

Secure Coding Principles & Practices



KEY DATA

Course Name:

Certified Secure Code Engineer

Language: English

Format:

Instructor-led (Lecture and Lab)

Prerequisite:

- Mandatory – basic knowledge of a high level language such as C, Java, Perl, PHP and C#
- Knowledge of information security principles
- Understanding of networking protocols
- Completion of CPTS & CPTe or equivalent knowledge

Student Materials:

- Student Workbook
- Student CD
- Secure Coding: Principles & Practices Book
- Mile2 Notebook
- Mile2 CD Case
- Mile2 Pen

BENEFITS OF THIS COURSE

This 4-day course delivers a strong and in-depth view into fundamentals of software security and secure coding. Through a mixture of instructor-led lectures and hands on exercises, students will have learn how to identify security flaws early in the development process and how design and code to eliminate these flaws. All examples and lessons are presented in a variety of high level and scripting languages.

COURSE OVERVIEW

Software security is big concern for organizations today. More and more attacks are being directed towards software applications and understanding these attacks and how to design software to protect against these attacks is becoming more of a necessity. This course seeks to provide a foundation for those professionals who are responsible for designing, architecting, coding and testing software solutions through a series of lectures and hands on labs.

UPON COMPLETION

Students will have a sound understanding of common vulnerabilities found in today's software and how to defend against them. Students will gain knowledge in creating secure designs and code and how to apply this knowledge to their daily tasks.

COURSE DETAILS

Modules:

- Module 1 -Software Security Explained
- Module 2 -Setting the Stage (The Attack)
- Module 3 -Risk Management (Developer to Developer)
- Module 4 -Threat Modeling
- Module 5 -The Secure Software Development Life Cycle
- Module 6 -Secure Architecture Design
- Module 7 -Secure Coding
- Module 8 -Cryptography
- Module 9 -Attacking Databases
- Module 10 –Attacking Web Technologies

DETAILED MODULE DESCRIPTION

Module 1: Software Security Explained

In order for students to understand how to write secure software, they must understand what software security is. In module 1, students will gain a deep understanding of key security concepts and issues in with securing software.

1. Definition of Software Security
2. Understanding Software Security
3. Foundation of Security
4. Challenges With Security
5. The Rise of Insecure Software
6. Software Security Methodology
7. The Teams 's Roles and Responsibilities
8. Developer's Role and Responsibilities
9. Common Vulnerabilities
10. Staying Informed

Module 2: Setting the Stage (The Attack)

Protecting your system means knowing your attacker. This module covers the process that attackers follow when targeting and attacking systems. Students get an introduction to the penetration testing process, how to use these techniques to test their software and a variety of tools used during the process.

1. Learning Attack Methods
2. Developer's Point of View
3. Know the Attacker
4. Methodology of Penetration Testing
5. Reconnaissance
6. Methods of Obtaining Information
7. Passive vs. Active Reconnaissance
8. Footprinting Defined
9. Footprinting Countermeasures
10. Enumeration
11. Introduction to Port Scanning
12. Case Study and Lab

Module 3: Risk Management

If you know what the risks are, how do you track them? Risk management is a key concept needed to identify and track risks throughout the development process to ensure that risks are captured and mitigated appropriately.

1. Important Terms
2. The Importance of Risk Management
3. When Should it Start
4. The Risk Management Process
 - a. Know The Business
 - b. Identify Risk
 - c. Classify Risk
 - d. Develop Mitigation Plan
 - e. Implement
 - f. Validate
5. Risk Analysis
6. Report Your findings
7. Case Study and Lab

Module 4: Threat Modeling

How do you identify weak points in your software? This module covers a framework that can be used to identify various attack vectors in software designs and how to use this information to secure your system.

1. Threat Modeling Defined
2. The Threat Modeling Process
 - a. Identify Security Objectives
 - b. Application Review
 - c. Application Decomposition
 - d. Identify Threats
 - e. Identify Vulnerabilities
 - f. Determine Countermeasures
3. Threat Methodologies
 - a. Stride
 - b. Dread
4. How to Respond to Threats
5. Mitigating Threats
6. Case Study and Lab

Module 5: Secure SDLC

To consistently write secure software, the process of building software must incorporate security. This module covers the process of designing and building secure software and the problems with traditional methods.

