

# REST Services – Cross-Origin Resource Sharing

Peter Levinsky IT Roskilde

17.09.2020

Situation:

# Solution:

- Jsonp – add program lines to the script i.e. easy to work around
- Cors – negotiation between browser and REST service – Go for today (more secure)

# How CORS works:

Two types:

1. Simple request (Get-requests)
1. Preflighted request (All others)

# Simple Request (Get):

- For simple request i.e. GET, HEAD or POST.  
Note: POST only if  
content type = application/x-www-form-urlencoded | multipart/form-data | text/plain  
*i.e. NOT JSON*
- Request:
- **Host:** api.mydomain.com  
**Origin:** https://www.mydomain.com
- Response:
- **Access-Control-Allow-Origin:** https://www.mydomain.com  
**Content-Type:** application/json

# Preflighted Request:

- **CORS – Handshake -- Before the REST – Call**
- Request:
- **OPTIONS** /widgets/ HTTP/1.1  
Host: api.mydomain.com  
**Origin:** https://www.mydomain.com  
**Access-Control-Request-Method:** POST – could be a list  
**Access-Control-Request-Headers:** Authorization, Content-Type
- 
- Response:
- HTTP/1.1 200 OK  
**Access-Control-Allow-Origin:** https://www.mydomain.com  
**Access-Control-Allow-Methods:** POST, GET, OPTIONS  
**Access-Control-Allow-Headers:** Authorization, Content-Type  
Content-Type: application/json
- Possibly: Access-Control-Max-Age: 600 → 600 sec = 10 min

# Preflighted Request #2:

- **Data**
  
- Request:
- POST /widgets/ HTTP/1.1  
Host: api.mydomain.com  
Authorization: 1234567  
Content-Type: application/json  
**Origin:** https://www.mydomain.com → equal to domain in handshake
  
- 
- Response:
- HTTP/1.1 200 OK  
**Access-Control-Allow-Origin:** https://www.mydomain.com  
Content-Type: application/json

# How to CORS in C# .Net Core:

- Three approaches
  1. Quick, but not so configurable
  2. Detailed setup
  3. Granulated setup
- ***A Quick***
- When creating your azure web-service – configure azure web service to ‘allow all’  
Works when running in azure.



# How to CORS in C# .Net Core - detailed:

- 3 steps

1. Install NuGet-package (Microsoft.AspNetCore.Cors)

2. In the Startup.cs – file

ConfigureServices method Add services.AddCors();

3. In the Startup.cs – file

Configure method Add

app.UseCors(  
options =>

{

options.AllowAnyOrigin().AllowAnyMethod().AllowAnyHeader();

// allow everything from anywhere

});

Alter:

options.WithMethods("GET", "PUT").AllowAnyOrigin().AllowAnyHeader();

# How to CORS in C# .Net Core - Granulated:

1. Install NuGet-package (Microsoft.AspNetCore.Cors)
2. In the Startup.cs – file  
ConfigureServices method Add services.AddCors(options =>  
{  
options.AddPolicy("AllowAll",  
builder => builder.AllowAnyOrigin());  
});
3. In the Startup.cs – file  
Configure method Add  
app.UseCors("AllowAll");
4. Annotations  
[EnableCors("AllowAll")]

# Situation:

# Situation:

# Situation:

# Situation:

# Situation:

# Situation: