Authentication and Authorization with Blazor

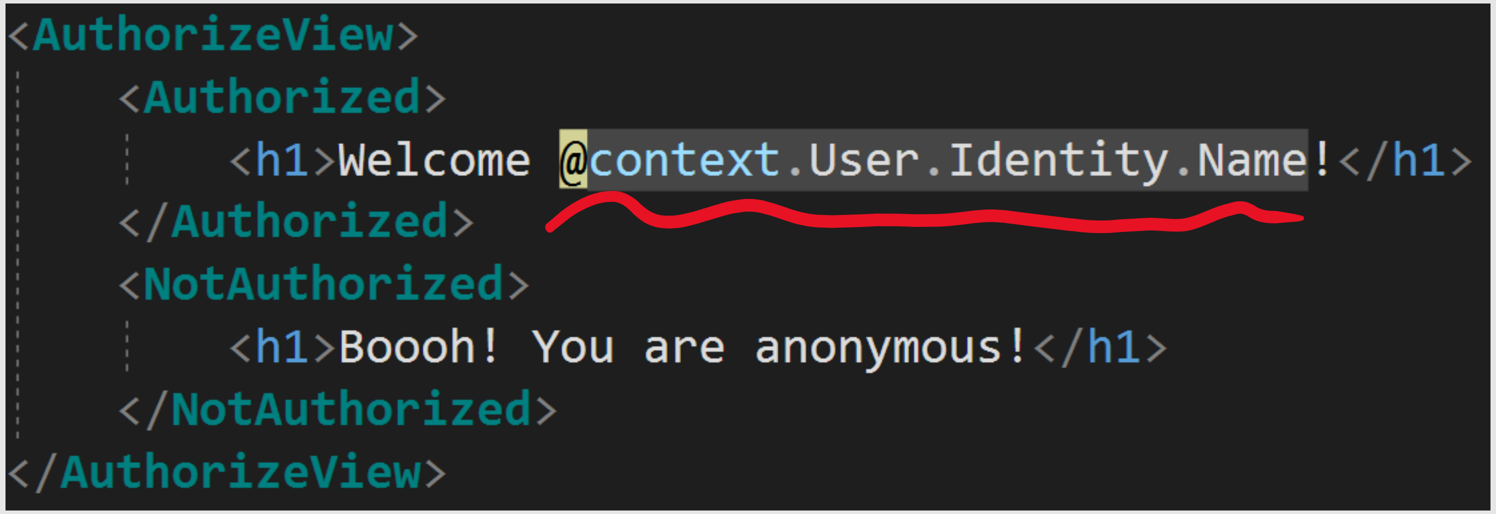
Marco De Sanctis
Developer Technologies MVP
info@marcodesanctis.it - @crad77

Agenda

- A glimpse of the object model for security
- How security works in Blazor Server
- How Blazor WebAssembly is different
- Role-based authorisation
- A better way of modelling permissions
- Some other goodies (depends on time ☺)

Which tools are available?

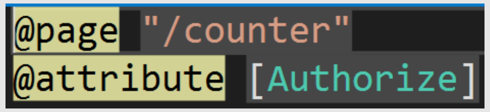
AuthorizeView



```
<AuthorizeView>
    <Authorized>
        <h1>Welcome @context.User.Identity.Name!</h1>
    </Authorized>
    <NotAuthorized>
        <h1>Boooh! You are anonymous!</h1>
    </NotAuthorized>
</AuthorizeView>
```

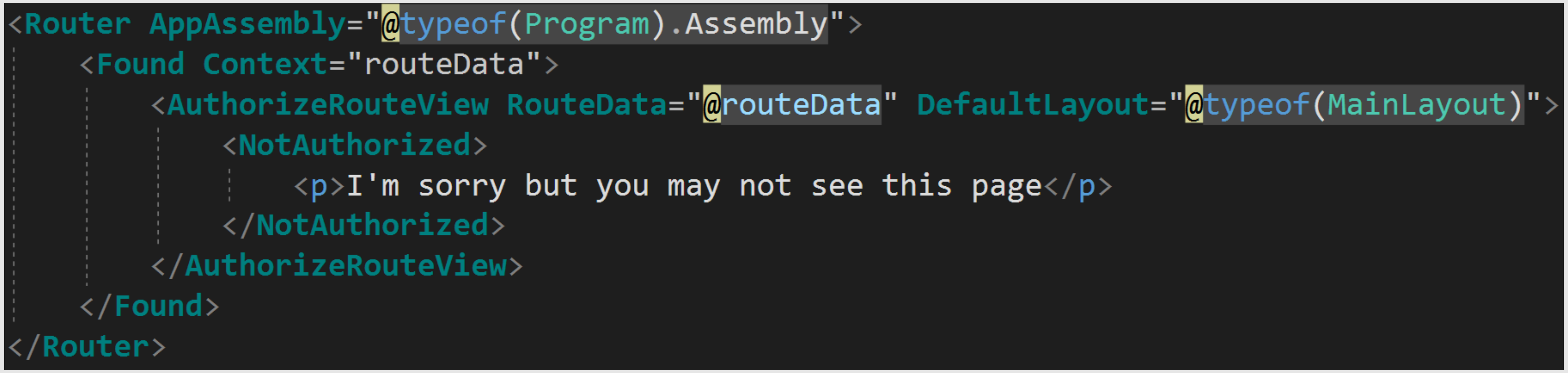
Which tools are available?

