Code responsibly with generative AI in C# (desktop applications) (CRWGAIC)

ID CRWGAIC **Preis** CHF 2'250.– (exkl. MwSt.) **Dauer** 3 Tage

Zielgruppe

C#-Entwickler, die Copilot oder andere GenAI-Tools verwenden

Voraussetzungen

Allgemeine C#-Entwicklung

Kursziele

- Das Wesentliche der verantwortungsvollen KI verstehen
- Vertrautmachen mit grundlegenden Konzepten der Cybersicherheit
- Ansätze und Grundsätze der Eingabevalidierung
- Ermittlung von Schwachstellen und deren Folgen
- Lernen Sie die besten Sicherheitspraktiken in C#
- Korrekte Implementierung verschiedener Sicherheitsmerkmale
- Verwaltung von Schwachstellen in Komponenten von Drittanbietern
- Verstehen, wie Kryptographie die Sicherheit unterstützt
- Lernen, wie man kryptografische APIs in C# richtig verwendet
- All dies im Kontext von GitHub Copilot

Kursinhalt

Tag 1

Verantwortungsbewusst kodieren mit GenAI

- Was ist verantwortungsvolle KI?
- Was ist Sicherheit?
- Bedrohung und Risiko
- Arten von Cybersicherheitsbedrohungen - die CIA-Triade
- Folgen von unsicherer Software
- Sicherheit und verantwortungsvolle KI in der Softwareentwicklung
- GenAI-Werkzeuge für die Kodierung: Copilot, Codeium und andere

- Überprüfung der Eingaben
  - Input-Validierungsprinzipien
  - Denylisten und Zulassungslisten
  - Was zu validieren ist - die Angriffsfläche
  - Wo soll validiert werden - Verteidigung in der Tiefe
  - Wann validieren - Validierung vs. Umwandlung
- Einspritzung
  - Code-Einspritzung
  - OS-Befehlsinjektion
  - Übung - Befehlsinjektion
  - Bewährte Praktiken zur Injektion von OS-Befehlen
  - Vermeidung von Befehlseingaben mit den richtigen APIs
  - Übung - Bewährte Praktiken der Befehlsinjektion
  - Übung - Experimentieren mit der Befehlsinjektion in Copilot
  - Fallstudie - Befehlsinjektion in Ruckus
- Probleme im Umgang mit Ganzzahlen
  - Darstellung von Zahlen mit Vorzeichen
  - Integer-Visualisierung
  - Integer-Überlauf
  - Labor - Integer-Überlauf
  - Verwirrung mit Vorzeichen / ohne Vorzeichen
  - Fallstudie - Die Stockholmer Börse
  - Labor - Verwechslung mit Vorzeichen / ohne Vorzeichen
  - Labor - Experimentieren mit vorzeichenlosen/vorzeichenbehafteten Verwechslungen in Copilot
  - Ganzzahlige Trunkierung
  - Bewährte Praktiken
  - Upcasting
  - Prüfung der Vorbedingungen
  - Prüfung nach der Bedingung
  - Integer-Verarbeitung in C#
  - Labor - Kontrollierte Arithmetik
  - Übung - Experimentieren mit Integer-Überlauf in Copilot
- Dateien und Datenströme
  - Pfadüberquerung
  - Übung - Pfadüberquerung
  - Zusätzliche Herausforderungen in Windows
  - Fallstudie - Dateispoofing in WinRAR
  - Bewährte Verfahren zur Pfadüberquerung
  - Labor - Kanonisierung von Pfaden

