Securing Our API

Kevin Dockx
ARCHITECT

@KevinDockx https://www.kevindockx.com

Coming Up

The Hybrid Flow

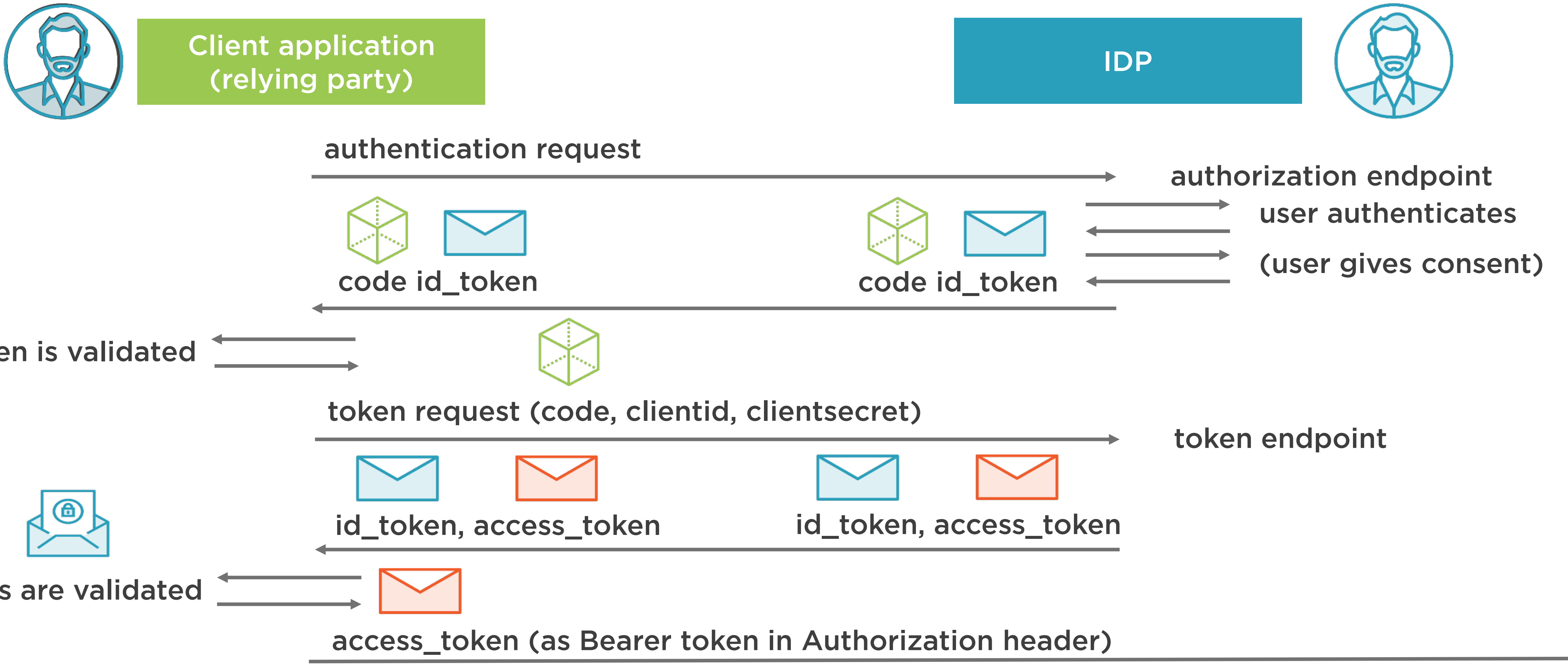
Passing an Access Token and Validating It

Specifics to Keep in Mind When Reading, Creating, Updating and Deleting Resources

Including Identity Claims in an Access Token

Role-based Authorization

The Hybrid Flow



Client application (relying party)

IDP

authentication request

authorization endpoint

user authenticates

(user gives consent)

code id_token

code id_token

token is validated

token request (code, clientid, clientsecret)