AuthorizeAttribute



```
@page "/counter"
@attribute [Authorize]
```

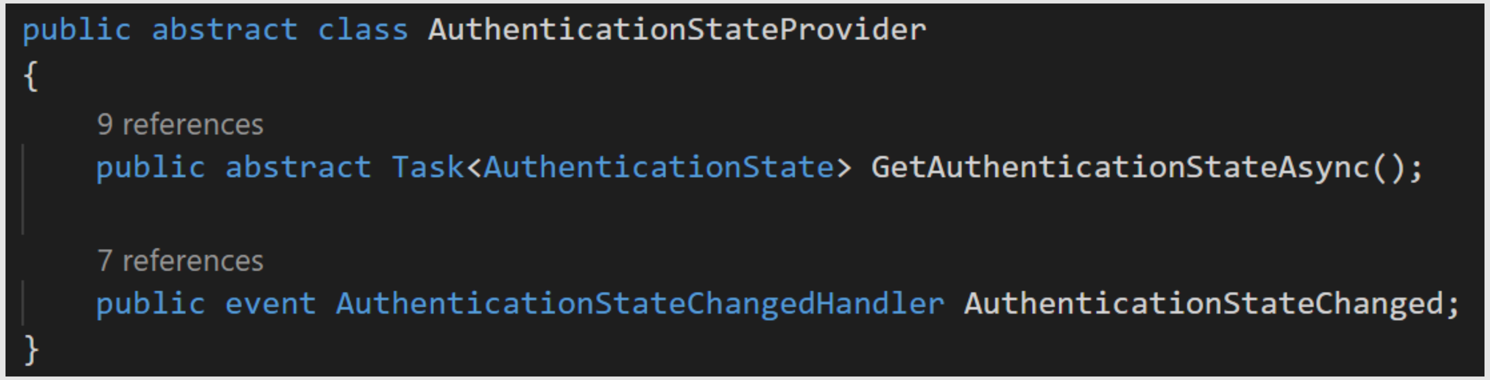
AuthorizeRouteView



```
<Router AppAssembly="@typeof(Program).Assembly">
    <Found Context="routeData">
        <AuthorizeRouteView RouteData="@routeData" DefaultLayout="@typeof(MainLayout)">
            <NotAuthorized>
                <p>I'm sorry but you may not see this page</p>
            </NotAuthorized>
        </AuthorizeRouteView>
    </Found>
</Router>
```

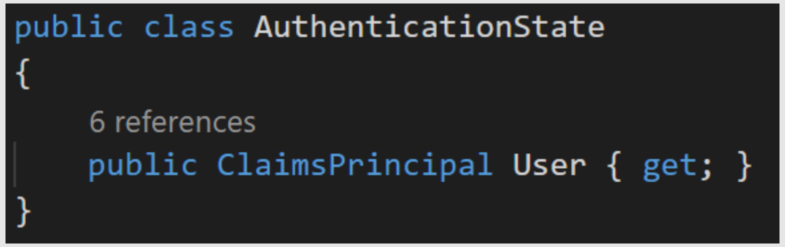
Which tools are available?

