IT-Sicherheit - Bewusstseins-Training für Software-Entwickler, -Architekten und -Tester
Security by Design

"*Falls Sie glauben, dass Technologie Ihre Sicherheitsprobleme lösen kann, dann verstehen Sie das Problem nicht, oder haben von Technologie keine Ahnung!*"
Bruce Schneier, Experte für Computersicherheit, 2000

Das Sicherheits-Bewusstseins-Training soll Ihr Verständnis für sichere Software-Entwicklung entfachen, erweitern und schärfen. Es soll Sie zu einem verantwortungs-bewussten Software-Entwickler/-Architekten/-Tester machen, der neben der Softwarequalität auch stets die IT-Sicherheit fest im Blick hat. In vier Lektionen wird praxisbezogenes Wissen zum Thema IT-Sicherheit im Entwicklungs- und Testprozess vermittelt und vertieft.
Dazu gehören erprobte Sicherheits-Konzepte, bewährte Sicherheits-Techniken, Software-Entwicklungsstandards und vor allem alltagstaugliche Werkzeuge für den Projektalltag.

Nach dem Sicherheits-Bewusstseins-Training sind Sie in der Lage:

- Eigene Systeme/Projekte auf Schwachstellen zu untersuchen.
- Werkzeuge aus der Werkzeugkiste des Sicherheitsexperten anzuwenden.
- Mögliche Sicherheitslücken im eigenem Quellcode zu erkennen, zu bewerten und zu beseitigen.
- Sichere Software nach Standards der IT-Sicherheit zu entwickeln.
- Die Prinzipien "Security by design" und "Privacy by design" anzu wenden und in eigenen Projekten gewinnbringend umzusetzen.

Dauer
3 Tage

Zielgruppe
Software-Entwickler, -Architekten und -Tester

Voraussetzungen
Programmiererfahrung in einer gängigen Programmier-Sprache wie z. B. Java, C#, C++, etc...

Lektion 1: „Der Schwachstelle auf den Zahn gefühlt“
Zur Einführung in das komplexe Thema ITSicherheit in der Softwareentwicklung werden drei der "bekanntesten" Sicherheitslücken aus den OWASP Top Ten vorgestellt. Innerhalb einer Beispielumgebung werden die Sicherheitslücken ausgenutzt und die Folgen untersucht. Abschließend diskutieren und bewerten wir, welche Fehler in der Implementierung und im Entwicklungsprozess gemacht wurden, und wie sie sich vermeiden lassen.

Am Ende der Lektion wissen Sie:

- Was eine Sicherheitslücke ist.
- Wie und wodurch eine Sicherheitslücke verursacht werden kann.
- Wie eine Sicherheitslücke ausgenutzt werden kann.
- Welche Bedrohung von einer Sicherheitslücke ausgehen kann.

Frank Bahn
training@fb-it-solutions.de

Lektion 2: „Bewusstseinserweiterung“
In dieser Lektion werden die OWASP Organisation, die OWASP Top Ten und die Handlungsempfehlungen für sichere Softwareentwicklung, u. a. die OWASP Secure Coding Cheat Sheets, vorgestellt. Nach einer allgemeinen Einführung widmen wir uns ausführlich der OWASP Top Ten Hitliste. Sie lernen, wie die Fehler aus der OWASP Top Ten entstehen, welche Folgen sie haben und wie sie zu vermeiden sind.
Abschließend werden ausgewählte Handlungsempfehlungen u. a. aus den OWASP Secure Coding Cheat Sheets vorgestellt und diskutiert.

Am Ende der Lektion haben Sie verstanden:

- Was die OWASP Top Ten Hitliste und die Secure Coding Cheat Sheets sind.
- Wie Sie IT-Sicherheit Coding Standards wie z. B. OWASP Secure Coding Cheat Sheets für eigene Projekte gewinnbringend nutzen können.

Lektion 3: „Die Werkzeugkiste“
In dieser Lektion stellen wir die Werkzeugkiste des sicherheitsbewussten Software-Entwicklers/-Architekten/-Testers zusammen. Ziel ist es, den eigenen Entwicklungen mit den Waffen des Gegners auf den Zahn zu fühlen.
Sie lernen die Vorgehensweisen Vulnerability Scan und Penetration Test kennen und mittels der vorgestellten Werkzeuge anzuwenden. Wir diskutieren und bewerten die jeweiligen konzeptionellen Vor- und Nachteile. Abschließend entwickeln wir Szenarien für den sinnvollen Einsatz der Werkzeugkiste im sicherheitsbewussten Entwicklungsprozess (Secure Development LifeCycle).

Nach der Lektion:

- Sind Sie in der Lage, eigene Systeme/Projekte auf vorhandene Sicherheitslücken zu untersuchen.
- Haben Sie die konzeptionellen Unterschiede zwischen Vulnerability Scan und Penetration Test verstanden und können beide Techniken anwenden.
- Sind Sie in der Lage gefundene Sicherheitslücken zu bewerten, zu dokumentieren und im Team fachgerecht zu diskutieren.
- Sind Sie in der Lage, bekannte Sicherheitslücken nach Common Vulnerability Scoring System (CVSS) und Common Vulnerabilities and Exposures (CVE) einzuordnen und zu bewerten.

Lektion 4: „Die Herausforderung“
In letzten Lektion stellen Sie Ihr erworbenes Wissen auf die Probe. Sie werden einer unbekannte Anwendung mit den vorgestellten Werkzeugen zu Leibe rücken und versuchen herauszufinden, ob und welche Sicherheitslücken die Anwendung aufweist. Mögliche Sicherheitslücken werden Sie analysieren, bewerten, dokumentieren und mit anderen Workshop-Teilnehmern diskutieren. Abschließenden erarbeiten und diskutieren Sie mit anderen Workshop-Teilnehmern mögliche Lösungsansätze wie die Sicherheitslücken zu schließen wären.

Nach der Lektion:

- Sind Sie in der Lage eine unbekannte Anwendung auf Sicherheitslücken zu untersuchen.
- Sind Sie in der Lage Reverse Engineering, Vulnerability Scan und Penetration Test durchzuführen.
- Haben Sie verstanden, wie wichtig die vorgestellten Methoden sind, und dass Sie von Projektbeginn an in den sicheren Entwicklungsprozess integriert werden sollten.