COURSE CATALOG

TEAM Professor Application Security

SECURITY INNOVATION

RSA

Contact Us: RSAUniversity@rsa.com

Table of Contents

TEAM Professor Application Security

Fundamentals

AWA 101

Fundamentals of Application Security

Duration: 60 minutes
Languages: English, Japanese, Chinese, German, Spanish (LA), French (CA)

This course introduces the fundamentals of application security. It discusses the main drivers for application security, fundamental concepts of application security risk management, the anatomy of an application attack, some common attacks, the concept of input validation as a primary risk mitigation technique, and key security principles and best practices for developing secure applications.

AWA 102

Software Security Awareness

Duration: 60 minutes
Languages: English, Japanese

Similar to functionality, performance, and reliability, security is another crucial component of an application's quality. Recognizing the risk that software vulnerabilities represent, understanding their root causes, and addressing these issues early in the software development lifecycle are essential for being able to help your organization build secure software. By the end of this course, students will be familiar with the main characteristics of a secure software development lifecycle and the activities that an organization should perform to develop secure software. Additionally, students will recognize the need to address software security in their everyday work.

AWA 110

Fundamentals of Security Awareness for Mobile Devices

Duration: 30 minutes
Languages: English, Japanese

This course discusses the security risks of using mobile devices and introduces the five fundamentals of secure mobile computing. Mobile devices are designed to operate outside the confines of enterprise networks, and it is incumbent on the organization and the individual to understand and implement security best practices that mitigate risks to privacy, confidential data, reputation, and other assets. Course coverage includes the risks of using Wi-Fi connections, and discusses the security of tablets, notebooks, smartphones, and external storage devices.

AWA 111

Fundamentals of Security Awareness for Social Media

Duration: 30 minutes
Languages: English, Japanese

This course introduces you to social media security and why it's important to both employees and employers. It provides a general overview of how to stay safe and secure. The course addresses general privacy and security best practices that can be applied across all social media sites. The course also covers specific security issues and security best practices for each popular social network: Facebook, Twitter, Google Plus, and LinkedIn. Finally, the course discusses privacy and security issues, and best practices for managing company pages, and addresses employer policies for social media usage by employees.

SECURITY INNOVATION

Secure Coding

COD 101

Fundamentals of Secure Development

Duration: 80 minutes
Languages: English, Japanese, Chinese, German

This course introduces you to the need for secure software development, as well as the models, standards, and guidelines that you can use to understand security issues and improve the security posture of your applications. It also describes key application security principles and secure coding principles, and explains how to integrate secure development practices into the software development lifecycle. Note: The training should take approximately 80 minutes to complete. This course is optimally viewed at a screen resolution of 1024x768.

COD 110

Fundamentals of Secure Mobile Development

Duration: 120 minutes
Languages: English, Japanese, Chinese

This course introduces developers to the common risks associated with Mobile applications including client side injection, sensitive data handling, network transition, application patching, web based attacks, phishing, third-party code, location security and privacy and denial of service. The student is then given an overview of the Mobile application development best practices to reduce these risks including input validation, output encoding, least privilege, code signing, data protection at rest and in transit, avoiding client side validation, and using platform security capabilities as they apply in mobile environments. Included is a discussion of threat modeling mobile applications. With knowledge checks throughout, the student who completes this course will have an understanding of mobile environment threats and risks, and the programming principals to use to address them.

COD 141

Fundamentals of Secure Database Development

Duration: 110 minutes
Languages: English, Japanese

In practice, the database represents the goal of many attackers, as this is where the information of value is maintained. However, functional requirements and security testing often focus on the interaction between a software user and the application, while the handling of data is assumed to be secure. This course is platform and technology agnostic, and will provide software architects and developers with an understanding of database development best practices.

COD 152

Fundamentals of Secure Cloud Development

Duration: 90 minutes
Languages: English, Japanese

This course introduces developers to the common risks associated with Cloud applications, including the security features of the different series models (IaaS, PaaS, and Saas), how to identify and mitigate the most common vulnerabilities, the unique security challenges of “Big Data”, and how to apply the Microsoft SDL to cloud applications. Threat coverage includes unauthorized account access, insecure APIs, shared technology, data leakage, and account hijacking, as well the importance of complying with regulatory requirements. With knowledge checks throughout, the student who completes this course will have an understanding of cloud computing threats and risks, and the programming principals to use to address them.