`AuthenticationStateProvider`



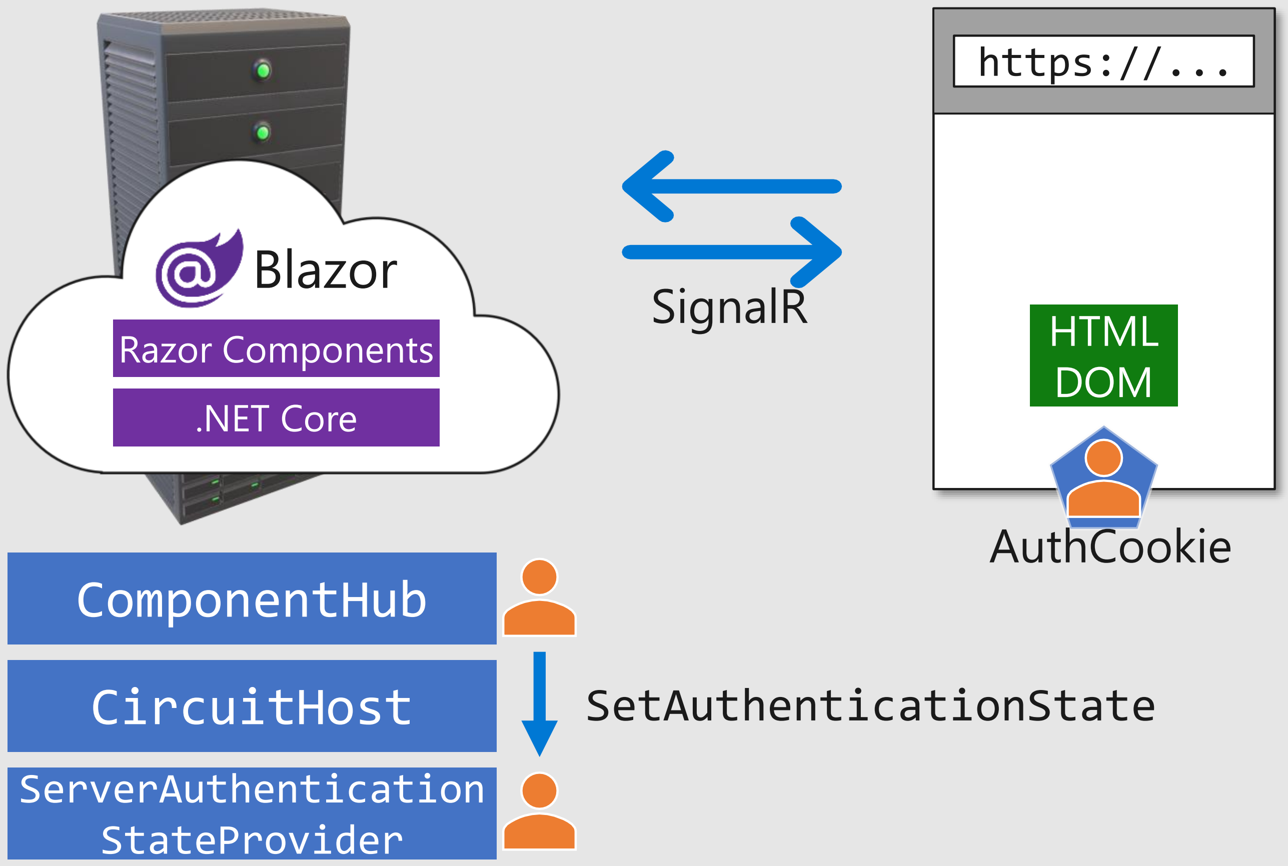
```
public abstract class AuthenticationStateProvider
{
    9 references
    public abstract Task<AuthenticationState> GetAuthenticationStateAsync();

    7 references
    public event AuthenticationStateChangedHandler AuthenticationStateChanged;
}
```

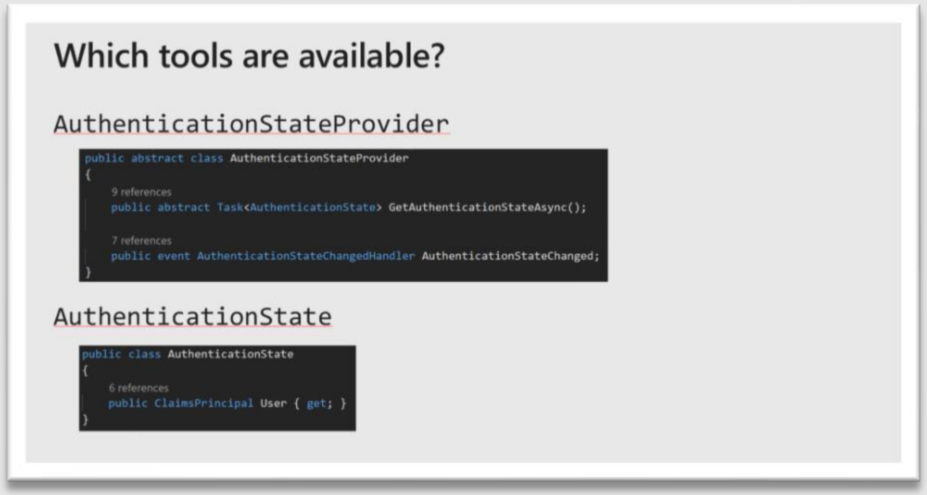
`AuthenticationState`



```
public class AuthenticationState
{
    6 references
    public ClaimsPrincipal User { get; }
}
```

In Blazor Server...





Which tools are available?

