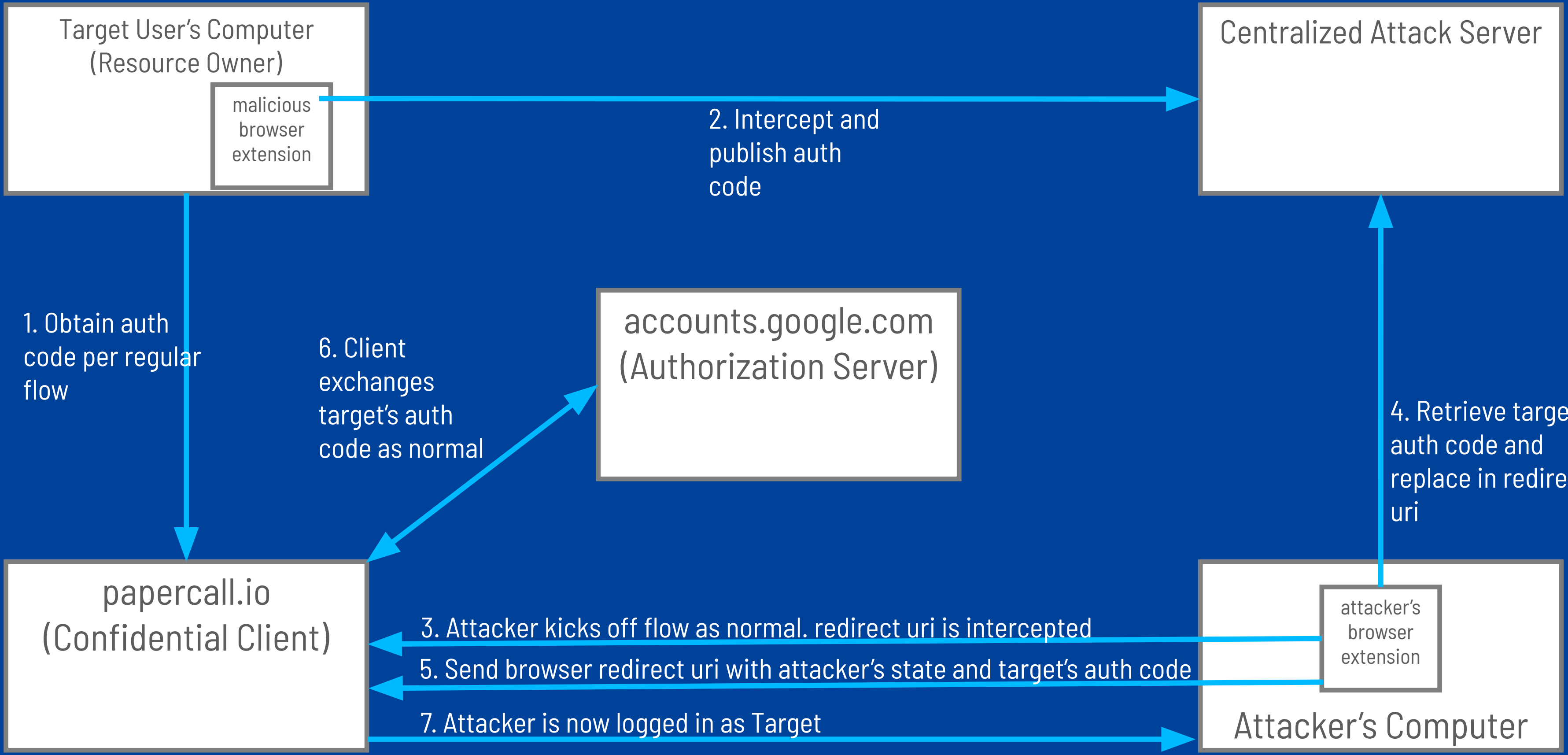
Exchange authorization code
for access token (back channel)

contacts.google.com

Talk to resource server
with access token
(back channel)

split

Authorization Code Injection Attack



Target User's Computer (Resource Owner)

malicious browser extension

Centralized Attack Server

accounts.google.com (Authorization Server)

papercall.io (Confidential Client)