COD 153

Fundamentals of Secure AJAX Code

Duration: 35 minutes
Languages: English, Japanese, Chinese

This course introduces security issues and challenges specific to AJAX applications. It provides an overview of AJAX technology, and presents common AJAX application vulnerabilities and attack vectors. Upon completion of this class, participants will be able to identify the differences between regular and AJAX applications, common AJAX vulnerabilities that attackers tend to exploit, and major threats to AJAX applications from cross-site scripting, cross-site request forgery, and injection attacks.

COD 160

Fundamentals of Secure Embedded Software Development

Duration: 90 minutes
Languages: English, Japanese, Chinese

In this course, you will learn about security issues inherent to embedded device architecture. You will also learn about techniques to identify system security and performance requirements, develop appropriate security architecture, select the correct mitigations, and develop policies that can ensure the secure operation of your system.

COD 190

Fundamentals of Secure Mobile Development for Embedded Systems

Duration: 30 minutes
Languages: English

This course provides additional training on Secure Mobile Development of particular importance to embedded software engineers. It includes mapping of content to specific compliance and regulatory requirements, links to key reference resources that support the topics covered in the module, and a "Knowledge Check" quiz that assesses mastery of key concepts.

COD 211

Creating Secure Code – Java Foundations

Duration: 30 minutes
Languages: English

This course provides additional training on Secure Mobile Development of particular importance to embedded software engineers. It includes mapping of content to specific compliance and regulatory requirements, links to key reference resources that support the topics covered in the module, and a "Knowledge Check" quiz that assesses mastery of key concepts.

COD 212

Creating Secure Code – C/C++ Foundations

Duration: 120 minutes
Languages: English, Japanese

This course presents best practices and techniques for secure application development in C/C++. It discusses basic application security principles, input validation in C/C++, common C/C++ application security vulnerabilities and mitigations, protecting data in C/C++, and conducting security code reviews.

COD 213

Creating Secure Code – Windows Foundations

Duration: 120 minutes
Languages: English, Japanese

This course provides students with knowledge and skills needed to understand Windows 7 security features and build applications that leverage Windows 7's built-in security mechanisms.

COD 215

Creating Secure Code - .NET Framework Foundations

Duration: 120 minutes
Languages: English, Japanese

This course describes .NET 4 security features, including concepts such as Code Access Security (CAS) and .NET cryptographic technologies. In addition, this course will introduce you to security changes in .NET 4 including level 2 security transparency, the new sandboxing and permission model, introduction of conditional APTCA, and changes to evidence objects and collections. This course provides secure coding best practices that will enable students to build more secure applications in .NET 4.

COD 217

Creating Secure Code – iPhone Foundations

Duration: 60 minutes
Languages: English, Japanese

This course teaches iPhone application programmers the principles necessary to build highly secure iPhone applications. This class discusses key iPhone application risks and vulnerabilities and the techniques you can use to defend against them. The student will learn secure programming principles for iPhone applications including how to use the developer security tools provided by iOS as well as secure development best practices for defending against Web based attacks, SQL injection, session hijacking, data theft and jailbreaking.

COD 218

Creating Secure Code – Android Foundations

Duration: 90 minutes
Languages: English, Japanese

This course teaches the principals necessary to develop secure Android applications. It discusses the Android security model, secure programming principals, security features provided by the Android OS, and key Android attack vectors and mitigation techniques. Included are Android secure development best practices for defending against SOL injection, Malware, IPC vulnerabilities, and data theft.

COD 219

Creating Secure Code – SAP ABAP Foundations

Duration: 90 minutes
Languages: English

This course presents best practices and techniques for secure SAP application development using Java and ABAP. It discusses basic application security principles, input validation in SAP applications, common application security vulnerabilities and mitigations, protecting data using encryption, and conducting security code analysis and code reviews.

COD 221

Web Vulnerabilities – Threats & Mitigations

Duration: 60 minutes
Languages: English, Japanese

This course provides all the information needed to understand, avoid, and mitigate the risks posed by Web vulnerabilities. Students are first provided with a detailed background on the most common and recent attacks against Web-based applications, such as cross-site scripting attacks and cross-site request forgery attacks. The course then delves into practical recommendations on how to avoid and/or mitigate Web vulnerabilities. Real-world examples are provided throughout the course to help students understand and defend against Web vulnerabilities.