AuthenticationStateProvider

```
public abstract class AuthenticationStateProvider
{
    9 references
    public abstract Task<AuthenticationState> GetAuthenticationStateAsync();

    7 references
    public event AuthenticationStateChangedHandler AuthenticationStateChanged;
}
```

AuthenticationState

```
public class AuthenticationState
{
    6 references
    public ClaimsPrincipal User { get; }
}
```

Which providers are supported in Blazor Server?

ASP.NET Core Identity (social providers included)

Azure Active Directory

Azure Active Directory B2C

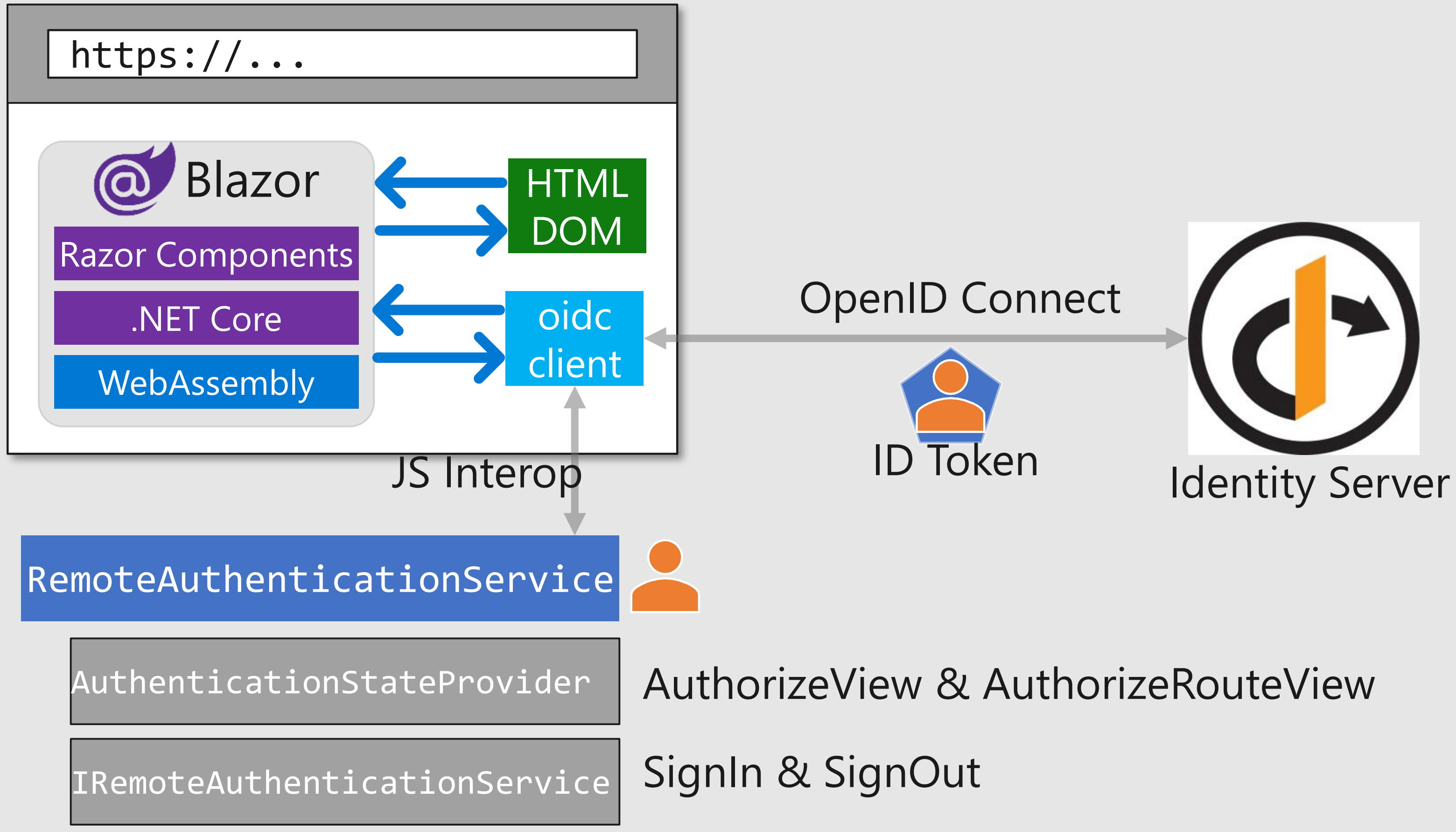
Basically, anything that works with ASP.NET Core

