Adding Authentication and Authorization

Daniel Villamizar

Senior Cloud Solutions Architect - MVP

@danielvillamizara – https://www.linkedin.com/in/danielvillamizara/

Module overview

Securing the Blazor application

Adding an identity provider

Authenticating in the application

Securing the API calls

Configuring authorization

Securing the Blazor Application

Securing the Blazor Application

Users will need to authenticate

Authentication: who you are
Authorization: what you can do

API will need to verify that call is allowed

Security Alert!

Since it’s all client-side, users can bypass any checks we build in!

We’ll need to validate on the server!

Building Blocks for Authentication

Nuget package: Microsoft.AspNetCore.Components.WebAssembly.Authentication

AuthenticationStateProvider service

CascadingAuthenticationState

Authentication via OIDC

- Supported via Microsoft.AspNetCore.Components.WebAssembly.Authentication package
- Defaults to Duende Identity Server
- Support for other OpenID Providers

Building Blocks for Authorization

AuthorizeView

[Authorize]

Can be combined with roles

There’s more to learn!

We will be covering a practical approach to securing our application,

Adding an Identity Provider

Introducing an Identity Provider

Creating your own security system is not a good idea!

Identity Providers take away the need

Offer authentication and API access checks

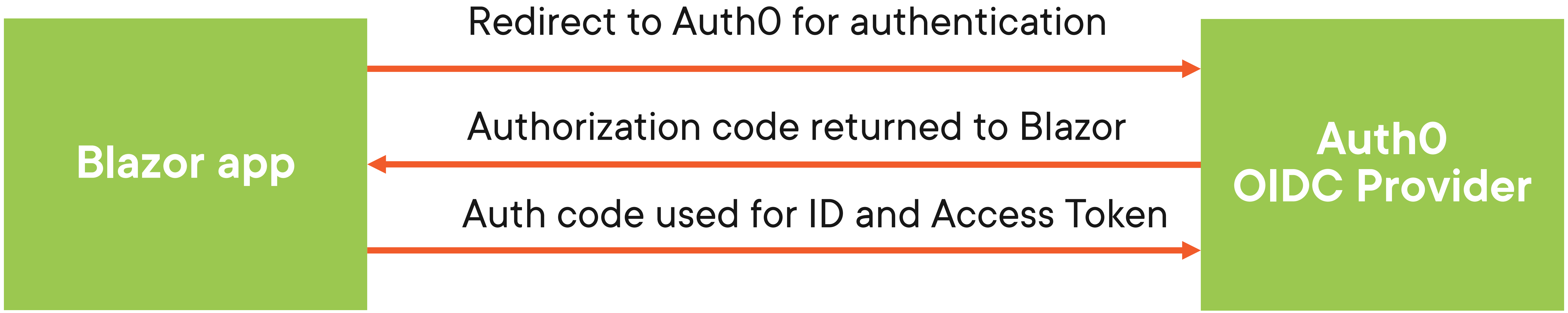
Common Identity Providers

- IdentityServer (Duende)
- Okta
- Auth0
- Azure Active Directory

Using Auth0

- Will handle login and registration
- Users are forwarded to Auth0
- Returns to app once complete
- For Blazor, we use the PKCE flow
- Free up to 7000 users

Using Auth0



Blazor app

Redirect to Auth0 for authentication

Authorization code returned to Blazor

Auth code used for ID and Access Token

Auth0
OIDC Provider

Demo

Creating the Auth0 account

Registering our application with Auth0

Authenticating in the Application

Adding Authentication Support with Auth0

1ST Add appsettings.json with Auth0 values

2ND NuGet package & script reference

3RD Configuration changes (AddOidcAuthentication())

4TH CascadingAuthenticationState and AuthorizeRouteView (App.razor)

5TH Components: Authentication.razor and AuthenticationStatus.razor

Default behaviors

- Redirect to /authentication/login
  - [Authorize] attribute
- /authentication/login-callback
- Authentication response processing

Demo

Adding authentication with Auth0

Configuring Authorization

Adding Authorization

[Authorize]

AuthorizeView component

Demo

Using the [Authorize] attribute

Adding the AuthorizeView component

Securing the API Calls

Demo

Registering the API with Auth0

Securing the API endpoints

Making authenticated API calls

Summary

Blazor favors use of OIDC

Authentication library helps greatly

Auth0 is a common provider for use with Blazor

API needs to be secured too

Up next:

Testing components