COD 222

PCI DSS v3.2 Best Practices for Developers

Duration: 60 minutes
Languages: English, Japanese

The Payment Card Industry Data Security Standard (PCI-DSS) Version 3.2 provides minimum requirements for addressing the security of software systems handling credit card information. Addressing the requirements during the design and build stages of the development lifecycle improves application security and simplifies compliance. This course will provide software developers with an in-depth understanding of application security issues within the PCI-DSS Version 3.2 and best practices for addressing each requirement.

COD 231

Introduction to Cross-Site Scripting – JSP

Duration: 20 minutes
Languages: English, Japanese, Chinese

In this course, students will learn to understand the mechanisms behind cross-site scripting vulnerabilities, describe cross-site scripting vulnerabilities and their consequences, and apply secure coding best practices to prevent cross-site scripting vulnerabilities.

COD 232

Introduction to Cross-Site Scripting – ASP.NET

Duration: 20 minutes
Languages: English, Japanese

In this course, students will learn about cross-site scripting vulnerabilities and their consequences, the mechanisms behind cross-site scripting attacks, and secure coding best practices to help prevent cross-site scripting vulnerabilities.

COD 241

Creating Secure Code – Oracle Foundations

Duration: 120 minutes
Languages: English, Japanese

This course provides the student with an understanding of the scope and requirements of database security as well as the risks presented by insecure database applications. It then teaches the best practices for secure database application development including privileges and access control, query construction, communication and storage, audit and resource usage. The course concludes with a discussion of common database attacks and how to prevent them, including SQL Injection, Information Disclosure and Privilege Escalation. This class teaches these principals using Oracle specific code and examples. After taking this course, the student will be able to understand the risks to database applications; apply security best practices when developing database applications; understand common database attacks; code applications with countermeasures to common database attacks.

COD 242

Creating Secure Code – SQL Server Foundations

Duration: 90 minutes
Languages: English, Japanese

This course provides the student with an understanding of the scope and requirement of database security as well as the risks presented by unsecure database applications. It then teaches the best practices for secure database application development including privileges and access control, query construction, communication and storage, audit and resource usage. The course concludes with a discussion of common database attacks and how to prevent them, including SQL Injection, Information Disclosure and Privilege Escalation. This class teaches these principals using SQL Server specific code and examples. After taking this course, the student will be able to understand the risks to database applications; apply security best practices when developing database applications; understand common database attacks; code applications with countermeasures to common database attacks.

COD 251

Creating Secure AJAX Code – ASP.NET Foundations

Duration: 90 minutes
Languages: English, Japanese

This course introduces secure ASP.NET coding principles for AJAX applications. It provides an overview of best practices to mitigate common vulnerabilities and protect against common attack vectors. Upon completion of this class, participants will be able to identify the threats to AJAX applications from cross-site scripting, cross-site request forgery, and injection attacks, and ways to implement countermeasures against these attacks by protecting client resources, validating input, protecting web services requests, preventing request forgeries, and securing data access.

COD 252

Creating Secure AJAX Code – Java Foundations

Duration: 35 minutes
Languages: English, Japanese, Chinese

This course introduces secure Java coding principles for AJAX applications. It provides an overview of best practices to mitigate common vulnerabilities and protect against common attack vectors. Upon completion of this class, participants will be able to identify the most common threats to AJAX applications from cross-site scripting, cross-site request forgery, and injection attacks, and ways to implement countermeasures against attacks by protecting client resources, validating input, restricting access to Ajax services, and preventing request forgeries.

COD 253

Creating Secure Cloud Code – AWS Foundations

Duration: 60 minutes
Languages: English, Japanese

This course examines the security vulnerabilities, threats, and mitigations for AWS cloud computing services. It includes coverage of Elastic Compute Cloud (EC2), Virtual Private Cloud (VPC), and four additional core AWS services: Identity and Access Management (IAM), DynamoDB Flat Database Service, Relational Database Service (RDS), and Simple Storage Service (S3). This course also discusses ancillary AWS Services. After completing this course, you will be able to identify the most common security threats to cloud development and best practices to protect against these threats. You will also be able to identify AWS security features and ways to integrate them into your AWS resources.

COD 254

Creating Secure Cloud Code – Azure Foundations

Duration: 90 minutes
Languages: English, Japanese

This course examines the security vulnerabilities, threats, and mitigations for Azure cloud computing services. After completing this course, you will be able to identify the most common security threats to cloud based applications and best practices to protect against these threats. Learners will also be able to identify key Azure security platforms and services that you can use to improve the security of your applications.

