

## Program szkolenia:

# Bezpieczeństwo serwisów ASP.NET Web API

## Informacje:

<b>Nazwa:</b>	<b>Bezpieczeństwo serwisów ASP.NET Web API</b>
<b>Kod:</b>	<b>NET-WebAPI</b>
<b>Kategoria:</b>	.NET
<b>Grupa docelowa:</b>	developerzy
<b>Czas trwania:</b>	2 dni
<b>Forma:</b>	50% wykłady / 50% warsztaty

---

Zaimplementowałeś doskonałe API zgodne z filozofią REST i chciałbyś je zabezpieczyć przed dostępem niepowołanych użytkowników, ale nie wiesz od czego zacząć? Ten kurs wprowadzi Cię w podstawy budowy bezpiecznych serwisów REST z wykorzystaniem biblioteki ASP.NET Web API.

Program szkolenia kładzie szczególny nacisk na najważniejsze zagadnienia związane z uwierzytelnianiem i autoryzacją klientów i użytkowników Twojego API.

Szkolenie jest przeznaczone dla projektantów i programistów tworzących serwisy REST i aplikacje SPA.

## Zalety szkolenia:

- Bezpieczeństwo serwisów REST w ASP.NET Web API
- Nacisk na nowoczesne aplikacje Webowe (SPA)
- Praktyczne gotowe do zastosowania wskazówki

## Szczegółowy program:

### 1. Bezpieczeństwo protokołu HTTP - bronimy się przed atakiem poprzez sieć

#### 1.1. Sieć jako wektor ataku

##### 1.1.1. Modyfikujemy żądania HTTP za pomocą serwera proxy

#### 1.2. Bezpieczeństwo w warstwie transportowej

#### 1.3. Certyfikaty X.509

#### 1.4. Protokół SSL

### 2. Jak działa ASP.NET Web API?

#### 2.1. Potok przetwarzania OWIN

#### 2.2. Filtr uwierzytelniający

#### 2.3. Filtr autoryzujący

### 3. Podstawowe tryby uwierzytelniania - bronimy się przed obcymi użytkownikami

#### 3.1. Windows Authentication

##### 3.1.1. Studium przypadku: aplikacja Webowa w firmowym intranecie

#### 3.2. Basic Authentication

##### 3.2.1. Studium przypadku: aplikacja Webowa w Internecie

##### 3.2.2. Problemy z mechanizmem Basic Authentication

#### 3.3. Sprawdzamy tożsamość klienta na podstawie jego certyfikatu X.509

##### 3.3.1. Studium przypadku: integracja systemów w korporacji

### 4. Single Page Applications -- implementujemy logowanie poprzez konto Google

#### 4.1. Czy użytkownik na pewno potrzebuje konta w naszym systemie?

#### 4.2. Bezpieczne logowanie użytkownika krok po kroku

##### 4.2.1. Włączamy SSL

##### 4.2.2. Rejestrujemy naszą aplikację w konsoli deweloperskiej Google

4.2.3. Konfigurujemy naszą aplikację

4.2.4. Gotowe!

## 5. Autoryzacja - bronimy się przed nadużywaniem uprawnień przez znanych użytkowników

5.1. Klient czy użytkownik?

5.2. Atrybut [Authorize] i [AllowAnonymous]

5.3. Implementujemy własną logikę autoryzacji