# **认证服务器配置**

# **修改客户端配置**

**new** Client

{

ClientId = "mvc",

ClientName = "MVC Client",

AllowedGrantTypes = GrantTypes.HybridAndClientCredentials,

ClientSecrets =

{

**new** Secret("secret".Sha256())

},

RedirectUris = { "http://localhost:5002/signin-oidc" },

PostLogoutRedirectUris = { "http://localhost:5002/signout-callback-oidc" },

AllowedScopes =

{

IdentityServerConstants.StandardScopes.OpenId,

IdentityServerConstants.StandardScopes.Profile,

"api1"

},

AllowOfflineAccess = **true**

};

# **客户端配置**

# **修改MVC客户端**

.AddOpenIdConnect("oidc", options =>

{

options.SignInScheme = "Cookies";

options.Authority = "http://localhost:5000";

options.RequireHttpsMetadata = **false**;

options.ClientId = "mvc";

options.ClientSecret = "secret";

options.ResponseType = "code id\_token";

options.SaveTokens = **true**;

options.GetClaimsFromUserInfoEndpoint = **true**;

options.Scope.Add("api1");

options.Scope.Add("offline\_access");}

);

# **使用访问令牌**

OpenID Connect中间件会自动为您保存令牌。这就是SaveTokens设置的作用。

从技术上讲，令牌存储在cookie的属性部分中。访问它们的最简单方法是使用Microsoft.AspNetCore.Authentication命名空间中的扩展方法。

例如，在您的声明视图中：

<dt>access token</dt><dd>@await ViewContext.HttpContext.GetTokenAsync("access\_token")</dd>

<dt>refresh token</dt><dd>@await ViewContext.HttpContext.GetTokenAsync("refresh\_token")</dd>

要使用访问令牌访问API，您需要做的就是检索令牌，并在HttpClient上设置它：

**public** **async** Task<IActionResult> CallApiUsingUserAccessToken(){

var accessToken = **await** HttpContext.GetTokenAsync("access\_token");

var client = **new** HttpClient();

client.SetBearerToken(accessToken);

var content = **await** client.GetStringAsync("http://localhost:5001/identity");

ViewBag.Json = JArray.Parse(content).ToString();

**return** View("json");

}