COD 255

Creating Secure Code – Web API Foundations

Duration: 120 minutes
Languages: English

This course introduces the fundamentals of secure web services development. It describes common web services threats that might put your application at risk, and reviews best practices that you should incorporate to mitigate the risks from web services attacks. After completing this course, you will be able to describe various web services threats, explain the cause and impact of web services attacks, and implement secure development best practices to help protect web services.

COD 256

Creating Secure Code – Ruby on Rail Foundations

Duration: 90 minutes
Languages: English

In this course, you will learn about best practices and techniques for secure application development with Ruby on Rails. After completing this course, you will be able to identify and mitigate injection vulnerabilities, such as SQL injection and cross-site scripting, build strong session management into your Rails applications, and prevent other common vulnerabilities, such as cross-site request forgery and direct object access.

COD 257

Creating Secure Python Web Applications

Duration: 45 minutes
Languages: English

In this course, you will learn about best practices and techniques for secure application development with Python. After completing the course, learners will be able to understand various types of injection vulnerabilities, including SQL injection and cross-site scripting. You will also be able to understand how to build strong session management into your Python web applications and how to prevent common vulnerabilities, such as cross-site request forgery, direct object access, and others. Finally, you will be able to recognize file system threats to web applications, including vulnerabilities with path traversal, temporary files, and insecure client redirects.

COD 292

Creating Secure Code – C/C++ Foundations for Embedded Systems

Duration: 30 minutes
Languages: English

This course provides additional training on C/C++ Foundations of particular importance to embedded software engineers. The module contains the following features: Mapping of content to specific compliance and regulatory requirements Links to key reference resources that support the topics covered in the module “Knowledge Check” quiz that assesses mastery of key concepts.

COD 311

Creating Secure ASP.NET Code

Duration: 240 minutes
Languages: Japanese

In this course, you will learn about ASP.NET MVC and Web API code security issues that affect MVC and Web API applications. You'll learn methods to protect your application from attacks against MVC's model-binding behavior, as well as methods to protect your application from cross-site scripting, cross-site request forgery, and malicious URL redirects. You will also study the Web API pipeline and how to implement authentication and authorization in Web API applications.

COD 312

Creating Secure C/C++ Code

Duration: 120 minutes
Languages: English, Japanese

In this course, you will learn techniques for securing your C/C++ applications. You will learn about secure memory management in C/C++, protecting and authenticating sensitive data with symmetric and public key cryptography, and secure communications with TLS.

COD 313

Creating Secure Java Code

Duration: 35 minutes
Languages: English, Japanese, Chinese

This course examines Java-specific security topics, including the Java security model, the Java authentication and authorization service (JAAS), and cryptography and key management. After completing this course, you will be able to identify and use the components of the Java security model. You will also be able to identify how to use JAAS to control user authentication and authorization in your Java application. In addition, you will be able to manage cryptographic key pairs and certificates in Java, and implement cryptography to sign and verify Java jar files.

COD 314

Creating Secure C# Code

Duration: 150 minutes
Languages: English, Japanese

This course describes methods to produce secure C# applications. It presents common security vulnerabilities that can be mitigated by proper input validation, other common security vulnerabilities and their mitigations, secure error handling and logging, and secure communication. The course also discusses unique features of C# and the .NET Framework that help protect against security vulnerabilities.

COD 315

Creating Secure PHP Code

Duration: 120 minutes
Languages: English, Japanese

This course teaches PHP programmers the security principals they need to know to build secure PHP applications. This class teaches programming principals for security in PHP such as proper session management, error handling, authentication, authorization, data storage, use of encryption and defensive programming as well as avoiding and mitigating vulnerabilities such as SQL Injections, Cross-Site Scripting (XSS), File Inclusion, Command Injection, Cross Site Request Forgery (CSRF) and Null Byte attacks. With interactive knowledge checks in each of the modules, after completing the course, the student will be able to program securely and defensively in PHP.

COD 317

Creating Secure iPhone Code in Objective-C

Duration: 90 minutes
Languages: English, Japanese, Chinese

This course examines in depth the development of secure iOS applications for Apple's iPad and iPhone devices. It provides an overview of common iOS application vulnerabilities and presents secure coding best-practices using Xcode with Objective-C. Upon completion of this class, participants will be able to identify and mitigate malicious user input, risks to data while backgrounding, threats to privacy and confidentiality, sensitive data exposure, insufficient transport layer protection and custom URL scheme abuses.

COD 318

Creating Secure Android Code in Java

Duration: 90 minutes
Languages: English, Japanese, Chinese

This course examines in depth the development of secure Java code for Android OS devices. It provides an overview of common Android application vulnerabilities and presents secure coding best-practices using Java and the Android SDK. Upon completion of this class, participants will be able to identify and mitigate weak authentication attacks, code injections, malicious user input, risks to stored data and data in transit, threats to privacy and confidentiality, insufficient transport layer protection and custom URL scheme abuses.

COD 351

Creating Secure HTML5 Code

Duration: 80 minutes
Languages: English, Japanese

This course examines in depth the development of secure HTML5 code. It provides an overview of common HTML5 application vulnerabilities and threats, and presents secure coding best-practices. Upon completion of this class, participants will be able to identify ways in which the expanded attack surface introduced with HTML 5 might impact your web applications. Participants will also be able to identify new security features available with HTML5, as well as countermeasures and best practices to mitigate the application's exposure to attack.

COD 352

Creating Secure jQuery Code

Duration: 90 minutes
Languages: English, Japanese, Chinese

In this course, you will learn about common client-side vulnerabilities and threats to jQuery applications, and techniques for mitigating these vulnerabilities and threats. You will also learn about how to implement new HTML5 security features to secure JQuery applications, and best practices to secure local storage and implement transport layer security. After completing this course, you will be able to describe the threats that can impact your jQuery code and describe the countermeasures to address these threats.

COD 392

Creating Secure C/C++ Code for Embedded Systems

Duration: 30 minutes
Languages: English

This course module is a supplement to the Security Innovation course “Creating Secure C/C++ Code”. It provides additional coverage on security topics that may be of particular importance to embedded software engineers. It includes mapping of content to specific compliance and regulatory requirements, links to key reference resources that support the topics covered in the module, and a “Knowledge Check” quiz that assesses mastery of key concepts.

COD 411

Integer Overflows – Attacks & Countermeasures

Duration: 60 minutes
Languages: English, Japanese

An integer overflow is a programming error that can severely impact a computer system’s security. Due to the subtlety of this bug, integer overflows are often overlooked during development. This course covers the security concepts, testing techniques, and best practices that will enable students to develop robust applications that are secure against integer overflow vulnerabilities.

COD 412

Buffer Overflows – Attacks & Countermeasures

Duration: 120 minutes
Languages: English, Japanese

This course provides all the required information to understand, avoid and mitigate the risks posed by buffer overflows. The students are first provided with a detailed background on the mechanisms of exploit of stack-based and heap-based buffer overflows. The course then delves into the protections provided by the Microsoft compiler and the Windows operating system, such as the /GS flag and Address Space Layout Randomization (ASLR), followed by practical advice on how to avoid buffer overflows during the design, development, and verification phases of the software development life cycle. Practical examples are provided throughout the course to help students understand and defend against buffer overflows.

Secure Design

DES 101

Fundamentals of Secure Architecture

Duration: 60 minutes
Languages: English, Japanese, Chinese, German

In the past, software applications were created with little thought to the importance of security. In recent times, businesses have become more rigorous about how they buy software. When looking at applications and solutions, companies don't just look at features, functionality, and ease of use. They focus on the total cost of ownership (TCO) of what they purchase. Security is a large and visible part of the TCO equation. In this course, students will examine the state of the industry from a security perspective. They will then look at some of the biggest security disasters in software design and what lessons can be learned from them. Finally, participants will understand and use confidentiality, integrity, and availability as the three main tenets of information security. Upon completion of this course, participants will understand the state of the software industry with respect to security by learning from past software security errors and will avoid repeating those mistakes, and they will understand and use confidentiality, integrity, and availability (CIA) as the three main tenets of information security.

DES 201

Fundamentals of Cryptography

Duration: 120 minutes
Languages: English

In this course, you will learn basic concepts of cryptography and common ways that it is applied, from the perspective of application development. You will learn the importance of randomness; the roles of encoding, encryption, and hashing; the concepts of symmetric and asymmetric encryption; the purpose of cryptographic keys; and the roles of message authentication codes (MACs) and digital signatures. In addition, you'll be introduced to key management, digital certificates, and the public key infrastructure (PKI). Most importantly, you'll understand that cryptography is extremely complex, and requires strong expertise to be properly implemented and validated.

