Understanding Authorization with OAuth2 and OpenID Connect

Kevin Dockx
ARCHITECT

@KevinDockx https://www.kevindockx.com

Coming Up

How OAuth2 works

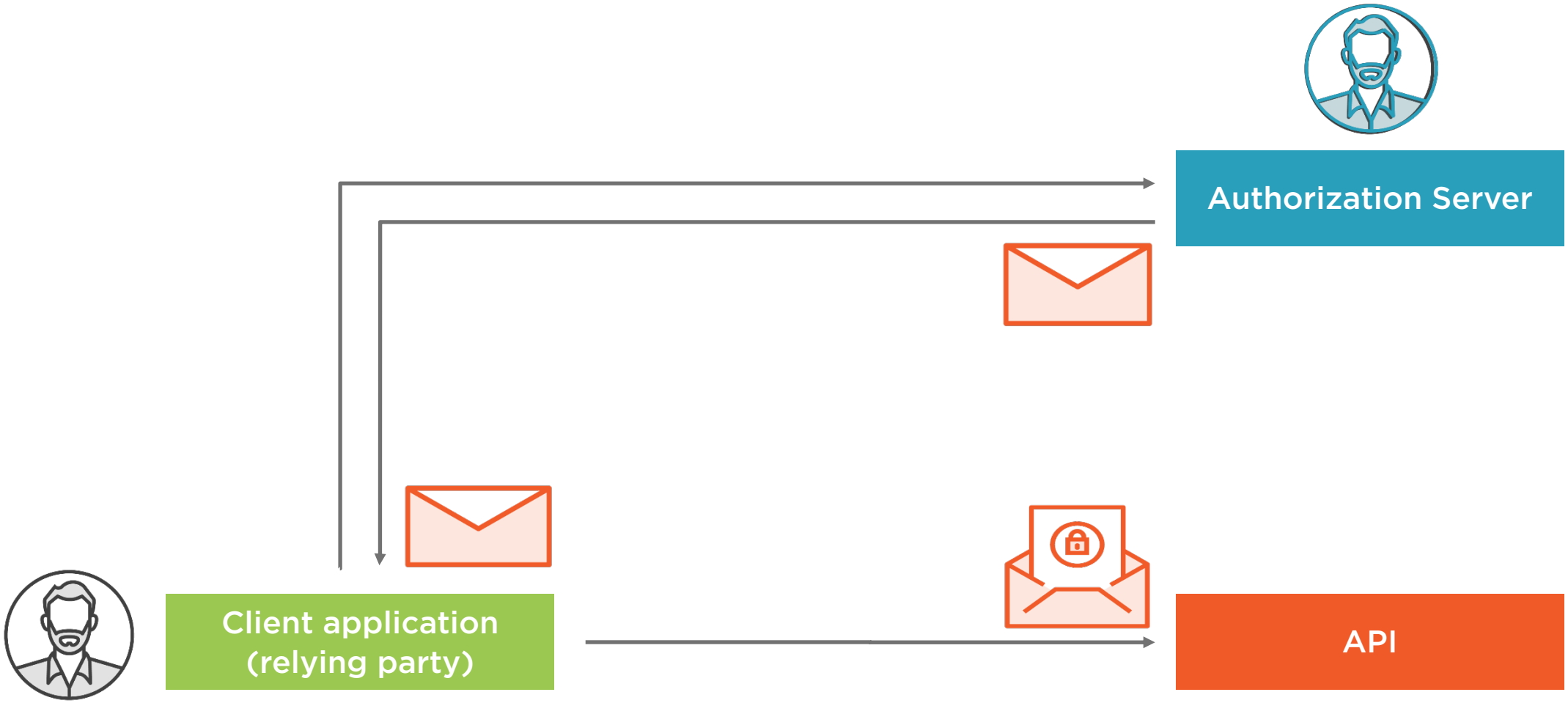
Why OIDC is preferred over OAuth2

Using OpenID Connect for authentication and authorization

OIDC/OAuth2 flows

Inspecting an access token

How OAuth2 Works



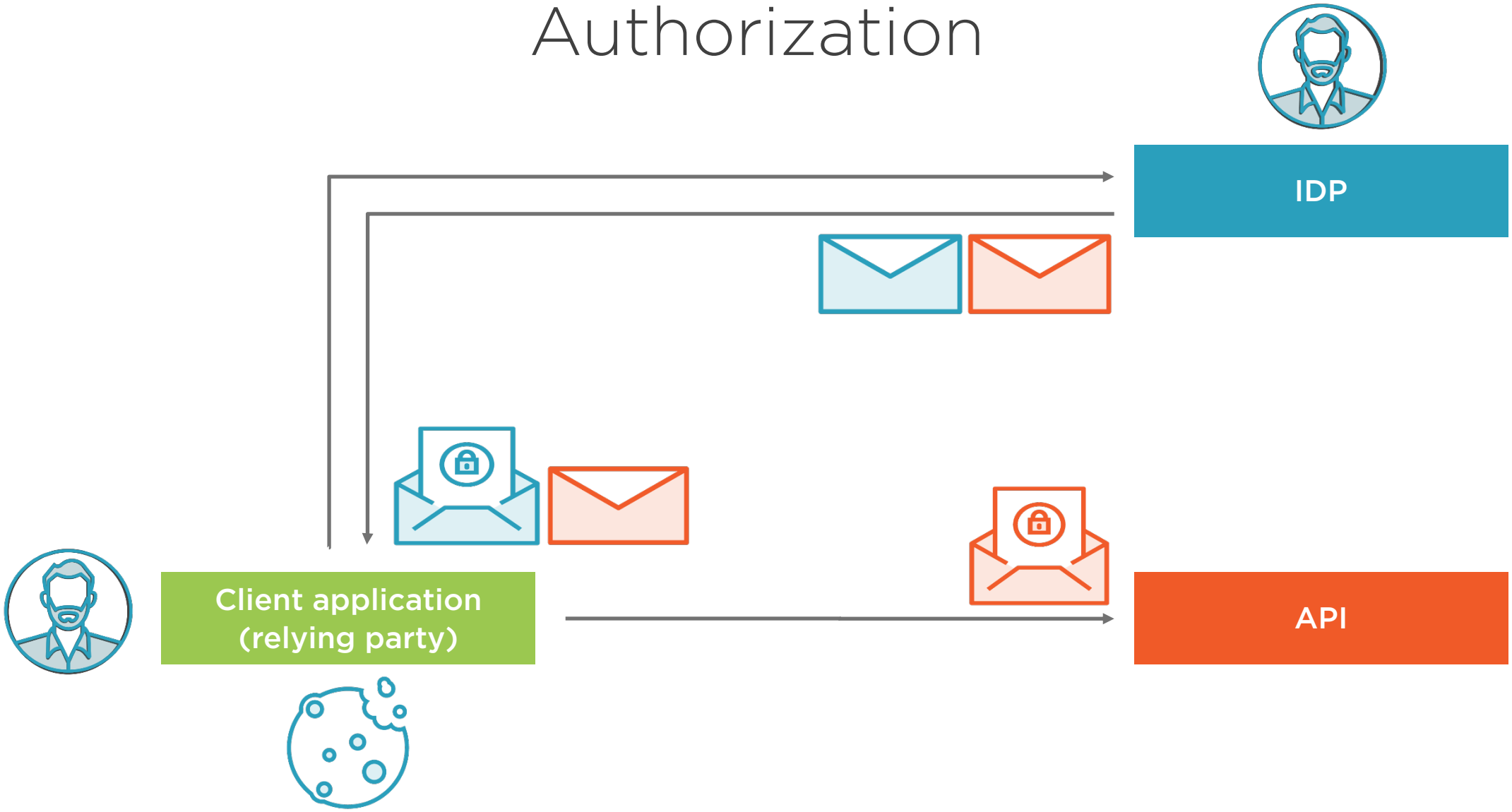
Why OpenID Connect Is Preferred Over OAuth2

The identity token can be linked to the access token (at_hash)

When using the hybrid flow, the identity token can be verified first

The nonce protects against replay attacks

Using OpenID Connect for Authentication and Authorization



OAuth2 and OpenID Connect Flows

Authorization Code

Tokens from token endpoint

Confidential clients

Long-lived access

Implicit

Tokens from authorization endpoint

Public clients

No long-lived access

Hybrid (OIDC only)

Tokens from authorization endpoint & token endpoint

Confidential clients

Long-lived access

OAuth2 and OpenID Connect Flows

Resource Owner Password Credentials (OAuth2 only)

In-app login screen

Only for trusted applications

Should be avoided

Client Credentials (OAuth2 only)

No user involvement

Confidential clients

For machine to machine communication

```
{
  "sub": "b7539694-97e7-4dfe-84da-b4256e1ff5c7",
