

Die Sicherheit von Softwarelösungen fängt schon bei der Entwicklung an. Um die typischen Software-Schwachstellen schon während der Implementierung zu vermeiden, sollten die Entwickler die Entstehung von Schwachstellen kennen und sich das Wissen aneignen, wie diese in der entsprechenden Programmiersprache verhindert werden können.

Dieser C#/.NET Secure Development Kurs vermittelt den Teilnehmern das nötige Wissen, damit sie in Zukunft sichere Applikationen erstellen können. Die Teilnehmer lernen im praktisch orientierten Kurs, was defensive Programmierung bedeutet und wie Sie damit ihre Applikationen nachhaltig absichern können.

Kursinhalt

- Theorie von typischen Software-Schwachstellen
- Identifizieren von Schwachstellen im Übungs-Code
- Beheben von Schwachstellen durch geeignete Programmier-Techniken
- Absichern von Applikationen mittels verfügbaren Funktionen

Vorgehensweise

In den zwei Kurstagen werden folgende Themen jeweils nacheinander wie folgt gemeinsam erarbeitet:

- Einführung Web-Angriffstechniken
- Finden von Schwachstellen im Übungs-Code
- **Hands-on:** Anwenden von C#/.NET Verteidigungsstrategien

Voraussetzungen

- Kenntnisse in C#/.NET-Programmierung
- Vertrautheit mit Web-Technologien (HTML, ASP.Net, WCF, Javascript)
- Grundkenntnisse im Umgang mit IIS
- Eigenes Notebook mit aktueller Entwicklungsumgebung

Teilnehmer

Teilnehmer dieses Kurses sind:

- C#/.NET Entwickler
- Software Architekten

Max. Anzahl Teilnehmer: 10

Ihr Vorteil

- Mit wenig Zeitaufwand erhalten Sie fundiertes Wissen
- Lernen Sie effektiv durch praktische Übungen
- Lernen Sie, wie Sie in Zukunft nachhaltige und sichere .NET Anwendungen entwickeln

Kursort & Zeiten

- Raum Zürich
- Kurszeiten: 9:00 – 16:45
- Firmeninterne Durchführung auf Anfrage

Die Kursdaten finden Sie auf unserer Webseite

Dauer & Preis

- 2-tägiger Kurs
- CHF 2'450.- (pro Teilnehmer, exkl. MwSt.)
- Inkl. Verpflegung

Frühbucher Bonus

Melden Sie sich mindestens 6 Wochen vor der Kursdurchführung an und Sie erhalten als Geschenk ein aktuelles Tablet

Tag 1

Einführung sichere Software

Validierung

Enkodierung

Injection

Insecure Direct Object References

Tag 2

Cross site request forgery (CSRF)

Session Management

Authentisierung und Autorisierung

Error Handling