

**Consent****Details****About****This website uses cookies**

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary**Preferences****Statistics****Marketing****Deny****Allow selection****Allow all**

Configuring mixed authentication and authorization in Optimizely CMS using Okta and ASP.NET Identity.

Securing an Optimizely CMS has always been a key consideration in nearly every project. While the built-in ASP.NET Identity provider is available, many organizations prefer cloud-based identity solutions that can centrally manage users and roles.

In some cases, these two approaches need to work together – combining local ASP.NET Identity



Cookiebot
by Usercentrics

Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



- **API:** for auth of API endpoint requests

Okta authentication uses two schemas:

- **Okta MVC:** linked to the CMS schema
- **Okta JWT:** linked to the API schema

Additionally, there are:



Consent	Details	About
---------	---------	-------

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary

Preferences

Statistics

Marketing

```
var identityCookieName = config.Get(AuthScheme.Identity).Cookie.Name;

if (string.IsNullOrWhiteSpace(identityCookieName))
{
    return false;
}
```

```
        }

    return context.Request.Cookies.ContainsKey(identityCookieName);
}
```



Cookiebot
by Usercentrics

Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



```
        signingCertificate: null,
        encryptionCertificate: null,
        createSchema: true,
        o =>
```

```
{
```

```
    o.RequireHttps = !isDevelopment;
```

```
    // OIDC application for the content management API
```

```
    // uses the client credentials flow
```



Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



```
PostLogoutRedirectUri = options.SignOutRedirectUrl,  
Scope = new List<string> { "openid", "profile", "email" },  
GetClaimsFromUserInfoEndpoint = true,  
OpenIdConnectEvents = new OpenIdConnectEvents  
{
```

```
OnRedirectToIdentityProvider = context => {
    // Prevent caching
    context.Response.Headers.CacheControl = "no-store, no-tr
    context.Response.Headers["CDN-Cache-Control"] = "no-stor

    // Prevent redirect loop
    context.Response.StatusCode = 302;
    context.Response.Headers.Location = "https://www.okta.com";
}
```



Cookiebot
by Usercentrics

Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



The **Okta.AspNetCore** NuGet package is used here, providing the **AddOktaMvc** extension method to configure authorization server details along with the required scopes, which at minimum

include **openid**, **profile**, and **email**. Additionally, the **getClaimsForUserInfoEndpoint** flag ensures that the **userInfo** endpoint is called to retrieve all available claims.

Notably, several authentication events handle specific modifications and edge cases. The **OnRedirectToIdentityProvider** event prevents caching and redirect loops and ensures that a 401



Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



appropriate access rights to the CMS interface.

```
private static async Task SynchronizeRoles(HttpContext httpContext, ClaimsPr
{
```

```
if (principal?.Identity is not ClaimsIdentity claimsIdentity)
{
    return;
}

var optiClaims = new List<Claim>
```



Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



□	grid	dbo.tblSyncedUser
□	folder	Columns
▪	pkID	(PK, int, not null)
▪	UserName	(nvarchar(255), not null)
▪	LoweredUserName	(nvarchar(255), not null)
▪	Email	(nvarchar(255), null)
▪	GivenName	(nvarchar(255), null)



Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



```
return services.PostConfigureAll<OpenIdConnectOptions>(o =>
{
    var validationParameters = GetTokenValidationParameters(AuthScheme.O
    validationParameters.ValidIssuer = options.Domain;
    validationParameters.ValidAudience = options.ClientId;
```

```
        o.TokenValidationParameters = validationParameters;
    });
}
```



Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



were present in **OnTokenReceived**.

```
private static AuthenticationBuilder AddApiAuth(this AuthenticationBuilder a  
    authenticationBuilder
```

```
.AddJwtBearer(AuthScheme.OktaJwt, o =>
{
    o.Authority = UrlHelper.CreateIssuerUrl(options.Domain, Authoriz
    o.Audience = options.ApiAudience;
    o.RequireHttpsMetadata = true;
    o.TokenValidationParameters = GetTokenValidationParameters(AuthS
```



Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



```
{
    o.DefaultAuthenticateScheme = AuthScheme.Cms;
    o.DefaultChallengeScheme = AuthScheme.Cms;
})
.AddOktaAuth(options.Okta)
```

```
.AddPolicyScheme(AuthScheme.Cms, null, o =>
{
    o.ForwardDefaultSelector = httpContext =>
        httpContext.ContainsIdentityAuthCookie()
            ? AuthScheme.Identity
            : AuthScheme.OktetoMyc...
```



Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



```
{
    appBuilder.Run(async httpContext =>
    {
        if (httpContext.ContainsIdentityAuthCookie())
        {
```

```
    var signInManager = httpContext.RequestServices.GetRequiredService<SignInManager>();
    await signInManager.SignOutAsync();
}
else
{
    await httpContext.SignOutAsync();
}
```



Cookiebot
by Usercentrics

Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



Cancelling CMS Scheduled Jobs

From flags to tokens: making Optimizely CMS scheduled jobs more



Consent

Details

About

This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary



Preferences



Statistics



Marketing



Getting 404 when expecting 401

A short story about the mysterious behavior of an API in headless architecture.



This website uses cookies

We use cookies and similar technologies to enhance your browsing experience, analyze website traffic, and personalize content. By clicking "Allow All," you agree to the storing of cookies on your device to improve site navigation, provide social media features, and assist in our marketing efforts.

Necessary

Preferences

Statistics

Marketing