Blazor Security

Demo

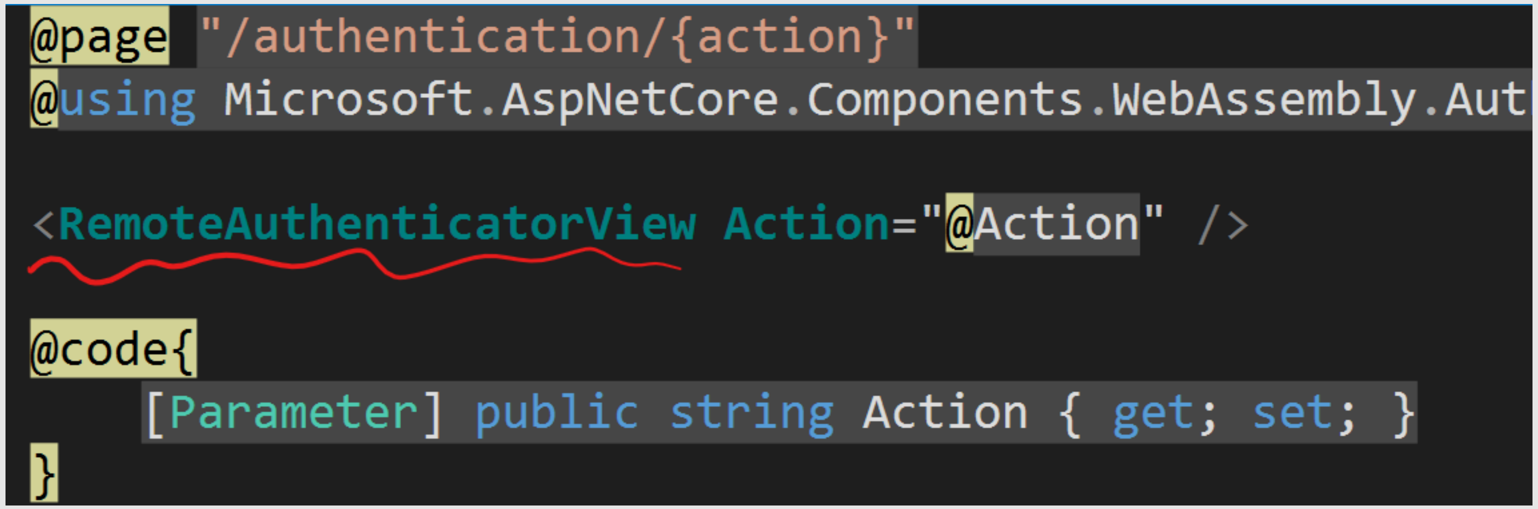
Blazor Server – Individual Accounts

Security in Blazor WASM



Authentication workflow

RemoteAuthenticatorView



```
@page "/authentication/{action}"
@using Microsoft.AspNetCore.Components.WebAssembly.Aut

<RemoteAuthenticatorView Action="@Action" />

@code{
    [Parameter] public string Action { get; set; }
}
```

Actions:

- Login
- Profile
- Register
- Logout

Which providers are supported in Blazor WASM?

Identity Server

Azure Active Directory

Azure Active Directory B2C

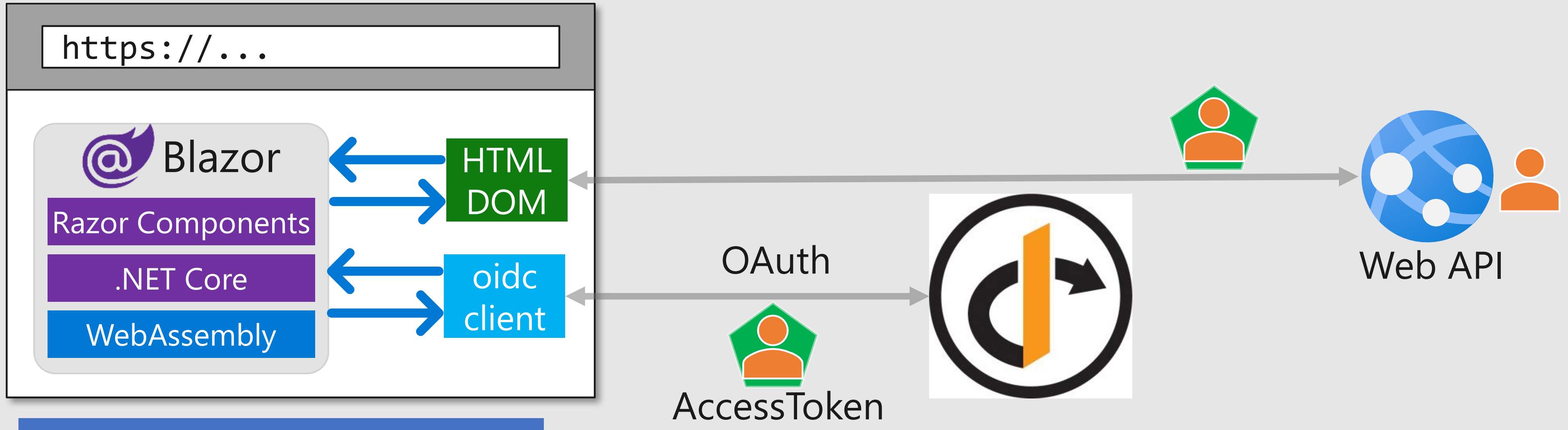
Basically, anything that implements OpenID Connect and OAuth2

