Enabling Long-lived Access and Token Stores

Kevin Dockx
ARCHITECT

@KevinDockx https://www.kevindockx.com

Coming Up

Learning about token expiration

Refreshing and storing tokens with IdentityModel.AspNetCore

Implementing token store support for custom flows

Learning About Token Expiration

When a token expires, the flow can be triggered again to get a new one

- Visible redirection
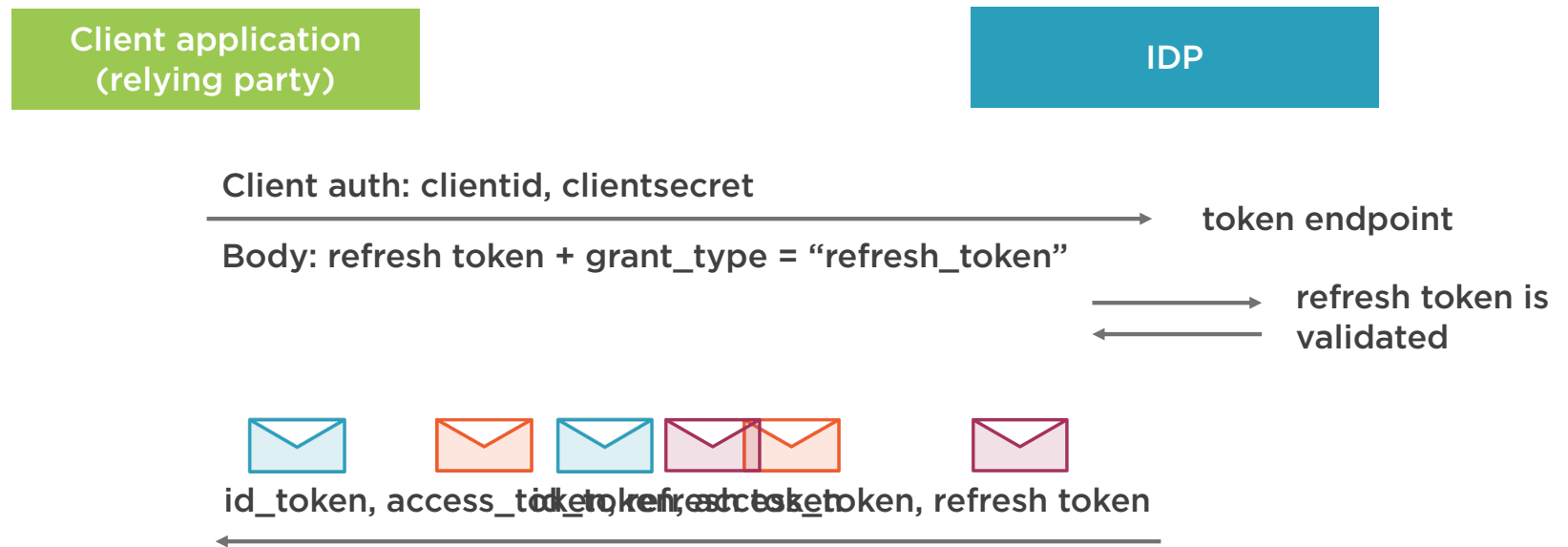- Potentially the user has to authenticate again

Learning About Token Expiration

Clients that can safely store secrets can use refresh tokens to get new tokens via the back channel

- A refresh token is a credential to get new tokens
- Avoids user interaction and browser redirection

Refresh Token Flow

Client application (relying party)

IDP

Client auth: clientid, clientsecret

token endpoint

Body: refresh token + grant_type = "refresh_token"

refresh token is validated

id_token, access_token, refresh token, access token, refresh token

Learning About Token Expiration

Scope: "offline_access"

– "Access to your applications and resources, even when you are offline"

– Offline in this context means the user is not logged in to the identity service

Learning About Token Expiration

IdentityModel.AspNetCore

- https://github.com/IdentityModel/ IdentityModel.AspNetCore

- Implements token refresh

- Implements token stores

Demo

Refreshing tokens with IdentityModel.AspNetCore

Refreshing and Storing Tokens When Flows Are Not Initiated by the User

Non-user initiated flows

- No refresh token required
- Execute the flow to get a new token

Tokens should be reused when not expired

- Avoids unnecessary requests to the identity service

Refreshing and Storing Tokens When Flows Are Not Initiated by the User

Use a token store to store tokens and fetch them from for reuse

- Cookie (user-initiated)
- In-memory, cache, database, ...

Demo

Adding token store support for custom flows

Summary

Expired tokens can be renewed by executing the flow again

- Potential visible redirection
- Potential re-authentication required

Refresh token

- Credential to get new token(s)
- Backchannel communication without visible redirection

Summary

IdentityModel.AspNetCore

- Token refresh
- Token stores

Avoid sending too many request to the identity service

- Re-execute flows only when tokens are about to expire