1. Secure SDLC Overview
2. A Secure Process
3. Manager's Point of view
4. Developer's Point of View
5. Why Change?
6. Consumer Expectations
7. Business Responsibility
8. Phases of The Development Lifecycle
 - a. Project Initiation/Concept
 - b. Gathering Requirements
 - c. Analysis and Architecture Design
 - d. Development
 - e. Unit Testing
 - f. Quality Assurance
 - g. Implementation
9. Case Study and Lab

Module 6: Secure Architecture and Design

Secure code starts with a secure design. Module 6 focuses on the process of designing software with security built in and the benefits of this method versus the traditional method of adding security during or after the code has been written.

1. Design It Secure
2. Design Considerations
3. The SD3 Framework
 - a. Secure By Design
 - b. Secure By Default
 - c. Secure in Deployment
4. Understanding the Environment
5. Technical Issue
6. Security in Layers
7. Attacks
 - a. Man-in-the Middle
 - b. Session Hijacking
8. Buy vs. Build
9. Filters
10. Case Study and Lab

Module 7: Writing Secure Code

Module 7 covers the key concepts students need to write secure code. Students will learn how to build validation and logging frameworks to prevent the most common attacks, log potential intrusion attempts and audit user and system activity. Students will also learn how to perform code reviews.

1. Data Validation
2. Defending the Attack
3. Error and Exception Handling
4. Logging and Auditing
5. Authentication
6. Web Authentication Methods
 - a. Basic and Digest Authentication
 - b. Form Based Authentication
 - c. Certificate Based Authentication
 - d. Strong Authentication
7. Authorization
8. Security Code Reviews
 - a. Know the Vulnerabilities
 - b. Know The Business Risks
 - c. When to Conduct the Review
 - d. Who Should Be Involved
 - e. What to Look For
 - f. Fixing the Issues
 - g. Automated Tools
9. Case Study and Lab

Module 8: Cryptography

Developers must understand the different encryption options available and when to choose one method over another. This module will provide students with a thorough understanding of encryptions methods, what criteria should be used when deciding to use an encryption method and the pros and cons of each.

1. Cryptography Introduction
2. Encryption
3. Implementation.
4. Symmetric Encryption
5. Symmetric Algorithms
6. Asymmetric Encryption
7. Key Exchange
8. Hashing
9. Hash Collisions
10. Common Hash Algorithms
11. Hybrid Encryption
12. Digital Signatures

13. SSL Hybrid Encryption
14. IPSEC
15. Transport Layer Security – SSH
16. PKI ~ Public Key Infrastructure Models
17. PKI-Enabled Applications
18. Quantum Cryptography
19. Attack Vectors
20. Case Study & Lab

Module 9: Attacking the Database

The reason why most software applications exist is to gather, process and store some type of data. Module 10 gives the student insight into how databases are attacked. Students get a low level view of specific database attacks and the tools and methods during these attacks.

1. Overview of Database Servers
2. Types of Databases
3. Tables, Records, Attributes, Domains
4. Data Normalization, SQL , Object-Oriented Database Management
5. Relational Database Systems
6. Vulnerabilities and Common Attacks
 - a. SQL Injection
 - b. Why SQL Injection
 - c. SQL Connection Properties
 - d. SQL Injection: Enumeration
7. Extended Stored Procedures
8. Shutting Down SQL Server
9. Direct Attacks
10. Attacking Database Servers
11. Obtaining Sensitive Information
12. Hardening Databases

Module 10: Attacking the Web

With so many software applications moving to the Internet, web applications have become one of the most attacked surfaces in the software space. Module 9 gives students the opportunity learn about the different attacks used against web technologies and how to protect their systems.

1. Common Security Threats
2. Sans Seven Management Errors
3. Progression of The Professional Hacker
4. Anatomy of A web application attack
5. Web Attack Techniques
6. Components of a generic web application system
7. URL mappings to the web application system
8. Pen Testing tools and methodologies for Web Servers assessment

- 9. Understanding Web Application Security
- 10. Common Web Application Security Vulnerabilities
- 11. Authentication And Session Management
- 12. Password Guessing/Cracking Tools
- 13. Case Study and Lab