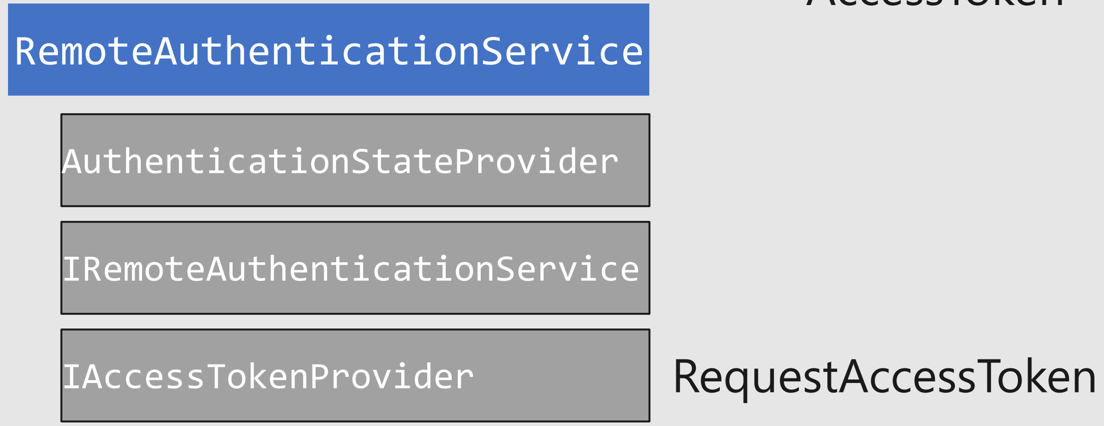
Blazor Security

Demo

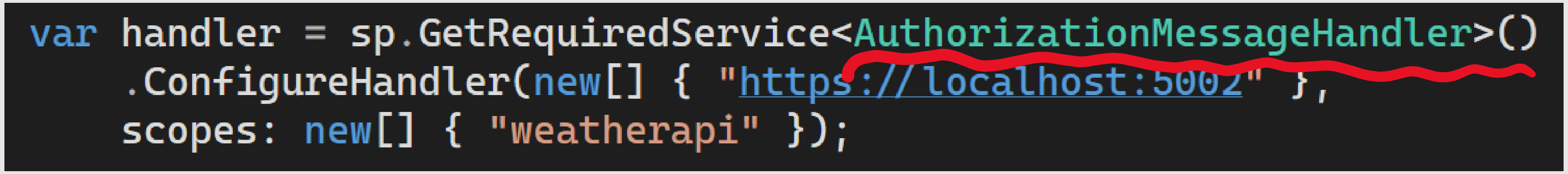
Blazor Wasm and IdentityServer

What if we need to call an API?




RemoteAuthenticationService
AuthenticationStateProvider
IRemoteAuthenticationService
IAccessTokenProvider
RequestAccessToken

It’s integrated in the HttpClient pipeline



```
var handler = sp.GetRequiredService<AuthorizationMessageHandler>()
    .ConfigureHandler(new[] { "https://localhost:5002" },
    scopes: new[] { "weatherapi" });
```