DES 212

Architecture Risk Analysis and Remediation

Duration: 60 minutes
Languages: English, Japanese, Chinese

This course defines concepts, methods, and techniques for analyzing the architecture and design of a software system for security flaws. Special attention is given to analysis of security issues in existing applications; however, the principles and techniques are applicable to systems under development. Techniques include accurately capturing application architecture, threat modeling with attack trees, attack pattern analysis, and enumeration of trust boundaries.

DES 213

Introduction to Security Tools & Technologies

Duration: 120 minutes
Languages: English, Japanese

Security tools allow organizations to systematically test applications for coding mistakes that could result in vulnerabilities. Many organizations purchase security tools such as web application scanners, source code static analysis and penetration testing software; few organizations understand how to effectively select and leverage tools for their needs. This course will review the available types of tools, the relative strengths and weaknesses of each in identifying different classes of security flaws and how to best interpret, prioritize, and act on the output of the tools. This course will provide testing personnel with strategies for selecting and deploying tools, and understand which might best be utilized by internal or external resources.

DES 221

OWASP Top 10 – Threats & Mitigations

Duration: 120 minutes
Languages: English, Japanese, Chinese

This course examines in depth the vulnerabilities, threats, and mitigations described in the OWASP Top 10 2013. Upon completion of this class, participants will be able to identify and mitigate the greatest threats that web application developers face, including: Injection, Broken Authentication and Session Management, Cross-Site Scripting (XSS), Insecure Direct Object References, Security Misconfiguration, Sensitive Data Exposure, Missing Function Level Access Control, Cross-Site Request Forgery (CSRF), Using Components with Known Vulnerabilities, and Unvalidated Redirects and Forwards.

DES 292

Architecture Risk Analysis & Remediation for Embedded Systems

Duration: 30 minutes
Languages: English

This course module provides additional Architecture Risk Analysis and Remediation training of particular importance to embedded software engineers. It includes mapping of content to specific compliance and regulatory requirements, links to key reference resources that support the topics covered in the module, and a “Knowledge Check” quiz that assesses mastery of key concepts.

DES 311

Creating Secure Application Architecture

Duration: 120 minutes
Languages: English, Japanese, Chinese

This course covers a set of key security principles that students can use to improve the security of application architecture and design. Principles of this course include applying defense to harden applications and make them more difficult for intruders to breach, reducing the amount of damage an attacker can accomplish, compartmentalizing to reduce the impact of exploits, using centralized input and data validation to protect applications from malicious input, and reducing the risk in error code paths.

DES 352

Creating Secure Over the Air (OTA) Automotive System Updates

Duration: 90 minutes
Languages: English

In this course, you will learn about the secure design considerations for over-the-air (OTA) updates for automotive systems. After completing this course, you will be able to identify the benefits and risks of OTA automotive system updates, understand the importance of public key cryptography to the security of these updates, and identify secure design considerations for development, delivery, and installation of OTA automotive system updates.

DES 391

Creating Secure Application Architecture for Embedded Systems

Duration: 30 minutes

Languages: English

This course module provides additional training on Creating Secure Application Architecture of particular importance to embedded software engineers. It includes mapping of content to specific compliance and regulatory requirements, links to key reference resources that support the topics covered in the module, and a “Knowledge Check” quiz that assesses mastery of key concepts.

SECURITY INNOVATION

Security Engineering

ENG 101

Microsoft SDL for Managers

Duration: 60 minutes
Languages: English, Japanese, Chinese, German

This course introduces students to the Microsoft SDL, an industry leading software-security assurance process, developed by Microsoft to build trustworthy software products. The goal of this course is to help students understand and identify the Security Development Life Cycle (SDL) requirements for building and deploying secure software applications. The course demonstrates the benefits teams gain by following the SDL, and it provides managers with information regarding their role and responsibilities in ensuring the team follows the SDL. Additionally, this course describes common problems that can delay or stop product shipping.

