

Migrating to Swashbuckle.AspNetCore version 5

Asked 4 months ago Active 1 month ago Viewed 2k times

I'm trying to migrate from version 4.0.1 to 5.0.0-rc2 of Swashbuckle in a .NET Core 3 Preview 5 Web API project.

11 I've got the project compiling and the Swagger UI working, but I can't get Bearer authentication to work, which I think is due to me not setting up the new format security correctly.

This is my old code that worked in version 4:

★
3

```
c.AddSecurityDefinition("Bearer", new ApiKeyScheme
{
    Description = "JWT Authorization header using the Bearer scheme. \r\n\r\n Enter
'Bearer' [space] and then your token in the text input below.\r\n\r\nExample: \"Bearer
12345abcdef\"",
    Name = "Authorization",
    In = "header",
    Type = "apiKey"
});

var security = new Dictionary<string, IEnumerable<string>>
{
    {"Bearer", new string[] { }},
};

c.AddSecurityRequirement(security);
```

And this is what I've changed it to for v5:

```
c.AddSecurityDefinition("Bearer", new OpenApiSecurityScheme
{
    Description = "JWT Authorization header using the Bearer scheme. \r\n\r\n Enter
'Bearer' [space] and then your token in the text input below.\r\n\r\nExample: \"Bearer
12345abcdef\"",
    Name = "Authorization",
    In = ParameterLocation.Header,
    Type = SecuritySchemeType.ApiKey,
    Scheme = "bearerAuth"
```

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

```
{
    {
        new OpenApiSecurityScheme
        {
            Reference = new OpenApiReference {
                Type = ReferenceType.SecurityScheme,
                Id = "tomsAuth" }
            }, new List<string>() }
    });
```

I think my issue is probably in this part of the code:

```
new OpenApiSecurityScheme
{
    Reference = new OpenApiReference {
        Type = ReferenceType.SecurityScheme,
        Id = "tomsAuth" }
    }, new List<string>() }
```

I think that bit should probably have "Bearer" in it somewhere, but I'm not sure where?

Additional info

This is how I'm setting up the JWT authentication in the first place. This code hasn't changed and was working when I was using Swashbuckle 4.0.1:

```
var appSettings = appSettingsSection.Get<AppSettings>();
var key = Encoding.ASCII.GetBytes(appSettings.Secret);

services.AddAuthentication(x =>
{
    x.DefaultAuthenticateScheme = JwtBearerDefaults.AuthenticationScheme;
    x.DefaultChallengeScheme = JwtBearerDefaults.AuthenticationScheme;
})
.AddJwtBearer(x =>
{
    x.Events = new JwtBearerEvents
    {
        OnTokenValidated = context =>
        {
            var userService =
                appServices.GetService<IUserService>();
            var token = context.Token;
```

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

```

        if (user == null)
        {
            // return unauthorized if user no longer exists
            context.Fail("Unauthorized");
        }

        return Task.CompletedTask;
    }
};
x.RequireHttpsMetadata = false;
x.SaveToken = true;
x.TokenValidationParameters = new TokenValidationParameters
{
    ValidateIssuerSigningKey = true,
    IssuerSigningKey = new SymmetricSecurityKey(key),
    ValidateIssuer = false,
    ValidateAudience = false
};
});

```

c#

swagger-ui

swashbuckle

edited May 26 at 6:26



JL2210

5,205

4

12

45

asked May 21 at 8:47



tomRedox

6,988

4

40

79

2 Answers



Got this working in the end by trial and error. This is the code that works for me:

30



```

c.AddSecurityDefinition("Bearer", new OpenApiSecurityScheme
{
    Description =
        "JWT Authorization header using the Bearer scheme. \r\n\r\n Enter 'Bearer'
[space] and then your token in the text input below.\r\n\r\nExample: \"Bearer
12345abcdef\"",
    Name = "Authorization",
    In = ParameterLocation.Header,

```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```
c.AddSecurityRequirement(new OpenApiSecurityRequirement()
{
    {
        new OpenApiSecurityScheme
        {
            Reference = new OpenApiReference
            {
                Type = ReferenceType.SecurityScheme,
                Id = "Bearer"
            },
            Scheme = "oauth2",
            Name = "Bearer",
            In = ParameterLocation.Header,
        },
        new List<string>()
    }
});
```

I suspect there are probably properties being set there that don't actually need to be explicitly set, but the above is working for me.

answered May 22 at 8:19



tomRedox

6,988 4 40 79

1 It works for me under .net core 3.0 as well – [Taras Pelenio](#) Sep 27 at 13:13

6

OpenAPI 3.0 comes with Bearer authentication, which is a security scheme with type: http and scheme: bearer.

So instead of using an API key scheme you have to set the security scheme type to HTTP Authentication and then define the name of the HTTP Authorization scheme as defined in [RFC7235](#). In this case "bearer".

After you've defined the security scheme you can apply it by adding it as a security requirement.

```
//First we define the security scheme
c.AddSecurityDefinition("Bearer", //Name the security scheme
    new OpenApiSecurityScheme{
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```
Scheme = "bearer" //The name of the HTTP Authorization scheme to be used in the
Authorization header. In this case "bearer".
});

c.AddSecurityRequirement(new OpenApiSecurityRequirement{
    {
        new OpenApiSecurityScheme{
            Reference = new OpenApiReference{
                Id = "Bearer", //The name of the previously defined security scheme.
                Type = ReferenceType.SecurityScheme
            }
        }, new List<string>()
    }
});
```

edited Sep 10 at 18:37

answered Sep 10 at 14:22



Pavlos

85 1 6

1 @B--rian I've updated my answer, hope this is sufficient. I'd be happy to elaborate more. – Pavlos Sep 10 at 17:43

1 @Pavlos thanks, that's a really helpful answer, the comments really help – tomRedox Sep 10 at 18:55

3 Note that when doing this the text to be pasted in the textfield no longer needs to be prefixed with the string "Bearer " – Daniel Sep 27 at 17:27

protected by [Community](#) ♦ Jun 8 at 0:11

Thank you for your interest in this question. Because it has attracted low-quality or spam answers that had to be removed, posting an answer now requires 10 [reputation](#) on this site (the [association bonus](#) does not count).

Would you like to answer one of these [unanswered questions](#) instead?