COURSE OVERVIEW

Course Name:
(CN-CSC+)
CertNexus Cyber Secure Coder

COURSE DURATION: 3 Days

Gauteng:
3rd Floor, 34 Whitely Road
Melrose Arch
Johannesburg
2196

Gauteng:
192 on Bram
192 Bram Fischer Drive
Ferndale, Randburg
Johannesburg
2160

Cape Town:
3rd Floor, Thomas Pattullo Building
19 Jan Smuts St
Cape Town
8000

Durban:
9 Mountview Close
Broadlands
Mount Edgecombe
Durban
4302

087 941 5764
sales@impactful.co.za
impactful.co.za

INTRODUCTION

The stakes for software security are very high, and yet many development teams deal with software security only after the code has been developed and the software is being prepared for delivery. As with any aspect of software quality, to ensure successful implementation, security and privacy issues should be managed throughout the entire software development lifecycle. This course presents an approach for dealing with security and privacy throughout the entire software development lifecycle. You will learn about vulnerabilities that undermine security, and how to identify and remediate them in your own projects. You will learn general strategies for dealing with security defects and misconfiguration, how to design software to deal with the human element in security, and how to incorporate security into all phases of development.

DELIVERY METHOD

Our courses have flexible delivery options:

- In-person classroom training at the Impactful training facilities
  - Johannesburg, Durban, Cape Town
- Virtual instructor-led training
- Nationally: on-site at the client

IMPACTFUL
Powered by LRMG

TARGET AUDIENCE

This course is designed for software developers, testers, and architects who design and develop software in various programming languages and platforms, including desktop, web, cloud, and mobile, and who want to improve their ability to deliver software that is of high quality, particularly regarding security and privacy. This course is also designed for students who are seeking the CertNexus Cyber Secure Coder (CSC) Exam CSC-210 certification

PREREQUISITES

This course presents secure programming concepts that apply to many different types of software development projects. Although this course uses Python®, HTML, and JavaScript® to demonstrate various programming concepts, you do not need to have experience in these languages to benefit from this course. However, you should have some programming experience, whether it be developing desktop, mobile, web, or cloud applications. Logical Operations provides a variety of courses covering software development that you might use to prepare for this course, such as:

- Python® Programming: Introduction
- Python® Programming: Advanced
- HTML5: Content Authoring with New and Advanced Features
- SQL Querying: Fundamentals (Second Edition)

COURSE OBJECTIVES

In this course, you will employ best practices in software development to develop secure software.
You will:

- Identify the need for security in your software projects.
- Eliminate vulnerabilities within software.
- Use a Security by Design approach to design a secure architecture for your software.
- Implement common protections to protect users and data.
- Apply various testing methods to find and correct security defects in your software.
- Maintain deployed software to ensure ongoing security.

This course includes hands on activities for each topic area. The goal of these activities is to demonstrate concepts utilizing two universal languages Python and Java Script. Developers who use alternate languages will be able apply the principles from the activities to any coding languages.

Hands on exercises are designed to keep the typing of code to a bare minimum. CertNexus provides students with all of the code they need to complete activities. The activities do not require a “deep dive” into code to understand the principles being covered.

COURSE CONTENT

Lesson 1: Identifying the Need for Security in Your Software Projects

- Topic A: Identify Security Requirements and Expectations
- Topic B: Identify Factors That Undermine Software Security
- Topic C: Find Vulnerabilities in Your Software
- Topic D: Gather Intelligence on Vulnerabilities and Exploits

Lesson 2: Handling Vulnerabilities

- Topic A: Handle Vulnerabilities Due to Software Defects and Misconfiguration
- Topic B: Handle Vulnerabilities Due to Human Factors
- Topic C: Handle Vulnerabilities Due to Process Shortcomings

Lesson 3: Designing for Security

- Topic A: Apply General Principles for Secure Design
- Topic B: Design Software to Counter Specific Threats

Lesson 4: Developing Secure Code

- Topic A: Follow Best Practices for Secure Coding
- Topic B: Prevent Platform Vulnerabilities
- Topic C: Prevent Privacy Vulnerabilities

Lesson 5: Implementing Common Protections

- Topic A: Limit Access Using Login and User Roles
- Topic B: Protect Data in Transit and At Rest
- Topic C: Implement Error Handling and Logging
- Topic D: Protect Sensitive Data and Functions
- Topic E: Protect Database Access

Lesson 6: Testing Software Security

- Topic A: Perform Security Testing
- Topic B: Analyse Code to find Security Problems
- Topic C: Use Automated Testing Tools to Find Security Problems

Lesson 7: Maintaining Security in Deployed Software

- Topic A: Monitor and Log Applications to Support Security
- Topic B: Maintain Security after Deployment

Appendix A: Mapping Course Content to Cyber Secure Coder (Exam CSC-210)