ENG 102

Introduction to the Microsoft SDL

Duration: 60 minutes
Languages: English, Japanese

This course introduces the Security Development Lifecycle (SDL), a key security engineering process that was spawned from Microsoft's Trustworthy Computing Initiative. Students will learn how to design and implement products that meet an organization's security needs. Upon completion of this course, the participant will be able to identify the benefits of the Security Development Lifecycle, recognize the importance of the Final Security Review, follow the necessary steps to meet SDL requirements, and identify the appropriate tools required by the SDL.

ENG 201

SDLC Gap Analysis and Remediation Techniques

Duration: 45 minutes
Languages: English, Japanese, Chinese, German

Whether an organization is implementing its first Security Development Lifecycle program or working to optimize its SDLC, periodic review of the SDLC to identify areas for improvement is a recommended best practice. This course reviews key Security Engineering activities and instructs students on identifying measurable goals and appropriate standards, assessing existing development processes, building an activity matrix, and creating a remediation roadmap. This course provides an understanding of the goals, processes, and best practices for auditing software security processes within the context of the Microsoft Security Development Life Cycle.

ENG 211

How to Create Application Security Design Requirements

Duration: 60 minutes
Languages: English, Japanese, Chinese, German

Security is an important component of an application's quality. To preserve the confidentiality, integrity, and availability of application data, software applications must be engineered with security in mind beginning with the design phase. Without defined security requirements, design choices will be made without security guidance and security testing cannot be effective. This course provides technical and non-technical personnel with the tools to understand, create and articulate security requirements as part of a software requirement documents. In this course, students will learn to apply the application security maturity (ASM) model to the development process, understand the security-engineering process, and describe the key security-engineering activities to integrate security in the development life cycle. Students will also be able to determine software security objectives, apply security design guidelines, and create threat models that identify threats, attacks, vulnerabilities, and countermeasures, in addition to learning to conduct security architecture and design reviews that help identify potential security problems, and minimize the application's attack surface.

ENG 301

How to Create an Application Security Threat Model

Duration: 90 minutes
Languages: English, Japanese, Chinese

Building secure software begins with creating a threat model to understand the potential threats to an application. The treat modeling process starts by asking what an attacker's goals might be, what information would be valuable to an attacker, and how would an attacker go about gaining access to that information? In this course, students will learn to identify the goals of threat modeling and the corresponding Software Development Lifecycle (SDLC) requirements, identify the roles and responsibilities involved in the threat modeling process, recognize when and what to threat model, and identify the tools that help with threat modeling. Students will learn to use the threat modeling process to accurately identify, mitigate, and validate threats.

ENG 311

Attack Surface Analysis & Reduction

Duration: 60 minutes
Languages: English, Japanese, Chinese

Attack surface analysis and reduction is an exercise in risk reduction. The attack surface of an application represents the number of entry points exposed to a potential attacker of the software. The larger the attack surface, the larger the set of methods that can be used by an adversary to attack. The smaller the attack surface, the smaller the chance of an attacker finding a vulnerability and the lower the risk of a high impact exploit in the system. This course provides an understanding of the goals and methodologies of attackers, identification of attack vectors, and how to minimize the attack surface of an application. In this course, students will learn to define the attack surface of an application, and how to reduce the risk to an application by minimizing the application's attack surface.

ENG 312

How to Perform a Security Code Review

Duration: 60 minutes
Languages: English, Japanese

Application developers may use a variety of tools to identify flaws in their software. Many of these tools, however, cannot be deployed until late in the development lifecycle; dynamic analysis tools require a staging site and sample data, and some static analysis tools require a compiled build. Manual code reviews, in contrast, can begin at any time and require no specialized tools - only secure coding knowledge. Manual code reviews can also be laborious if every line of source code is reviewed. This course provides students with guidance on how to best organize code reviews, prioritize those code segments that will be reviewed, best practices for reviewing source code and maximize security resources.

ENG 352

How to Create an Automotive Systems Threat Model

Duration: 90 minutes
Languages: English

In this course, you will learn how to integrate threat modeling into your secure automotive software development lifecycle. The course provides step-by-step instructions for performing threat modeling, and its recommendations are aligned with the NHTSA’s proposed “Characterization of Potential Security Threats in Modern Automobiles”.

ENG 391

How to Create an Application Security Threat Model for Embedded Systems

Duration: 30 minutes
Languages: English

This course module provides additional training on How to Create an Application Security Threat Model of particular importance to embedded software engineers. It includes mapping of content to specific compliance and regulatory requirements, links to key reference resources that support the topics covered in the module, and a “Knowledge Check” quiz that assesses mastery of key concepts.