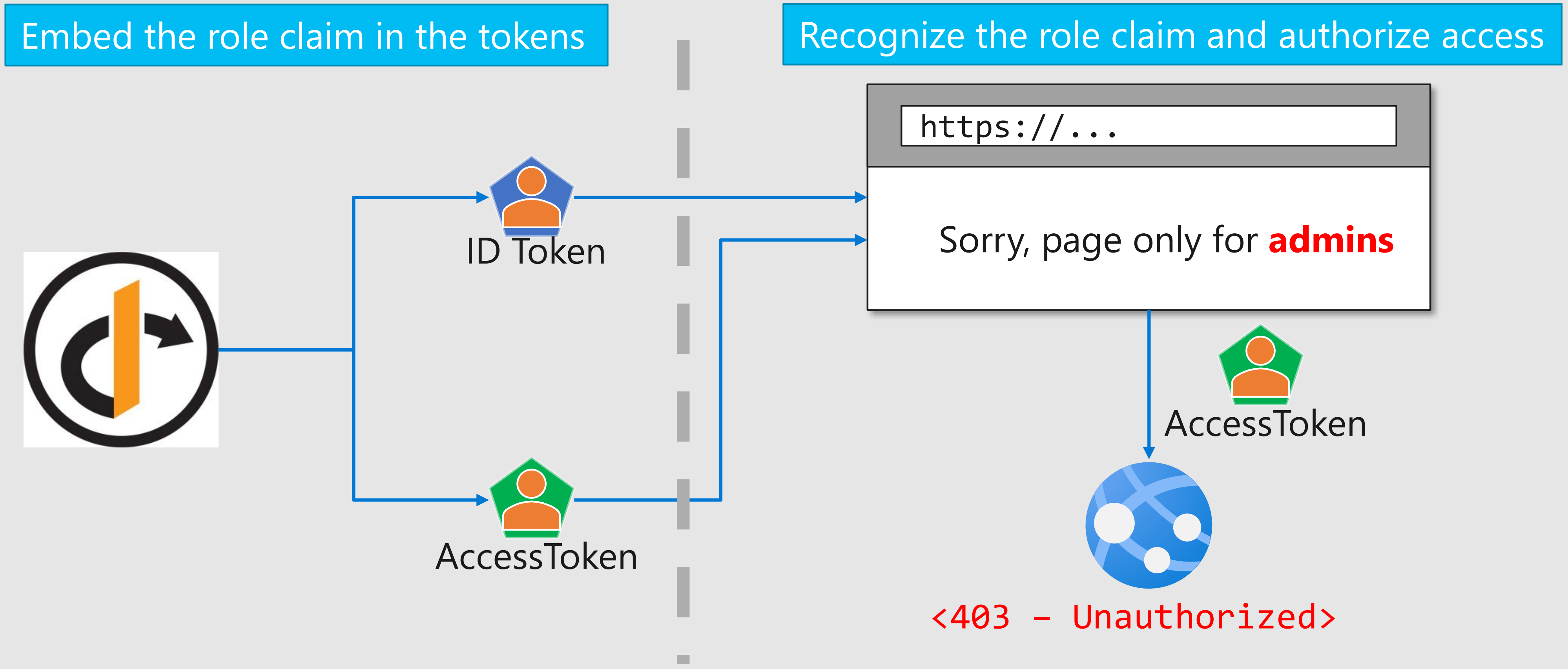
It automatically injects an Access Token into the Http request

Blazor Security
Demo

Authorize a Web API call with OAuth 2

Let’s talk about roles

A role is just a claim like any others


Embed the role claim in the tokens
Recognize the role claim and authorize access
https://...
Sorry, page only for **admins**
ID Token
AccessToken
AccessToken
<403 – Unauthorized>

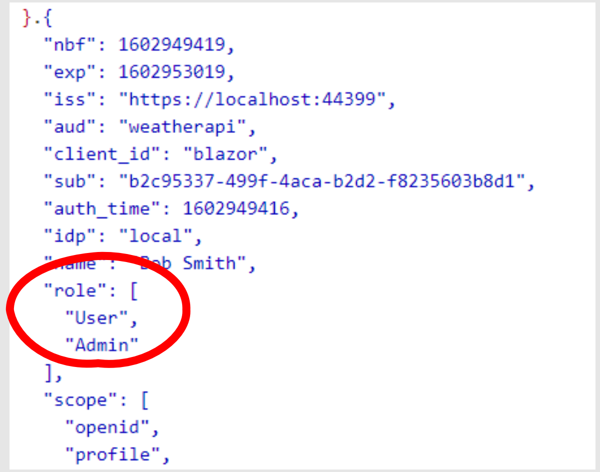
Blazor Security

Demo

Role-based authorization

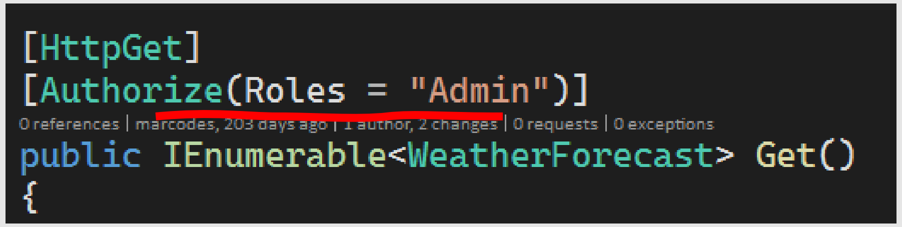
What’s wrong with roles

- A user is assigned to one or more roles
- We receive these roles as claims



```
}.{
  "nbf": 1602949419,
  "exp": 1602953019,
  "iss": "https://localhost:44399",
  "aud": "weatherapi",
  "client_id": "blazor",
  "sub": "b2c95337-499f-4aca-b2d2-f8235603b8d1",
  "auth_time": 1602949416,
  "idp": "local",
  "name": "Bob Smith",
  "role": [
    "User",
    "Admin"
  ],
  "scope": [
    "openid",
    "profile",
```