- Übung - Experimentieren mit der Pfadverfolgung in Copilot

Tag 2

Überprüfung der Eingaben

- Unsichere Reflexion
  - Reflexion ohne Validierung
  - Labor - Unsichere Reflexion
  - Übung - Experimentieren mit unsicherer Reflexion in Copilot
- Unsicherer nativer Code
  - Abhängigkeit von nativem Code
  - Übung - Unsicherer nativer Code
  - Bewährte Praktiken für den Umgang mit nativem Code
- Sicherheitsmerkmale
  - Authentifizierung
    - Grundlagen der Authentifizierung
    - Multi-Faktor-Authentifizierung (MFA)
    - Fallstudie - Der InfinityGauntlet-Angriff
    - Zeitbasierte Einmal-Passwörter (TOTP)
  - Passwortverwaltung
    - Verwaltung eingehender Passwörter
    - Speichern von Kontopasswörtern
    - Passwort im Transit
    - Labor - Reicht das Hashing von Passwörtern aus?
    - Wörterbuchangriffe und Brute-Forcing
    - Salzen
    - Adaptive Hash-Funktionen für die Passwortspeicherung
    - Übung - Verwendung adaptiver Hash-Funktionen in C#
    - Übung - Verwendung adaptiver Hash-Funktionen in Copilot
    - Fallstudie - Fehlende Authentifizierung und Klartext-Passwortspeicherung bei Veeam
    - Passwort-Politik
    - NIST-Authentifikator-Anforderungen für gespeicherte Geheimnisse
    - Migration der Passwort-Datenbank
    - Fest kodierte Passwörter
    - Bewährte Praktiken
    - Labor - Hartkodiertes Passwort
  - Schutz sensibler Informationen im Speicher
    - Herausforderungen beim Schutz der Erinnerung
    - Fallstudie - Diebstahl geheimer Microsoft-Schlüssel über Dump-Dateien
    - Speicherung sensibler Daten im Speicher
    - Fallstudie - KeePass-Passwortleck über Zeichenketten
  - Informationsexposition
    - Offenlegung durch extrahierte Daten und Aggregation
    - Fallstudie - Strava-Datenexposition
  - Sicherheit der Plattform
    - Sicherheit der .NET-Plattform
    - Schutz von .NET-Code und -Anwendungen
    - Unterzeichnung des Codes
  - Denial of Service
    - Überschwemmungen
    - Erschöpfung der Ressourcen
    - Fragen der algorithmischen Komplexität
    - Denial of Service mit regulären Ausdrücken (ReDoS)
    - Labor - ReDoS
    - Labor - Experimentieren mit ReDoS in Copilot
    - Der Umgang mit ReDoS
  - Verwendung anfälliger Komponenten
    - Fallstudie - Der Angriff auf die Lieferkette von Polyfill.io
    - Management von Schwachstellen
    - Übung - Auffinden von Schwachstellen in Komponenten von Drittanbietern
  - Sicherheit von KI-generiertem Code
    - Praktische Angriffe auf Tools zur Codegenerierung
    - Abhängigkeits-Halluzination durch generative KI
    - Fallstudie - Eine Geschichte der Schwächen von GitHub Copilot (bis Mitte 2024)

Tag 3

Kryptographie für Entwickler

- Grundlagen der Kryptographie
- Krypto-APIs in C#
- Elementare Algorithmen
- Hashing
  - Grundlagen des Hashings
  - Hashing in C#
  - Übung - Hashing in C# (Erkundung mit Copilot)
- Erzeugung von Zufallszahlen
  - Pseudo-Zufallszahlengeneratoren (PRNGs)
  - Kryptografisch sichere PRNGs
  - Schwache und starke PRNGs
  - Verwendung von Zufallszahlen in C#
  - Übung - Verwendung von Zufallszahlen in C# (Erkundung mit Copilot)
  - Fallstudie - Equifax-Kontosperrung
- Schutz der Vertraulichkeit
  - Symmetrische Verschlüsselung
    - Blockchiffren

- Betriebsarten
- Betriebsarten und IV - bewährte Verfahren
- Symmetrische Verschlüsselung in C#
- Symmetrische Verschlüsselung in C# mit Streams
- Übung - Symmetrische Verschlüsselung in C# (Erkundung mit Copilot)
- Fallstudie - Padding-Orakel in RCE gegen Citrix ShareFile verwendet

■ Asymmetrische Verschlüsselung
- Der RSA-Algorithmus
- RSA in C#
- Kombination von symmetrischen und asymmetrischen Algorithmen

■ Schlüsselaustausch und Vereinbarung
- Austausch von Schlüsseln
- Diffie-Hellman-Schlüsselvereinbarungsalgorithmus
- Die wichtigsten Fallstricke beim Austausch und bewährte Verfahren

◦ Schutz der Integrität
■ Nachrichten-Authentifizierungs-Code (MAC)
■ HMAC-Berechnung in C#
■ Übung - MAC-Berechnung in C#
■ Digitale Unterschrift
- Digitale Unterschrift mit RSA
- Elliptische Kurven Kryptographie
- ECC-Grundlagen
- Digitale Unterschrift mit ECC
- Digitale Unterschrift in C#
- Übung - Digitale Signatur mit ECDSA in C#

◦ Häufige Sicherheitslücken in Software
■ Code quality
- Codequalität und Sicherheit

■ Umgang mit Daten
- Initialisierung und Bereinigung
- Initialisierungszyklen der Klasse
- Übung - Initialisierungszyklen (Erkundung mit Copilot)

