



Swissdec @ GovTech Hackathon 2024

Erkenntnisse

- Das eigentliche Problem liegt woanders (SOAP mit Sicherheit in .Net ist schmerzhaft)
- Swisdec muss gewisse Entscheidungen überdenken
- Die Challenge muss attraktiver sein, um mehr zu mobilisieren

Prototyp API

The screenshot shows a web browser window with the Swagger UI interface. The browser's address bar displays the URL `https://localhost:44380/swagger/index.html`. The Swagger logo is visible in the top left corner, with the text "Supported by SMARTBEAR" below it. A dropdown menu labeled "Select a definition" is open, showing the selected API definition "Swissdec.SUA.Api v1". Below this, the API title "Swissdec.SUA.Api" is displayed, followed by the version "1.0" and the specification type "OAS3". The URL `https://localhost:44380/swagger/v1/swagger.json` is shown below the title. The main content area displays the API endpoint "Swissdec.SUA.Api" and the method "POST /certificate/request".

Swagger UI

https://localhost:44380/swagger/index.html

Swagger
Supported by SMARTBEAR

Select a definition **Swissdec.SUA.Api v1**

Swissdec.SUA.Api 1.0 OAS3
<https://localhost:44380/swagger/v1/swagger.json>

Swissdec.SUA.Api

POST /certificate/request

Prototyp Code

```
1 using System.Security.Cryptography;
2 using System.Security.Cryptography.X509Certificates;
3
4 1 reference | - changes | -authors, -changes
5 static X509Certificate2 CreateSelfSignedCertificate(string subjectName)
6 {
7     using var rsaKey = RSA.Create(2048);
8     var certificateRequest = new CertificateRequest(subjectName, rsaKey, HashAlgorithmName.SHA256, RSASignaturePadding.Pkcs1);
9     var notBefore = DateTimeOffset.UtcNow;
10    var notAfter = notBefore.AddMonths(1);
11    return certificateRequest.CreateSelfSigned(notBefore, notAfter);
12 }
13
14 var builder = WebApplication.CreateBuilder(args);
15 builder.Services.AddEndpointsApiExplorer();
16 builder.Services.AddSwaggerGen();
17
18 var app = builder.Build();
19 app.UseSwagger();
20 app.UseSwaggerUI();
21 app.MapPost("/certificate/request", async (CertificateRequestModel certificateRequestModel, HttpContext httpContext) =>
```

Prototyp Client

```
Eingabeaufforderung
$curl "https://localhost:44380/certificate/request" -X POST -H "Content-Type: application/json" -d
"{\"commonName\": \"GovTechHackaton2024\", \"pfxpassword\": \"iamgod\"}" -o GovTechHackaton2024.pfx
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100  2442    0  2382  100    60  12627    318  --:--:--  --:--:--  --:--:-- 12989

$certutil -dump GovTechHackaton2024.pfx
Geben Sie das PFX-Kennwort ein:

===== Zertifikat 0 =====
===== Verschachtelungsebene 1 anfangen =====
Element 0:
Seriennummer: a1d39a8a99f6b279
Aussteller: CN=GovTechHackaton2024
```