- We use the roles to authorize API endpoints



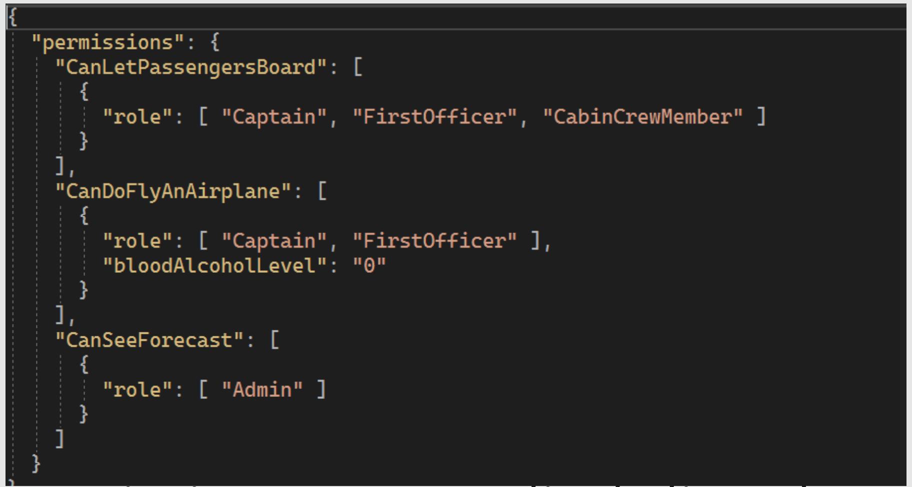
```
[HttpGet]
[Authorize(Roles = "Admin")]
0 references | marcodes, 203 days ago | 1 author, 2 changes | 0 requests | 0 exceptions
public IEnumerable<WeatherForecast> Get()
{
```

What’s wrong with roles

- Roles are sparse across the entire codebase: no holistic view of permissions
- Refactoring permissions is hard, ensuring consistency is even harder (e.g. do we have endpoints available to “Users” but not to “Admins”?)
- There’s no easy way for the FrontEnd to know what a user can or cannot do and modify the user interface accordingly.

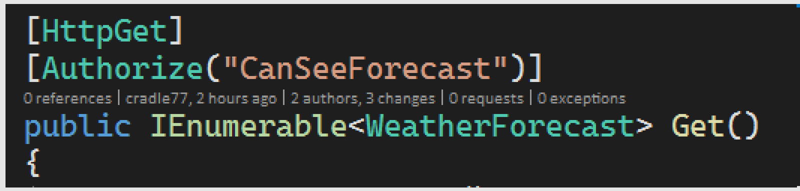
Enter Permissions!

- A centralized place to describe functional permissions

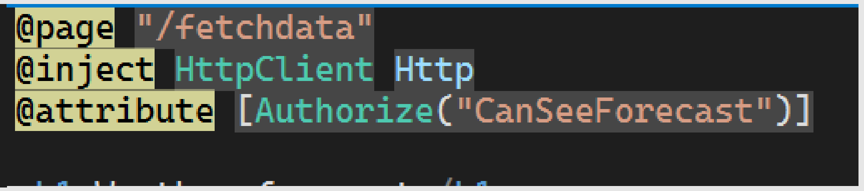


```
{
  "permissions": {
    "CanLetPassengersBoard": [
      {
        "role": [ "Captain", "FirstOfficer", "CabinCrewMember" ]
      }
    ],
    "CanDoFlyAnAirplane": [
      {
        "role": [ "Captain", "FirstOfficer" ],
        "bloodAlcoholLevel": "0"
      }
    ],
    "CanSeeForecast": [
      {
        "role": [ "Admin" ]
      }
    ]
  }
```

- Permissions are applied directly to controllers and pages



```
[HttpGet]
[Authorize("CanSeeForecast")]
0 references | cradle77, 2 hours ago | 2 authors, 3 changes | 0 requests | 0 exceptions
public IEnumerable<WeatherForecast> Get()
{
```



```
@page "/fetchdata"
@inject HttpClient Http
@attribute [Authorize("CanSeeForecast")]
```

- A `/mypermissions` endpoint to retrieve what a user can do
- Inspired by https://github.com/PolicyServer

Blazor Security

Demo

Permissions in Blazor and ASP.NET Core

Some more info on my blog

Securing Blazor WebAssembly with Identity Server 4

https://medium.com/@marcodesanctis2/securing-blazor-webassembly-with-identity-server-4-ee44aa1687ef

Role-based security with Blazor and Identity Server 4

https://medium.com/@marcodesanctis2/role-based-security-with-blazor-and-identity-server-4-aba12da70049

…more coming soon…

Thank you! ☺

@crad77
info@marcodesanctis.it

Get the bits at
https://github.com/cradle77/BlazorSecurityDemos