ENG 392

Attack Surface Analysis and Reduction for Embedded Systems

Duration: 30 minutes
Languages: English

This course module provides additional training on Attack Surface Analysis and Reduction of particular importance to embedded software engineers. It includes mapping of content to specific compliance and regulatory requirements, links to key reference resources that support the topics covered in the module, and a “Knowledge Check” quiz that assesses mastery of key concepts.

ENG 393

How to Perform a Security Code Review for Embedded Systems

Duration: 30 minutes
Languages: English

This course module provides additional training on Performing Security Code Reviews of particular importance to embedded software engineers. The module contains the following features: Mapping of content to specific compliance and regulatory requirements Links to key reference resources that support the topics covered in the module “Knowledge Check” quiz that assesses mastery of key concepts.

Secure Testing

TST 101

Fundamentals of Security Testing

Duration: 120 minutes
Languages: English, Japanese

This course introduces security-testing concepts and processes that will help students analyze an application from a security perspective and to conduct effective security testing. The course focuses on the different categories of security vulnerabilities and the various testing approaches that target these classes of vulnerabilities. Several manual and automated testing techniques are presented which will help identify common security issues during testing and uncover security vulnerabilities.

TST 191

Fundamentals of Security Testing for Embedded Systems

Duration: 30 minutes
Languages: English

This course module provides additional Fundamentals of Security Testing training of particular importance to embedded software engineers. The module contains the following features: Mapping of content to specific compliance and regulatory requirements Links to key reference resources that support the topics covered in the module “Knowledge Check” quiz that assesses mastery of key concepts.

TST 201

Classes of Security Defects

Duration: 180 minutes
Languages: English, Japanese, Chinese

This course equips students with the knowledge needed to create a robust defense against common security defects. Students will learn why and how security defects are introduced into software, and will be presented with common classes of attacks, which will be discussed in detail. Along with examples of real life security bugs, students will be shown techniques and best practices that will enable the team to identify, eliminate, and mitigate each class of security defects. Additional mitigation techniques and technologies are described for each class of security defect.

TST 291

Classes of Security Defects for Embedded Systems

Duration: 30 minutes
Languages: English

This course module provides additional training on Security Defects Classes of particular importance to embedded software engineers. It includes mapping of content to specific compliance and regulatory requirements, links to key reference resources that support the topics covered in the module, and a “Knowledge Check” quiz that assesses mastery of key concepts.

TST 211

How to Test for the OWASP Top 10

Duration: 90 minutes
Languages: English, Japanese

The Open Web Application Security Project (OWASP) Top Ten is a listing of critical security flaws found in web applications. Organizations that address these flaws greatly reduce the risk of a web application being compromised, and testing for these flaws is a requirement of the Payment Card Industry Standards (PCI-DSS) as well as other regulatory bodies. This course explains how these flaws occur and provides testing strategies to identify the flaws in web applications.

TST 401

Advanced Software Security Testing – Tools & Techniques

Duration: 120 minutes
Languages: English, Japanese

This course delves deeply into the techniques for testing specific security weaknesses. The class is broken down into the three areas where bugs are most often found: insecure interaction between components, risky resource management, and poor defenses. Tools and techniques for security testing are presented, including ten different types of attacks such as SQL Injection, Command Injection, Cross-site Scripting, Buffer Overflow and Access Spoofing. After taking this course, the student will be able to understand the ten types of attacks; know which tools to use to test for these attacks; test software applications for susceptibility to the ten specific attacks; describe the expected mitigations required to prevent these attacks.

TST 411

Exploiting Buffer Overflows

Duration: 120 minutes
Languages: English, Japanese

This course provides students with all the required information to help understand and mitigate buffer-overflow exploits. It first introduces the concepts necessary to recognize the threats posed by these exploits, and to comprehend the mechanisms behind exploitation of stack-based and heap-based buffer overflows. The course then delves into the different challenges faced by exploit code and how different exploitation techniques overcome environmental limitations.

TST 491

Advanced Software Security Testing for Embedded Systems

Duration: 30 minutes
Languages: English

This course module provides additional Software Security Testing of particular importance to embedded software engineers. The module contains the following features: Mapping of content to specific compliance and regulatory requirements Links to key reference resources that support the topics covered in the module “Knowledge Check” quiz that assesses mastery of key concepts.