attacker's browser extension

Attacker's Computer

1. Obtain auth code per regular flow

2. Intercept and publish auth code

3. Attacker kicks off flow as normal. redirect uri is intercepted

4. Retrieve target's auth code and replace in redirect uri

5. Send browser redirect uri with attacker's state and target's auth code

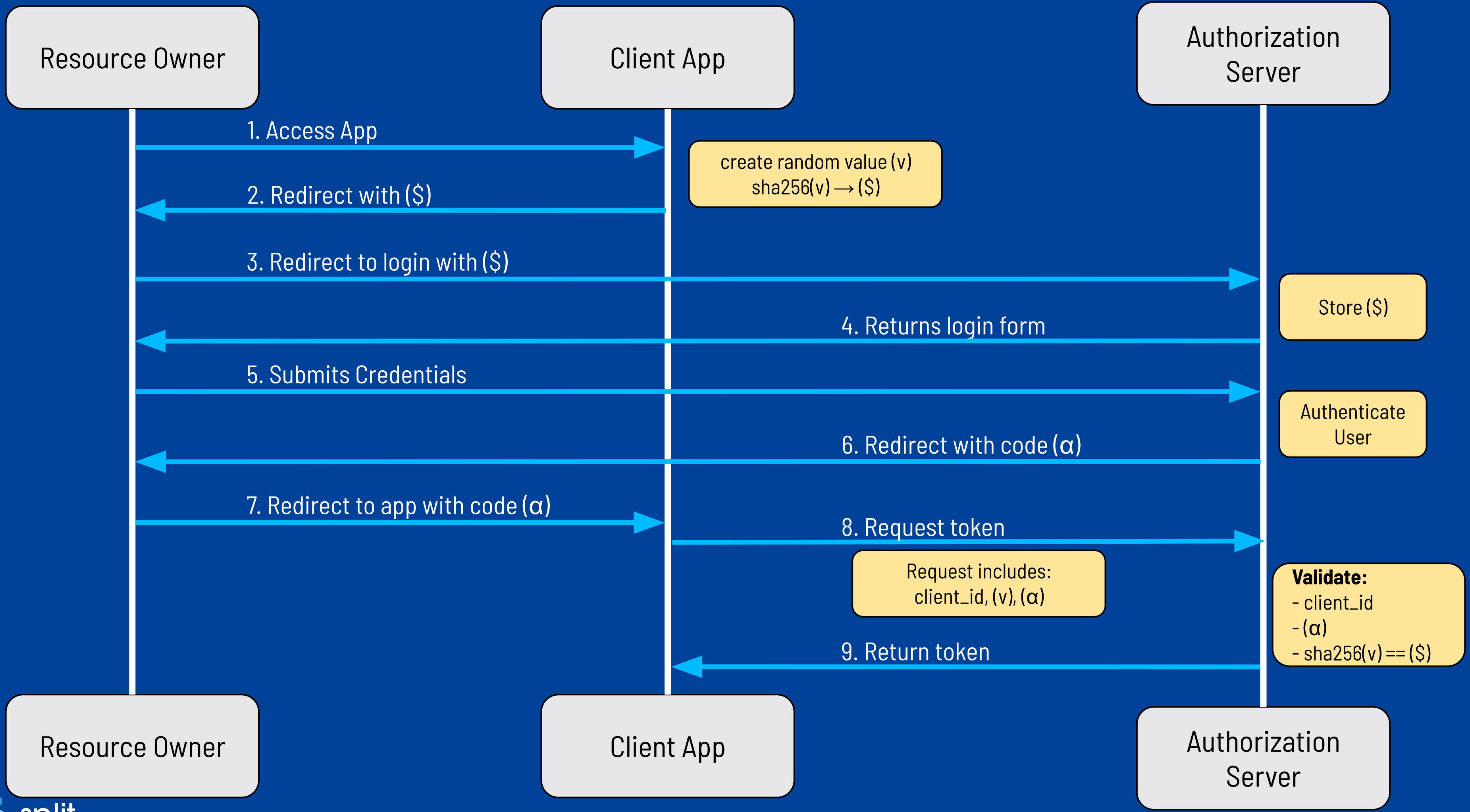
6. Client exchanges target's auth code as normal

7. Attacker is now logged in as Target

split

DEMO
Auth Code Injection

split



Resource Owner

Client App

Authorization Server

1. Access App

create random value (v)
sha256(v) → ($)

2. Redirect with ($)

3. Redirect to login with ($)

Store ($)

4. Returns login form

5. Submits Credentials

Authenticate User

6. Redirect with code (α)

7. Redirect to app with code (α)

8. Request token

Request includes:
client_id, (v), (α)

Validate:
- client_id
- (α)
- sha256(v) == ($)

9. Return token

Resource Owner

Client App

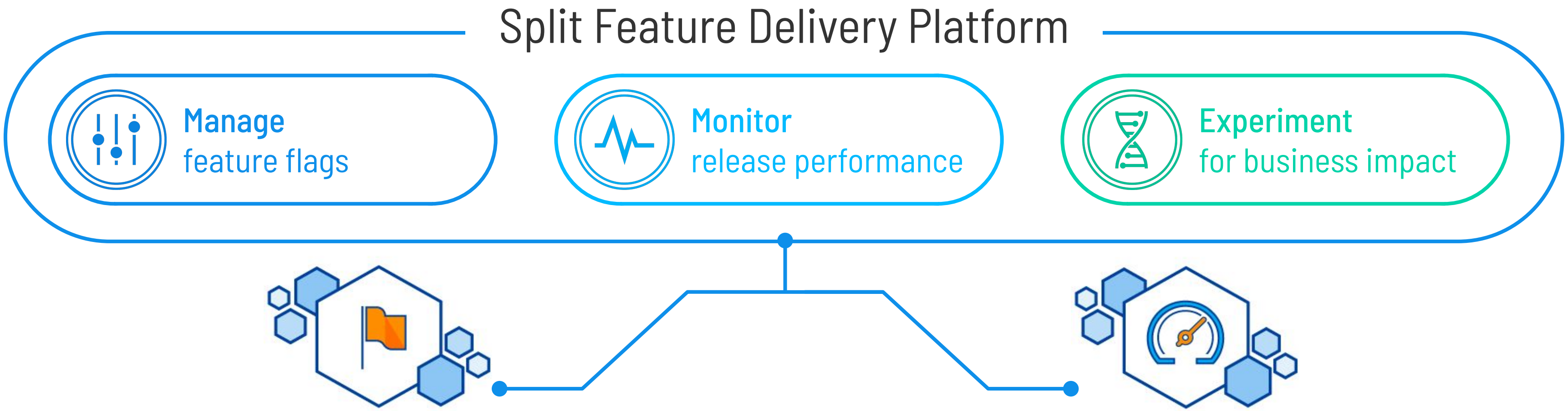
Authorization Server

DEMO

PKCE CLI

Impact-driven development requires data

Only Split's unified platform closes The Impact Gap



Split Feature Delivery Platform

Manage
feature flags

Monitor
release performance

Experiment
for business impact

Feature flags
control user access and
record 'who sees what'

Performance & behavioral data
from engineering, product,
and business sources

Hacking OAuth: Pitfalls and Remedies

Micah Silverman
@afitnerd

More on OAuth 2.0: **oauth.com**

@splitsoftware
Free developer account: **split.io/signup**