  "iss": "https://localhost:44303",
  "aud": [
    "imagegalleryapi“,
    "https://localhost:44303/resources"],
  …
}
```

Inspecting an Access Token

Access tokens are often JWTs, but don’t have to be (e.g.: reference tokens)

```
{
  "sub": "b7539694-97e7-4dfe-84da-b4256e1ff5c7",
  "iss": "https://localhost:44303",
  "aud": [
    "imagegalleryapi“,
    "https://localhost:44303/resources"],
  …
}
```

Inspecting an Access Token

The intended audience

- Our image gallery API
- Resources at level of the IDP (e.g. when calling the UserInfo endpoint)

```
{ …
  "client_id": "imagegalleryclient",
  "nbf": 1491235799,
  "exp": 1491235869,
  "auth_time": 1491235794,
  …
}
```

Inspecting an Access Token

The client identifier signifies the client application that requested the access token

```
{ …
  "scope": [
    "openid",
    "imagegalleryapi",
    "profile"],
  "amr": ["pwd"]
}
```

Inspecting an Access Token

The scopes in this token give access to API resources and Identity resources

Summary

Use OIDC for authentication and authorization, as it's the superior protocol

The advised flow is the authorization code flow with PKCE protection

- The hybrid flow is a valid alternative

Summary

OAuth2-only flows

- ROPC should be avoided
- Client credentials is for machine to machine communication