token endpoint

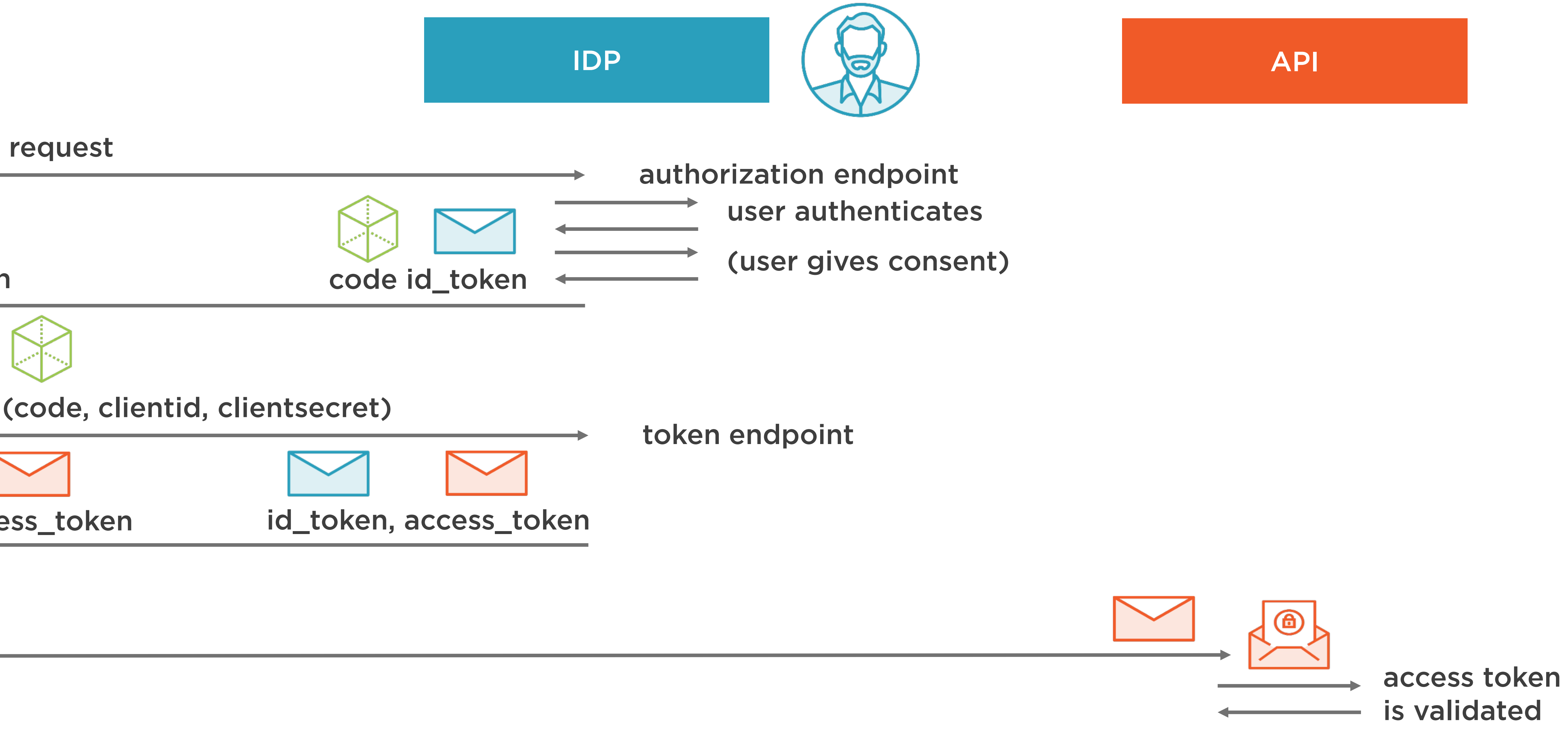
id_token, access_token

id_token, access_token

tokens are validated

access_token (as Bearer token in Authorization header)

The Hybrid Flow



IDP

API

request

authorization endpoint

user authenticates

(user gives consent)

n

code id_token

(code, clientid, clientsecret)

token endpoint

ess_token

id_token, access_token

access token
is validated

Demo

Securing Access to Our API

Demo

Passing an Access Token to Our API

Demo

Showing an Access Denied Page

Demo

Protecting the API When Getting a Resource Collection

Demo

Protecting the API When Getting a Single Resource

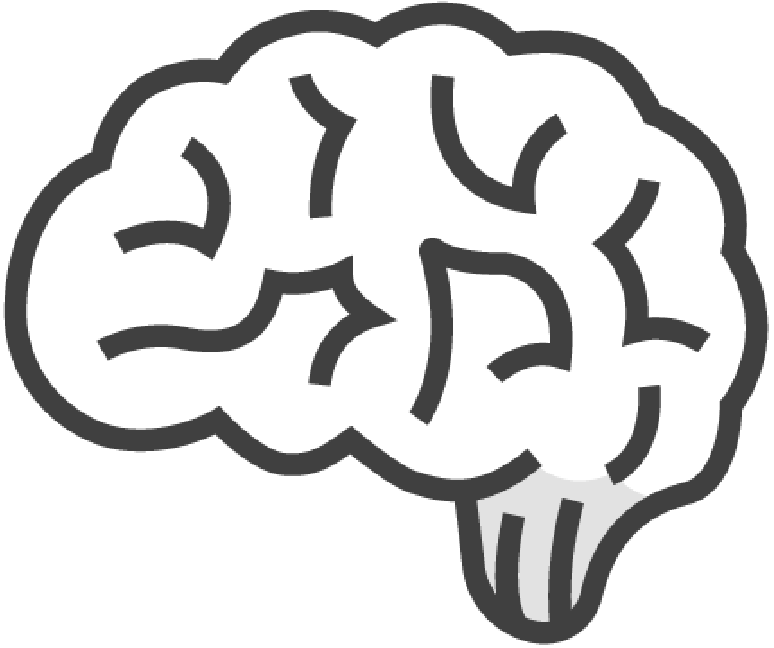
Demo

Protecting the API When Updating a Resource

Demo

Protecting the API When Deleting a Resource

Including Identity Claims in an Access Token



Sometimes an API needs access to identity claims

When defining a resource scope (API resource), include the required claims in the claims list

Demo

Including Identity Claims in an Access Token

Demo

Protecting the API When Creating a Resource (With Roles)

Summary

Access tokens are passed to the API as Bearer tokens

AccessTokenValidationmiddleware can be used to validate an access token at level of the API

Summary

Take care of

- Checking that a resource belongs to the current user
- Getting the user identifier from the token
- Avoiding invalid input by using separate classes

Role-based authorization