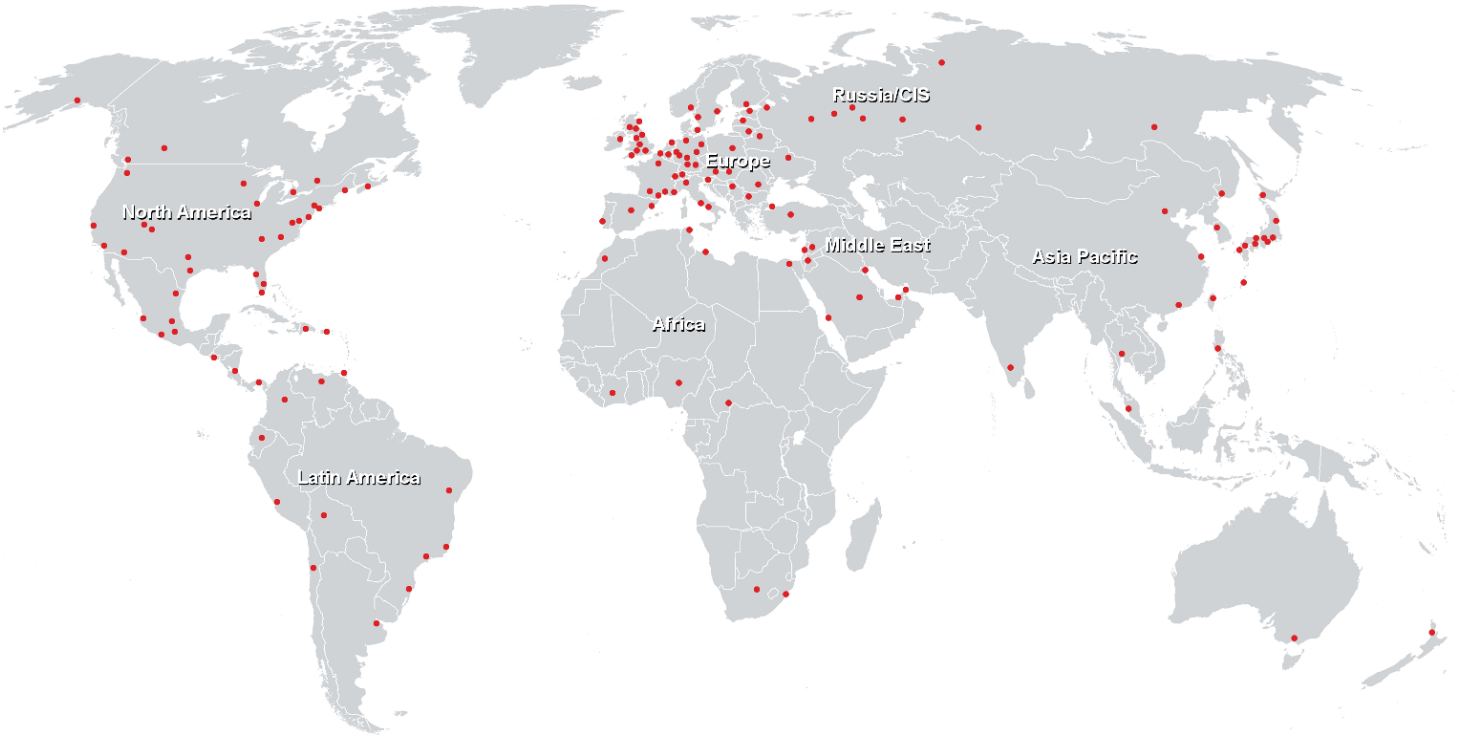
■ Fallstricke der objektorientierten Programmierung
- Vererbung und Overriding
- Veränderlichkeit
- Lab - Veränderbares Objekt (Erkundung mit Copilot)

■ Serialisierung
- Herausforderungen bei Serialisierung und Deserialisierung
- Integrität - Deserialisierung nicht vertrauenswürdiger Datenströme
- Integrität - bewährte Verfahren zur Deserialisierung
- Vorausschauende Deserialisierung
- Eigenschaftsorientiertes Programmieren (POP)
- Erstellen einer POP-Nutzlast
- Übung - Erstellen einer POP-Nutzlast
- Übung - Verwendung der POP-Nutzlast
- Fallstudie - Deserialisierung RCE in Veeam

◦ Einpacken
■ Grundsätze der sicheren Kodierung
■ Grundsätze der robusten Programmierung von Matt Bishop
■ Sichere Gestaltungsprinzipien von Saltzer und Schroeder
■ Und was nun?
■ Quellen zur Softwaresicherheit und weiterführende Literatur
■ .NET- und C#-Ressourcen
■ Verantwortungsvolle KI-Prinzipien in der Softwareentwicklung
■ Generative AI - Ressourcen und zusätzliche Anleitungen

Code responsibly with generative AI in C# (desktop applications) (CRWGAIC)

Weltweite Trainingscenter



Fast Lane Institute for Knowledge Transfer (Switzerland) AG

Husacherstrasse 3
CH-8304 Wallisellen
Tel. +41 44 832 50 80

info@flane.ch, https://www.flane.ch