Swapindo

Installationsbeschreibung Mutual Authentication

von
Benjamin Ellmer (S1910237013)

March 19, 2022

Contents

1 Vorwort

Das Ergebnis meines Projekts ist eine Library für die Mutual Authentication geworden. In diesem Dokument finden Sie eine Beschreibung, wie diese Library verwendet werden kann. In der Abgabe unter Implementierung finden Sie 4 Ordner: KeyManagment, JWT, mTLS und MutualAuthenticationLibrary. KeyManagment enthält alle files für das key management und eine Beispiel CA mit Beispiel keys. Das Passwort für alle files ist B3njam1n. Unter JWT und mTLS finden Sie Beispielprojekte, die die Authentifizierungsmechanismen implementieren.

2 Installation

Library zu Projektmappe hinzufügen:
Projektmappe -> Hinzufügen -> Vorhandenes Projekt

Library zu Projektmappe hinzufügen:
Abhängigkeiten -> Projektverweis hinzufügen -> MutualAuthentication-Library

3 Key Management

Keys für die CA erstellen, mittels:
./create-ca.sh *$password $subject*

Keys für einen Service erstellen, mittels:
./gen-service-keys.sh *$passwordService $passwordCA $subject $caPath*

Die pfx files der services müssen dann in die Projektverzeichnisse der services kopiert werden. Das ca.crt file muss auch in das Projektverzeichnisse aller services kopiert werden.

4 Implementierung mTLS

4.1 Server

Zertifikatauthentifizierung erlauben (Program.cs)

```
public static IHostBuilder CreateHostBuilder(string[] args) =>
    Host.CreateDefaultBuilder(args)
        .ConfigureWebHostDefaults(webBuilder => {
            webBuilder.UseStartup<Startup>();
            webBuilder.ConfigureKestrel(kestrelOptions => {
                kestrelOptions.ConfigureHttpsDefaults(httpOptions => {
                    httpOptions.ClientCertificateMode = ClientCertificateMode.AllowCertificate;
                });
            });
        });
```

Certificate Authority Service hinzufügen (Startup.cs)

```
services.AddSingleton<ICertificateAuthorityService>(context => new CertificateAuthorityService(options => {
    options.Add(new X509Certificate2(@"ca.crt", "B3njam1n"));
}));
```

MTLS Konfigurieren (Startup.cs)

```
// Setup MTLS Authentication
services.AddAuthentication(CertificateAuthenticationDefaults.AuthenticationScheme).AddCertificate(options => {
    options.AllowedCertificateTypes = CertificateTypes.Chained;
    options.RevocationMode = X509RevocationMode.NoCheck;
    options.Events = new MTLSAuthenticationEvents();
});
```

MTLS verwenden (Startup.cs)

```
app.UseAuthentication();
app.UseAuthorization();
```

4.2 Client

MTLS Client hinzufügen (Startup.cs)

```
// Inject HttpClient, that stores a certificate in the ClientCertificate field
services.AddCertificateHttpClient("mTLSClient", new X509Certificate2(@"adService.pfx", "B3njam1n"));
```

MTLS Client injecten (...Controller.cs)

```
public AdsController(IHttpClientFactory factory) {
    _httpClient = factory.CreateClient("mTLSClient");
}
```

5 Implementierung self-signed JWT

5.1 Server

Zertifikatauthentifizierung erlauben (Program.cs)

```
public static IHostBuilder CreateHostBuilder(string[] args) =>
    Host.CreateDefaultBuilder(args)
        .ConfigureWebHostDefaults(webBuilder => {
            webBuilder.UseStartup<Startup>();
            webBuilder.ConfigureKestrel(kestrelOptions => {
                kestrelOptions.ConfigureHttpsDefaults(httpOptions => {
                    httpOptions.ClientCertificateMode = ClientCertificateMode.AllowCertificate;
                });
            });
        });
```

Certificate Authority Service hinzufügen (Startup.cs)

```
services.AddSingleton<ICertificateAuthorityService>(context => new CertificateAuthorityService(options => {
    options.Add(new X509Certificate2(@"ca.crt", "B3njam1n"));
}));
```

Token Validation Service hinzufügen (Startup.cs)

```
services.AddSingleton<ITokenValidationService>(context => {
    return new JWTValidationService("adservice");
});
```

Token Authentifizierung aktivieren (Startup.cs)

```
app.UseAuthorization();

app.UseJWTAuthentication();

app.UseEndpoints(endpoints => {
    endpoints.MapControllers();
});
```

5.2 Client

Token Creator Service hinzufügen (Startup.cs)

```
services.AddSingleton<ITokenCreatorService>(context => {
    return new JWTCreatorService(new X509Certificate2(@"adService.pfx", "B3njam1n"), options => {
        options.DefaultValidityPeriod = new System.TimeSpan(0, 1, 0);
    });
});
```

Http Clients hinzufügen (Startup.cs)

```
services.AddHttpClient("clientNoCertificate");
services.AddCertificateHttpClient("clientWithCertificate", new X509Certificate2(@"adService.pfx", "B3njam1n"));
```

Http Clients Injecten (...Controller.cs)

```
public AdsController(IHttpClientFactory factory) {
    _httpClientNoCert = factory.CreateClient("clientNoCertificate");
    _httpClientWithCert = factory.CreateClient("clientWithCertificate");
}
```